IMPACT AND ADVANCES OF AUTOMATIC CONTROL IN LATINAMERICA

EDITORS

Dr. O. Lucía Quintero (EAFIT)

Dr. Oscar Camacho (ULA - EPN)

Dr. Gerardo Espinosa-Perez (UNAM)

Dr. Gustavo Scaglia (UNSJ)

Dr. Danilo Chavez (EPN)

From the Conference proceedings of the
XVII LATIN AMERICAN CONFERENCE IN AUTOMATIC CONTROL
Universidad EAFIT
School of Sciences
2016

Editorial Artes y Letras S.A.S

IMPACT AND ADVANCES OF AUTOMATIC CONTROL IN LATINAMERICA

Editors

Oscar Camacho, Danilo Chávez, Gerardo Espinosa-Pérez,

O. Lucia Quintero M and Gustavo Scaglia.

From the Conference proceedings of the

XVII LATIN AMERICAN CONFERENCE IN AUTOMATIC CONTROL

Universidad EAFIT

School of Sciences

2016

Editors

Oscar Camacho, Dr.
Investigador PROMETEO
Departamento de Automatización y Control Industrial.
Facultad de Ingeniería Eléctrica y Electrónica.
Escuela Politécnica Nacional. Quito
Ecuador.
Profesor Titular.
Escuela de Ingeniería Eléctrica.
Facultad de Ingeniería.
Universidad de Los Andes.
Mérida 5101.
Venezuela
E-Mail: ocamacho@ula.ve; oscar.camacho@epn.edu.ec

Danilo Chávez, Dr.
Profesor Agregado
Jefe del Laboratorio de Robotica y Sistemas Inteligentes
Departamento de Automatizacion y Control Industrial (DACI)
Escuela Politecnica Nacional (EPN)
Quito – Ecuador
Telf: (+593) 2 2976 300 ext 5452
Cel: (+593) 979188086
E-Mail: danilo.chavez@epn.edu.ec

Gerardo Espinosa Pérez, Dr.
Facultad de Ingeniería UNAM
Edificio de Posgrado “Bernardo Quintana"
2o piso
Circuito exterior s/n
Ciudad Universitaria
04510, México D.F.
E-mail: gerardoe@unam.mx
Teléfono: +52 (55) 56.22.30.13

Olga Lucía Quintero Montoya. Dr.
Profesora Asociada
Investigador Senior - Colciencias
Dir. Grupo de Investigación Modelado Matemático
Dir. Académica Doctorado en Ingeniería Matemática
Departamento Ciencias Matemáticas
Escuela de Ciencias
Teléfono (57 4) 261 9500, extensión 9064
E-Mail: oquinte1@eafit.edu.co

Gustavo Scaglia. Dr.
Instituto de Ingeniería Química - UNSJ - Argentina
Investigador Adjunto CONICET - Argentina
Facultad de Ingeniería - Univ. Nac. de San Juan
Tel.: +54 264 4211700 Ext. 453, luego/then 32 - Fax: +54 264 4200289
Av. Lib. San Martín (Oeste) 1109. J5400ARL San Juan - ARGENTINA
E-Mail: gscaglia@unsj.edu.ar

ISBN: 978-958-8483-34-4

Preface

It is a great pleasure to write this preface to the book where articles prepared for the XVII CLCA Latin American Conference of Automatic Control are published. Fifty nine (59) excellent quality articles are certainly very impressive. These articles present the latest advancements and applications in 16 different areas, including the traditional areas and the relatively new biomedical and bioprocesses areas. The authors, all Latin-Americans, come from throughout the American continents and some from Europe.

The main purpose of these conferences is to promulgate advances, sometimes including mistakes that have been made to avoid committing them again. Being the promulgation of advances the main objective, the XVII Conference has achieved its mission with me. Only reading the titles of the articles related to process control has been enough to satisfy it. I'm ready to read and learn.

All this is essential for the advancement of science and technology, and for human relations between the countries represented at the Conference. But there is another aspect that I think it's important, and that is the students - our future. I sincerely hope that the articles in this book, and the presentations at the Conference serve as a motivation to students to engage in the research and application of automatic control.

Congratulations to the Committee of professionals in charge of the Conference, to the authors in their magnificent work, to all the participants, and to the students, good luck!

Carlos A Smith, Ph.D.
Professor Emeritus

Outlook of this book

We are glad to present to the scientific community this book entitled "*Impact and Advances of Automatic Control in Latinamerica*" as a representative and special sample of the contributions from ten countries from America and Europe. For this edition, we selected fifty nine papers surprisingly classified in 16 areas onto the Automatic Control field.

It seems that the recent advances in the knowledge about the human being motivate the development of new theoretical and applied subfields promoting the people welfare as objective function for all our control systems designs and applications.

Artificial intelligence, Automation, Biomedical/bioengineering, Bioprocesses, Intelligent control, Linear systems, Mobile robots, New energies, Nonlinear systems, Observers, Optimization, Power systems, Process control, Process control/green houses, Robotics and Signal processing are the selected topics that we want to share with students, researchers and practitioners.

Even if we couldn't agree about the classification of the papers in those areas, we are very happy to introduce the result of the hard work of many people that generously shared their new knowledge and smart solutions to specific problems with our very hard-to-believe community.

Here you can find a compendium of contributions that reflect such a wide variety in our field. From the mathematical formalism of the Control Theory and Signal Processing to the concern about the solution of problems related to traffic, new energies and differently abled people, among others.

We hope you enjoy as much as we did the reading of these papers and highly encourage you to follow the steps of the ones that everyday work to do Science from the Control Engineering point of view.

We would like to acknowledge Luz Elena and Alejandro for their incomparable support.

The Editors

List of contributions (in alphabetical order)

International Reviewers

Adriana Natacha Amicarelli	INAUT - Argentina
Andrea Angel Zea	EMGESA - Colombia
Antonio Morán	PUCP - Perú
Carlos Alberto Cadavid Moreno	UNIVERSIDAD EAFIT - Colombia
Carlos Andrés Sánchez López	CELSIA - Colombia
Ceso de la Cruz Casaño	Universidad Continental – Perú
César A Uribe	University of Illinois at Urbana-Champaign
Daniel Sierra-Sosa	UNIVERSIDAD EAFIT – Colombia
David Ortiz	UNIVERSIDAD EAFIT – Colombia
Emanuel Serrano	IIQ- UNSJ – Argentina
Fabian Leonardo Jaramillo Palacios	Universidad de Cuenca – Ecuador
Flavio Roberti	INAUT – Argentina
Gustavo Pérez	PUCP – Perú
Héctor Antonio Botero Castro	UNAL – Colombia
Jan. H. van Schuppen	TUDelft – The Netherlands
Javier Sotomayor M	PUCP – Perú
Jesús Antonio Hernandez Riveros	UNAL – Colombia
Juan Fernando García Tirado	ITM – Colombia
Juan Carlos Rivera	UNIVERSIDAD EAFIT – Colombia
Juan Diego Sánchez Torres	ITESO – México
Juan Guillermo Paniagua Castrillón	ITM – Colombia
Juan Marcos Toibero	INAUT – Argentina
Manuel Betancur Betancur	UPB – Colombia
Maria Gulnara Baldoquín de la Peña	UNIVERSIDAD EAFIT – Colombia
Mario Fernández-Fernández	Universidad de Talca – Chile
Natalia Martina López	GATEME – UNSJ- Argentina
Pablo Santiago Rivadeneira	UNAL – Colombia
Ricardo Carelli	INAUT – Argentina
Sergio Ponce	UTN - Argentina

A Hamiltonian approach for stabilization of Microgrids including Power converters dynamic ⋆

Sofía Avila-Becerril * **Gerardo Espinosa-Pérez** *

* *Facultad de Ingeniería, Universidad Nacional Autónoma de México, Edificio de Posgrado, 2o. piso, Ciudad Universitaria, 04510 México D.F., MEXICO (e-mail: soavbec@comunidad.unam.mx; gerardoe@unam.mx).*

Abstract: The Microgrids are part of a special class of power systems that offer an attractive option for the use of sustainable energy. The study imposes several challenges from the point of view of control; in the literature this problem has been addressed by assuming that the dynamics of power converters has been drastically simplified. In this article, we exploit the Hamiltonian structure exhibited by the network and we develop a distributed control scheme for a mesh topology including dynamic converters.

Keywords: Power Systems, Hamiltonian, Microgrids, Passivity.

1. INTRODUCTION

The evolution of Electric Power Systems has lead to the conception of a special kind of networks called *smart grids* (Farhangi (2010); Fang et al. (2012)) which join information technology with power systems engineering and has caused a change in the paradigm of Electric Power Systems. The *Microgrids* are part of these intelligent networks and according to Fang et al. (2012) are groups that combine loads, lines and distributed generation sources (e.g. solar panels and small wind turbines) interconnected with the main network via power converters.

On the one hand, Microgrids offer an attractive solution for sustainable energy power supply since they are based on the use of renewable energy sources, leading to a semi-autonomous distributed generation network capable to satisfy the requirements of (relatively) small communities as stated in Guerrero et al. (2013). On the other hand, the aforementioned characteristics still imposes theoretical challenges (see Hill and Chen (2006)) like stability properties analysis, design of protocols for reliable energy dispatch, achievement of power quality standards, among others.

From an structural point of view, the main complication to design, analyze and control, comes from the fact that due to the heterogeneous nature of the energy sources it is necessary to include, for each of them, a power converter whose objective is to shape the generated energy to make it compatible with the rest of the system. This situation imposes a two–level control problem since the power converters must be individually controlled first, to later on approach the control problem of the complete grid. The usual way to address the problem is with the so-called *droop control* (see Lasseter (2002); Barklund et al. (2008); Pedrasa and Spooner (2006); Marwali et al. (2007) and references therein) where, in general, assumptions as constant voltage amplitudes and a simplified model of the power converters are needed. While the results are based on graph theory. Simpson-Porco et al. (2013) show that the model of the Microgrid, consisting of loads and power converters equipped with frequency droop controllers, can be equivalently represented as the model of coupled oscillators. In this case, the attention is restricted to the active power flow under the assumption that the voltage magnitudes are fixed on each bus. In another general study, Schiffer et al. (2014) assume that the Microgrid has been reduced by the Kron reduction (also see Dorfler and Bullo (2013)). These converters are modeled as a chain of integrators, so that all nodes have a power converter and a frequency and voltage droop controllers can be implemented, the last allows them to propose a stability analysis of the Microgrid in terms consensus.

It is clear that considering simplified models for the power converters limits the possibility for including important phenomena exhibited by Microgrids like the related with power quality issues and disturbances during the operation of the power electronics based devices. Thus, in this paper the control problem of Microgrids including the power converters dynamic is approached.

This contribution considers, as illustrative case study, a *meshed* network equipped with energy sources that are connected to the grid via *DC/AC converters*. The control problem is to design a control law for the converters such that the closed–loop system achieves a prescribed power flow. The desired power flow is viewed as a desired trajectory for the Microgrid and the corresponding tracking control problem is solved. For this, it is exploited the Hamiltonian structure exhibited by the power converters and the network itself. In this sense, the whole system is represented as the interconnection of port-controlled

⋆ Part of this work was supported by DGAPA-UNAM under grant IN116516.

Hamiltonian systems via another port-controlled Hamiltonian system (PCH). As a result, the controller design and the stability analysis are remarkable simplified. Relaxations of some assumptions, technical proofs and extensive numerical testing are deferred to a journal article follow.

The rest of the paper is organized in the following way: Section 2 presents the Hamiltonian model for the meshed network. Section 3 is devoted to the power converter model and the design of local controllers. In Section 4 the complete system is presented and its stability properties are stated. Finally, the desired steady–state behavior is formulated in Section 5.

2. MICROGRID STRUCTURE

A Microgrid is an electrical network connected to a distribution system, of low and medium voltage, combining (typically renewable) energy generating units and loads. In the network can be identified: Distributed Generation (DG) units, power converters, transmission lines and loads. In this section the mathematical model of each component is presented from PCH perspective, later to present a comprehensive and modular model of Microgrid. Finally, the control law for the inverters are extended to a distributed control law that stabilizes the complete Microgrid.

2.1 Network dynamical model

In this section the structure and dynamic of the network that interconnects the generation systems (i.e. power converters) with the loads are addressed. The network is viewed as an electrical circuit showing that its dynamic behavior corresponds to the exhibited by a Hamiltonian system. An approach based on Graph theory (see Bollobás (1998)) that closely follows Avila-Becerril et al. (2015) is considered.

Consider that the power network is represented by a graph composed by n nodes and b edges. Hence, there exist $n-1$ independent current constraints and $b-n+1$ independent voltages constrains, established by Kirchhoff laws. Consider now a given tree of the circuit (integrated by the n nodes and $n-1$ edges such that no loops are formed) and its corresponding co-tree (given by the set of $b-(n-1)$ edges that do not belong to the tree). Under these conditions and exploiting the concepts of basic cutsets and loopsets of the graph (also see Wellstead (1979)), the current and voltage constraints can be obtained as

$$\begin{bmatrix} I & H \end{bmatrix} \begin{bmatrix} i_t \\ i_c \end{bmatrix} = 0, \qquad \begin{bmatrix} -H^T & I \end{bmatrix} \begin{bmatrix} v_t \\ v_c \end{bmatrix} = 0 \tag{1}$$

where the matrix $H \in \mathbb{R}^{(n-1)\times b-(n-1)}$, known as the *fundamental loop matrix*, completely characterizes the topology of the circuit, while I is a generic identity matrix of proper dimensions. From (1) above, it is clear that the structure of H determines how the tree currents and the co-tree voltages can be generated as linear combinations of the co-tree currents and the tree voltages, respectively, via

$$i_t = -Hi_c, \quad v_c = H^T v_t. \tag{2}$$

The entries of H (and H^T) are 1 if a co-tree current (a tree voltage) points into a given basic ambit (decreases in the same direction than a given basic loop), -1 if points out the basic ambit (if decreases in the opposite direction than the basic loop) and 0 if does not belong to the basic cutset (basic loopset).

As usual, the lumped elements are of three kinds, namely, sources, energy stores (inductors and capacitors) and dissipators. Besides, as in Brayton and Moser (1964), it is considered that the circuit is complete, so that the sources, the capacitors and some (voltage-controlled) dissipators are considered as branches while all the inductors and some (current-controlled) dissipators are chords, leading to the partitions

$$i_t = \begin{bmatrix} i_1 \\ i_C \\ i_{Rt} \end{bmatrix};\ v_t = \begin{bmatrix} v_1 \\ v_C \\ v_{Rt} \end{bmatrix};\ i_c = \begin{bmatrix} i_{Rc} \\ i_L \end{bmatrix};\ \ v_c = \begin{bmatrix} v_{Rc} \\ v_L \end{bmatrix} \tag{3}$$

where $v_1, i_1 \in \mathbb{R}^{n_1}$ are the port variables of the sources, $v_C, i_C \in \mathbb{R}^{n_2}$ the variables associated with the capacitors and $v_{Rt}, i_{Rt} \in \mathbb{R}^{n_3}$ the corresponding to the tree dissipators, such that $n_1+n_2+n_3 = n-1$. For the co-tree variables, $v_{Rc}, i_{Rc} \in \mathbb{R}^{n_4}$ conform the co-tree dissipators, while $v_L, i_L \in \mathbb{R}^{n_5}$ the inductors, with $n_4+n_5 = b-(n-1)$.

Under the partition presented, the matrix H takes the form

$$H = \begin{bmatrix} H_{1R} & H_{1L} \\ H_{CR} & H_{CL} \\ H_{RR} & H_{RL} \end{bmatrix} \tag{4}$$

where each sub-matrix, of appropriate dimensions, represents the interconnections among the different tree and co-tree elements.

The port variables of the capacitors and inductors are related with the energy function of the network $H_a : \mathbb{R}^{n_2\times n_5} \to \mathbb{R}_{\geq 0}$ defined as

$$H_a(q_C, \lambda_L) = \sum_{i=1}^{n_2} H_{aCi}(q_{Ci}) + \sum_{i=1}^{n_5} H_{aLi}(\lambda_{Li}) \tag{5}$$

where $q_{Ci} \in \mathbb{R}$ is the i-th entry of the electrical capacitor charges vector $q_C \in \mathbb{R}^2$ and $\lambda_{Li} \in \mathbb{R}$ the i-th entry of the linkage inductor fluxes vector $\lambda_L \in \mathbb{R}^5$. Then, it holds that

$$\dot{q}_{Ci} = f_{Ci};\ \ e_{Ci} = \frac{\partial H_a(q,\lambda)}{\partial q_{Ci}};\ \ i = 1,\dots,n_2 \tag{6a}$$

$$\dot{\lambda}_{Li} = e_{Li};\ \ f_{Li} = \frac{\partial H_a(q,\lambda)}{\partial \lambda_{Li}};\ \ i = 1,\dots,n_5 \tag{6b}$$

While the dissipators satisfies

$$f_{Rti} = \psi_{ti}(e_{Rti});\ \ e_{Rci} = \psi_{ci}(f_{Rci}) \tag{7}$$

where $\psi_{ti}(\cdot)$, $\psi_{ci}(\cdot)$ are assumed bijective functions and f_{Rti}, e_{Rti}, e_{Rci}, f_{Rci} are, respectively, the i-th entry of $f_{Rt} \in \mathbb{R}^{n_3}$, $e_{Rt} \in \mathbb{R}^{n_3}$, $e_{Rc} \in \mathbb{R}^{n_4}$, $f_{Rc} \in \mathbb{R}^{n_4}$.

Microgrid's Topology. Typical topologies can be found in Avila-Becerril et al. (2015), in terms of the *fundamental loop matrix*. In this paper, the analysis is focused in the so–called *Mesh* network since it states the more general and reliable topology. We assume that each transmission line is modeled by the π model Kundur et al. (1994), *i.e.*, a series circuit composed by a resistor and an inductor with a shunt capacitor. Under this structure, the lines are two–port systems giving the possibility to connect either a source or a load.

The structure of a Mesh topology satisfies:

F.1. All the sources are connected to all the loads via transmission lines.
F.2. The loads are parallel connected with one capacitor of the π model.
F.3. The sources are parallel connected with one capacitor of the π model.
F.4. There exist transmission lines that connects a source with another source.

As shown in Avila-Becerril et al. (2015), if the resistors involved in the π model are considered as tree dissipators then $n_3 = n_5$ and identifying the co–tree dissipators with the loads, $H_{RL} = I$, $H_{RR} = 0$ of proper dimension, while $H_{CR} = I$ and $H_{1R} = 0$. In addition, due to **F.1** and **F.2**, $n_1 = n_2 = n_4$ while **F.3** leads to the elimination, for analysis purposes, of each of the capacitors parallel connected to the sources.

Even that the network model can be formulated under this general scenario Avila-Becerril et al. (2015), with the aim to, first, situate the contribution with respect the related literature Schiffer et al. (2014), Simpson-Porco et al. (2013), and second, to simplify the presentation, the following assumption is considered:

A.1. The network is lossless or dominantly inductive.

Under assumption **A.1**, the tree dissipators are no longer included. The model can be obtained from equations (1) and (6), under F.1-4. An advantage of this model lies in property that it exhibits a port–controlled Hamiltonian structure (Van der Schaft (1999)). Indeed, defining the state variables $x_3 = q_C$ and $x_4 = \lambda_L$, the model can be written in matrix form as

$$\dot{x}_{34} = \mathbb{J}_{34}\nabla_{x34}H_a(x_3, x_4) - F_{34}f_{Rc} + G_{34}e_1 \qquad (8)$$

where $x_{34} = \begin{bmatrix} x_3^T & x_4^T \end{bmatrix}^T \in \mathbb{R}^{(n_1+n_3)}$, $\nabla_{x34}H_a(x_3, x_4) = \frac{\partial H_a(x_3,x_4)}{\partial x_{34}}$ and

$$\mathbb{J}_{34} = \begin{bmatrix} 0 & -H_{CL} \\ H_{CL}^T & 0 \end{bmatrix} = -\mathbb{J}_{34}^T; \;\; G_{34} = \begin{bmatrix} 0 \\ H_{1L}^T \end{bmatrix}$$

$$\mathbb{F}_{34} = \begin{bmatrix} I \\ 0 \end{bmatrix}$$

subject to the constraints

$$f_1 = H_{1L}f_L, \quad e_{Rc} = e_C = \psi_c(f_{Rc}) \qquad (9)$$

This equation shows the demanded current to the sources as a linear combination of the transmission line currents connected to them. In the sequel, it is assumed ideal sources in the sense that they can provide any amount of current, so that only steady state stability issues can be approached.

3. SOURCES AND LOADS DYNAMICAL MODELS

In this section we study the inclusion of power converter models into the Microgrid description and loads modeling from an energy–based perspective.

3.1 Source model

In the sequel, sources are viewed as a one–port dynamical systems with port–variables given by the output voltage e_{1i} and the delivered current f_{1i} with $i = 1, 2, \ldots, n_1$. These systems are composed by a microsource that fed a power converter. In this paper the microsources are considered as constant voltage sources. Note that including a source dynamic in a Hamiltonian framework states only an additional step in the formulation of a single Hamiltonian system that describes the whole Microgrid.

In Figure 1 is presented the topology of the i-th electronic inverter, *i.e.* it consist of the aforementioned voltage source which delivers a voltage $V_i \in \mathbb{R} > 0$, a switching array, that modulates the input voltage via the signal $u_i \in \mathbb{R}$ and a second order LC filter.

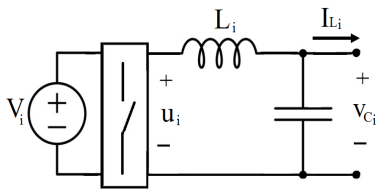


Fig. 1. Illustrative power converter DC/AC

Let $x_{1i} \in \mathbb{R}$ denote the linkage inductor flux and $x_{2i} \in \mathbb{R}$ the electrical capacitor charge of each converter. From a direct application of Kirchhoff laws, the dynamic model for the i-th device, $i = 1, 2, \ldots, n_1$, is given by

$$\dot{x}_{1i} = -C_i^{-1}x_{2i} + Vu_i \qquad (10a)$$

$$\dot{x}_{2i} = L_i^{-1}x_{1i} - f_{1i} \qquad (10b)$$

where it has been assumed a linear constitutive relationship for both the inductors and the capacitors with $L_i \in \mathbb{R} > 0$ the inductance and $C_i \in \mathbb{R} > 0$ the capacitance.

If $H_{ci} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_{\geq 0}$ is the stored energy function, given by

$$H_{ci}(x_{1i}, x_{2i}) = \frac{1}{2}L_i^{-1}x_{1i}^2 + \frac{1}{2}C_i^{-1}x_{2i}^2, \qquad (11)$$

then, model (10), can be equivalently written as a port–controlled Hamiltonian system of the form

$$\dot{x}_{12i} = \mathbb{J}_{12i}\nabla_{x12i}H_{ci}(x_{1i}, x_{2i}) + G_{12i}u_i - \begin{bmatrix} 0 \\ f_{1i} \end{bmatrix} \qquad (12)$$

where $x_{12i} = \begin{bmatrix} x_{1i} & x_{2i} \end{bmatrix}^T \in \mathbb{R}^2$ while

$$\mathbb{J}_{12i} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = -\mathbb{J}_{12i}^T; \;\; G_{12i} = \begin{bmatrix} V_i \\ 0 \end{bmatrix}$$

In this context, the port-variables of each converter are the capacitor voltage $e_{1i} = C_i^{-1}x_{2i} \in \mathbb{R}$ and the output current $f_{1i} \in \mathbb{R}$.

The structure considered for the power converters is only illustrative. Actually, the switching array is just a representation of several topologies used in practice and has been over simplified for presentation purposes. However, it has been widely shown (see for example Ortega et al. (2013)) that common converter topologies can be represented under a Hamiltonian structure.

It is now convenient to illustrate how the output voltage of the converter can be controlled to achieve a prescribed value. To do this, it is necessary to introduce the concept of *admissible trajectories* which refers to the set of state trajectories that a given dynamical system can reproduce

under a specific input. For the defined Microgrid sources, this set is given by the solutions of

$$\dot{x}^{\star}_{12i} = \mathbb{J}_{12i}\nabla_{x12i\star}H^{\star}_{ci} + G_{12i}u^{\star}_i - \begin{bmatrix} 0 \\ i_{Li} \end{bmatrix}, \qquad (13)$$

that recover the behavior of (12) under the input $u^{\star}_i$.

Equation (13) allows to identify $x^{\star}_{12i}$ as the desired steady–state behavior for the sources variables. Thus, it is possible to immediately formulate a control problem by defining the error signal $\tilde{x}_{12i} = x_{12i} - x^{\star}_{12i}$ and finding the control input u_i that renders this variable to zero. In the next proposition a controller that achieves asymptotic stabilization of the prescribed behavior for each non interconnected power converters is developed, leaving to Section 4 the Microgrid stabilization.

Proposition 1. Consider a DC/AC power converter of the form (12) and assume

A.2 The state x_{12i} and the output current f_{1i} are available for measurement.
A.3 The parameters L_i and C_i are known.
A.4 The prescribed steady–state behavior $x^{\star}_{12i}$ is known.

Under these conditions, the control law

$$u_i = V_i^{-1}[\dot{x}^{\star}_{1i} + C_i^{-1}x^{\star}_{2i} - K_{1i}L_i^{-1}\tilde{x}_{1i}] \qquad (14)$$

with the desired state satisfying

$$\dot{x}^{\star}_{2i} - L_i^{-1}x^{\star}_{1i} + i_{Li} - K_{2i}C_i^{-1}\tilde{x}_{2i} = 0 \qquad (15)$$

and $K_{1i} > 0$ and $K_{2i} > 0$, guarantees that

$$\lim_{t\to\infty} \tilde{x}_{12i} = 0$$

Proof. From (12) and (13) the error dynamic is given by

$$\dot{\tilde{x}}_{12i} = \mathbb{J}_{12i}\nabla_{\tilde{x}12i}\tilde{H}_{ci}(\tilde{x}_{1i}, \tilde{x}_{2i}) + G_{12i}\tilde{u}_i \qquad (16)$$

where

$$\tilde{H}_{ci}(\tilde{x}_{1i}, \tilde{x}_{2i}) = \frac{1}{2}L_i^{-1}\tilde{x}^2_{1i} + \frac{1}{2}C_i^{-1}\tilde{x}^2_{2i} \qquad (17)$$

and $\tilde{u}_i = u_i - u^{\star}_i$.

On the other hand, (14) and (15) can be written as

$$G_{12i}\tilde{u}_i = -K_{12i}\nabla_{\tilde{x}12i}\tilde{H}_{ci}(\tilde{x}_{1i}, \tilde{x}_{2i}) \qquad (18)$$

with $K_{12i} = diag\{K_{1i}, K_{2i}\} \in \mathbb{R}^{2\times 2}$. Substitution of (18) in (16) leads to the closed–loop error dynamic

$$\dot{\tilde{x}}_{12i} = [\mathbb{J}_{12i} - K_{12i}]\nabla_{\tilde{x}12i}\tilde{H}_{ci}(\tilde{x}_{1i}, \tilde{x}_{2i}) \qquad (19)$$

If $\tilde{H}_{ci}(\tilde{x}_{1i}, \tilde{x}_{2i})$ in (17) is considered as a Lyapunov function candidate, its time derivative along the trajectories of the error dynamic (19) is given by

$$\dot{\tilde{H}}_{ci}(\tilde{x}_{1i}, \tilde{x}_{2i}) = -\left(\nabla_{\tilde{x}12i}\tilde{H}_{ci}\right)^T K_{12i}\nabla_{\tilde{x}12i}\tilde{H}_{ci} < 0$$

expression that, due to the linear structure of the Lyapunov function, implies the claimed asymptotic stability property.

□

In Section 4 it is shown that these stability properties are preserved if these devices are interconnected through the network.

3.2 Load model

Modeling of electrical loads in a Hamiltonian framework is a topic that has been widely studied since many years ago Ortega et al. (2013). It is clear that the port–variables of the loads attached to the network, the current f_{Rc} and the voltage e_{Rc}, can be related with a port–controlled Hamiltonian system. Moreover, if the interconnection between this system and the network (8) is carried out in a power preserving way, it is obtained another Hamiltonian system.

We now assume that the system operates around a prescribed admissible trajectory, as already illustrated in (13) for the power converter sub–system. Hence, the concept of *Incremental Passivity*, studied in Pavlov and Marconi (2008) and in the case of electrical circuits in Jayawardhana et al. (2007), states a way to characterize the input–output behavior of the loads via the following assumption

A.5 The port–variables of each load, related by the constitutive relationship (9), satisfies the incremental passivity condition

$$(e_{Rc} - e^{\star}_{Rc})^T\left[\psi_c^{-1}(e_{Rc}) - \psi_c^{-1}(e^{\star}_{Rc})\right] > 0$$

for a given admissible trajectory $e^{\star}_{Rc}(t) \in \mathbb{R}^{n_1}$.

4. MICROGRID STABILIZATION

In this section a complete model for the approached Microgrid is presented. Consider first the interconnection between the sources (power converters) and the network. Notice that the n_1 individual power converters of the form (12) can be pilled up as

$$\dot{x}_1 = -C^{-1}x_2 + Vu$$
$$\dot{x}_2 = L^{-1}x_1 - f_1$$

with total stored energy function

$$H_c(x_1, x_2) = \frac{1}{2}x_1^T L^{-1}x_1 + \frac{1}{2}x_2^T C^{-1}x_2 \qquad (20)$$

where $x_1 = col\{x_{1i}\} \in \mathbb{R}^{n_1}$, $x_2 = col\{x_{2i}\} \in \mathbb{R}^{n_1}$, $V = diag\{V_i\} \in \mathbb{R}^{n_1\times n_1}$, $u = col\{u_i\} \in \mathbb{R}^{n_1}$, $L^{-1} = diag\{L_i^{-1}\} \in \mathbb{R}^{n_1\times n_1}$ and $C^{-1} = diag\{C_i^{-1}\} \in \mathbb{R}^{n_1\times n_1}$.

On the one hand, the vector $f_1 \in \mathbb{R}^{n_1}$ stand for the currents injected to the network satisfying the constraint (9), which in terms of the network stored energy takes the form

$$f_1 = H_{1L}\frac{\partial H_a(x_{34})}{\partial x_4} \qquad (21)$$

with $H_a(x_{34})$ as in (5) and where it has been used the identity (6b). On the other hand, the voltage of the source ports are given by the output converter voltages

$$e_1 = C^{-1}x_2 \qquad (22)$$

While the variables $f_{Rc}, e_{Rc} \in \mathbb{R}^{n_1}$ of the network load ports, under (7) and (6a), can be represented as

$$f_{Rc} = \psi_c^{-1}\left(\frac{\partial H_a(x_{34})}{\partial x_3}\right) \qquad (23)$$

Representing the power converters model in a Hamiltonian structure, using (20), together with (8), the port variables (21), (22) and (23), leads to a dynamic description of the complete Microgrid.

Proposition 2. The dynamic behavior of the complete Microgrid conformed by the network (8) with sources (10a-10b) and loads satisfying assumption **A.2**, can be represented by the port-Hamiltonian system

$$\dot{x} = \mathbb{J}_T\nabla_x H_T(x) - g_{RT}\Psi(x_{34}) + G_T u \qquad (24)$$

with state $x = \left[\, x_1^T \;\; x_2^T \;\; x_3^T \;\; x_4^T \,\right]^T \in \mathbb{R}^{(3n_1+n_2)}$, the total stored energy function

$$H_T(x) = H_c(x_1, x_2) + H_a(x_3, x_4)$$

and matrices of appropriate dimensions

$$\mathbb{J}_T = \begin{bmatrix} 0 & -I & 0 & 0 \\ I & 0 & 0 & -H_{1L} \\ 0 & 0 & 0 & -H_{CL} \\ 0 & H_{1L}^T & H_{CL}^T & 0 \end{bmatrix} = -\mathbb{J}_T^T; \;\; g_{RT} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & I \\ I & 0 \end{bmatrix};$$

$$G_T u = \begin{bmatrix} Vu \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \Psi(x_{34}) = \begin{bmatrix} 0 \\ \psi_c^{-1}\left(\dfrac{\partial H_a(x_{34})}{\partial x_3}\right) \end{bmatrix}$$

□

We now can develop a control strategy to stabilize the complete grid following the same procedure than the followed in Section 3.1. In this sense, it is necessary to define the *admissible trajectories* which for system (24) are the solutions of

$$\dot{x}^\star = \mathbb{J}_T \nabla_{x^\star} H_T(x^\star) - g_{RT}\Psi^\star(x_{34}^\star) + G_T u^\star \tag{25}$$

where $u^\star \in \mathbb{R}^{n_1}$ is the control input that generates the admissible trajectory $x^\star \in \mathbb{R}^{3n_1+n_2}$. So that, with the definition of the desired system, the error variable is set as $\tilde{x} = x - x^\star$ and their corresponding error dynamic is

$$\dot{\tilde{x}} = \mathbb{J}_T \nabla_{\tilde{x}} H_T(\tilde{x}) - g_{RT}\tilde{\Psi}(\tilde{x}_{34}) + G_T \tilde{u}$$

where $\tilde{u} = u - u^\star$. In this case, the associated stored energy–like function takes the form

$$\tilde{H}_T(\tilde{x}) = \frac{1}{2}\tilde{x}^T P x \tag{26}$$

with the matrix $P = diag\left\{L^{-1}, C^{-1}, C_a^{-1}, L_a^{-1}\right\} > 0$. Under the foregoing scenario, it is possible to formulate the main stabilization result of the paper, i.e. the proof that the local controllers developed for the power converters are capable of stabilize the entire Microgrid. This result is included in the next

Proposition 2. Consider a Microgrid system of the form (24). Assume **A.1**–**A.3** from Proposition 1 are verified and in addition assume that

A.4 The parameters L_a and C_a are known.

Under these conditions, the control law

$$u = V^{-1}[\dot{x}_1^\star + C^{-1}x_2^\star - K_1 L^{-1}\tilde{x}_1] \tag{27}$$

with $K_1 = diag\{K_{1i}\} \in \mathbb{R}^{n_1 \times n_1}$ and the desired state satisfying the constraints

$$\begin{gathered} \dot{x}_2^\star - L^{-1}x_1^\star + H_{1L}L_a^{-1}x_4^\star - K_2 C^{-1}\tilde{x}_2 = 0, \\ \dot{x}_3^\star + H_{CL}\nabla_{x_4^\star}H_T^\star - K_3\nabla_{\tilde{x}_3}\tilde{H}_T + \varphi_c^{-1}(v_C^\star) = 0 \\ \dot{x}_4^\star - H_{1L}^T\nabla_{x_2^\star} - H_{CL}^T\nabla_{x_3^\star} - K_4\nabla_{\tilde{x}_4}\tilde{H} = 0 \end{gathered} \tag{28}$$

where K_1, K_2, K_3, K_4 are diagonal positive gains, guarantees that

$$\lim_{t\to\infty} \tilde{x} = 0$$

Proof. The control law in equations (27) and (28) can be equivalently written as

$$G_T\tilde{u} = -\mathbb{K}_T\nabla_{\tilde{x}}\tilde{H}_T$$

with $\mathbb{K}_T = diag\{K_1, K_2, K_3, K_4\} \in \mathbb{R}^{(4n_1)\times(4n_1)} > 0$. Using this expression and considering (26) as a Lyapunov function, its time derivative along the trajectories of the closed loop system, under **A.5**, yields

$$\dot{\tilde{H}}_T \leq 0,$$

so that $\tilde{H}_T$ is non-increasing and its argument $\tilde{x}$ is bounded. Moreover, since $\dot{\tilde{H}}$ is zero only in $\tilde{x} = 0$ the equilibrium is asymptotically stable.

□

Remark. The importance of the presented result lies in the fact that the controller (27-28) guarantees that, for any admissible trajectory, the error between the actual value of the capacitors parallel connected with the loads, will tend to the desired value $x_3^\star$. Therefore, by ensuring that this desired value corresponds to a sinusoidal function with a prescribed amplitude and frequency, then both voltage and frequency stability of the power network will be achieved.

5. STEADY STATE DESIRED BEHAVIOR

The proposed method for specifying the desired values is based on the following rationale: The steady state behavior of the voltage $C^{-1}x_2^\star$ establish the voltage $C_a^{-1}x_3^\star$, accordingly a natural choice is to associate to each power converter the voltage

$$C_i^{-1}x_{2i}^\star = A_i^\star \sin(\omega_s t + \delta_i^\star),$$

where $\omega_s \in \mathbb{R}$ takes the same value for all the power converters, while the magnitude $A_i : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ and the phase $\delta_i \in \mathbb{R}_{\geq 0} \to \mathbb{S}$ must be determined to get an adequate power flow. Let the complex admittance be denoted as $Y_{ik} := G_{ik} + jB_{ik} \in \mathbb{C}$ with conductance $G_{ik} \in \mathbb{R}$ and susceptance $B_{ik} \in \mathbb{R}$ and let $\mathcal{N}_i$ be the set of neighbors of the $i-th$ node for which $Y_{ik} \neq 0$. That said, the desired active and reactive power at the $i-th$ node for a lossless microgrid $P_i^\star : \mathbb{S}^{n_1+n_2} \times \mathbb{R}^{n_1+n_2} \to \mathbb{R}$ and $Q_i^\star : \mathbb{S}^{n_1+n_2} \times \mathbb{R}^{n_1+n_2} \to \mathbb{R}$, are obtained as

$$P_i^\star = \sum_{k\sim\mathcal{N}_i} |B_{ik}|A_i^\star A_k^\star sin(\delta_i^\star - \delta_k^\star) \tag{29a}$$

$$Q_i^\star = |B_{ii}|A_i^{\star 2} - \sum_{k\sim\mathcal{N}_i} |B_{ik}|A_i^\star A_k^\star cos(\delta_i^\star - \delta_k^\star) \tag{29b}$$

with

$$B_{ii} := \hat{B}_{ii} + \sum_{k\sim N_i} B_{ik}$$

and $\hat{B}_{ii} \in \mathbb{R}$ the shunt susceptance.

The power flow equations above (29) are static and model the network when it is balanced, that is, the net sum of power consumption, injections and dissipated power is zero, and determine the desired steady state operation of the network. The steady state can be determined by finding, for a given set of load conditions, the active and reactive power flow of the network and the magnitudes and phase angles of all nodes.

In other words, one way to generate the desired trajectories is solving the power flow equations (29). We propose to fix the active and reactive power at the loads, through the resistances, and solve equations (29a) and (29b) in order to calculate the desired magnitude $A_i^\star$ and the phases $\delta_i^\star$ of the n_1 nodes with voltage $C^{-1}x_2^\star$. Once founded the voltages $C^{-1}x_2^\star = A_i^\star \sin(\omega_s t + \delta_i^\star)$ that meets the load power demand, the restriction (28) is incorporated for

$x_1^\star \in \mathbb{R}^{n_1}$ such that the control law $u \in \mathbb{R}^{n_1}$ in (14) can be implemented.

Note that the algebraic equations (29) are non linear, both in the voltage and in the angle, therefore the solution involves the use of a numerical methods, for example Newton-Raphson. In the last years, the control community has been engaged in research from solubility conditions of the power flow equations (see for example Simpson-Porco et al. (2015); Dvijotham et al. (2015) and the references therein) to the optimization problem (Madani et al. (2015); Wei and Bandi (2015) between others). However these studies are out of the scope of this paper.

6. CONCLUSION

This paper has addressed the problem of stabilizing a Meshed Microgrid which, unlike as is usual in the literature, has been included the dynamics of the power converters. It has been shown that a distributed control law developed for each source converter preserves its stabilizing properties even when the converters are interconnected to the network through a Mesh topology. Crucial to reach the presented results has been the Hamiltonian structure exhibited by the different devices that conform the grid.

ACKNOWLEDGEMENTS

Part of this work was supported by DGAPA-UNAM under grant IN116516.

REFERENCES

Avila-Becerril, S., Espinosa-Pérez, G., and Fernández-Carrillo, P. (2015). Stability and consensus of electrical circuits via structural properties. *IFAC-PapersOnLine*, 48(13), 111–116.

Barklund, E., Pogaku, N., Prodanović, M., Hernandez-Aramburo, C., and Green, T.C. (2008). Energy management in autonomous microgrid using stability-constrained droop control of inverters. *Transactions on Power Electronics, IEEE*, 23(5), 2346–2352.

Bollobás, B. (1998). *Modern graph theory*, volume 184. Springer Science & Business Media.

Brayton, R. and Moser, J. (1964). A theory of nonlinear networks i. *Quart. Appl. Math*, 22(1), 1–33.

Dorfler, F. and Bullo, F. (2013). Kron reduction of graphs with applications to electrical networks. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 60(1), 150–163.

Dvijotham, K., Chertkov, M., and Low, S. (2015). A differential analysis of the power flow equations. *arXiv preprint arXiv:1506.08814*.

Fang, X., Misra, S., Xue, G., and Yang, D. (2012). Smart gridthe new and improved power grid: A survey. *Communications Surveys & Tutorials, IEEE*, 14(4), 944–980.

Farhangi, H. (2010). The path of the smart grid. *Power and Energy Magazine, IEEE*, 8(1), 18–28.

Guerrero, J.M., Chandorkar, M., Lee, T.L., and Loh, P.C. (2013). Advanced control architectures for intelligent microgrids, part i: decentralized and hierarchical control. *IEEE Transactions on Industrial Electronics*, 60(4), 1254–1262.

Hill, D.J. and Chen, G. (2006). Power systems as dynamic networks. In *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, 4–pp. IEEE.

Jayawardhana, B., Ortega, R., García-Canseco, E., and Castanos, F. (2007). Passivity of nonlinear incremental systems: Application to pi stabilization of nonlinear rlc circuits. *Systems & control letters*, 56(9), 618–622.

Kundur, P., Balu, N., and Lauby, M. (1994). *Power system stability and control*, volume 7. McGraw-hill New York.

Lasseter, R.H. (2002). Microgrids. In *Power Engineering Society Winter Meeting, 2002. IEEE*, volume 1, 305–308. IEEE.

Madani, R., Lavaei, J., and Baldick, R. (2015). Convexification of power flow problem over arbitrary networks. In *IEEE 54th Ann. Conf. Decis. Contr.(CDC)*.

Marwali, M.N., Jung, J., Keyhani, A., et al. (2007). Stability analysis of load sharing control for distributed generation systems. *IEEE Transactions on Energy Conversion*, 22(3), 737.

Ortega, R., Perez, J.A.L., Nicklasson, P.J., and Sira-Ramirez, H. (2013). *Passivity-based control of Euler-Lagrange systems: mechanical, electrical and electromechanical applications*. Springer Science & Business Media.

Pavlov, A. and Marconi, L. (2008). Incremental passivity and output regulation. *Systems & Control Letters*, 57(5), 400–409.

Pedrasa, M.A. and Spooner, T. (2006). A survey of techniques used to control microgrid generation and storage during island operation. In *Proceedings of the 2006 Australasian Universities Power Engineering Conference (AUPEC'06)*, 1–6.

Schiffer, J., Ortega, R., Astolfi, A., Raisch, J., and Sezi, T. (2014). Conditions for stability of droop-controlled inverter-based microgrids. *Automatica*, 50(10), 2457–2469.

Simpson-Porco, J.W., Bullo, F., et al. (2015). A solvability condition for reactive power flow. In *2015 54th IEEE Conference on Decision and Control (CDC)*, 2013–2017. IEEE.

Simpson-Porco, J.W., Dörfler, F., and Bullo, F. (2013). Synchronization and power sharing for droop-controlled inverters in islanded microgrids. *Automatica*, 49(9), 2603–2611.

Van der Schaft, A. (1999). *L2-gain and passivity in nonlinear control*. Springer-Verlag New York, Inc.

Wei, E. and Bandi, C. (2015). Fairness considerations in network flow problems. In *2015 54th IEEE Conference on Decision and Control (CDC)*, 6909–6914. IEEE.

Wellstead, P. (1979). *Introduction to physical system modelling*. Academic Press London.

A Performance Evaluation Approach for Embedded Controllers of Mobile Robots [??]

G.A. Acosta [??] J.A. Saldarriaga, [??] J.A. Jiménez [??]

[*] *Politécnico Colombiano Jaime Isaza Cadavid, Medellín, Colombia, Cr. 40 7-151 (e-mail: gaacosta@elpoli.edu.co).*
[**] *Politécnico Colombiano Jaime Isaza Cadavid, Medellín, Colombia, Cr. 40 7-151 (e-mail: jaime_saldarriaga91091@elpoli.edu.co)*
[***] *Departamento de Ciencias de la Computación y la Decisión, Facultad de Minas, Universidad Nacional de Colombia, Medellín, Colombia, (e-mail: jajimen1@unal.edu.co)*

Abstract: The application of metrics quality of control (QoC) in mobile robotics could be a difficult task because not always is possible access the data of embedded control units of the robot effectors. This paper presents an heuristic geometric-based altrernative for performance evaluation of embedded controllers. Geometric characteristics of simple circular and linear paths described by the robot are considered for performance evaluation of the controllers. For experimental evaluation of our approach, two PI embedded controllers were designed and implemented in a two wheeled mobile robot (2WMR). Results of conducted experiments showed that our approach constitutes an alternative of performance evaluation of embedded controllers when it is not possible to access data of the control units.

Keywords: Mobile robots; PI controllers; Tuning methods; System identification; Performance analysis.

1. INTRODUCTION

Motion control of wheeled mobile robots (WMRs) remains an important research topic because it support motion tasks like path following and trajectory tracking [**?**]. A broad range of approaches to effectors control of WMRs have been reported. **?** used the quantitative feedback theory (QFT) to design a proportional-integral-derivative (PID) controller for a robot with four omnidirectional wheels. **?** present a review motion control of WMRs using model predictive control (MPC), also several experiments to comparison between path following and trajectory tracking for an omnidirectional mobile robot and an unicycle-type mobile robot were conduced and results discussed. Two-wheeled mobile robots (2WMRs) were developed by **?** and **?** and low level motion control implemented trough PID controllers.

Soft computing is another alternative for motion control of WMRs. A complete discussion on the application of fuzzy logic in reactive navigation of mobile robots is presented by **?**. The authors also present a case study in which basic behaviours such as goal reaching, emergency situation, obstacle avoidance, wall following and an action coordination system were implemented using fuzzy logic. **?** designed and implemented a Takagi-Sugeno fuzzy logic controller (FLC) for a two-wheeled inverted pendulum, and **?** developed a 2WMR for cleaning air ducts and corridors based on a reactive navigation architecture in which a fuzzy logic controller allows the robot to follow close walls.

Other works present comparative performance studies of different strategies to control effectors of 2WMRs. A performance analysis of three controllers, including lead-lag compensator, PID controller and FLC for DC motor control of a field survey 2WMR was conduced by **?**. A comparison of FLC, linear quadratic controller (LQR) and a PID controller to balancing the tilt angle of two wheeled inverted pendulum robot is presented by **?**. A comparative investigation about the performance of a PID and an FLC linear position and tilt angle controllers was developed by **?**. In these works, time-domain specifications such as rise time, settling time and percent overshoot were used for performance analysis of controllers. In this paper, we propose an alternative method for motor control evaluation based on geometric characteristics of two simple trajectories described by a 2WMR.

The main contributions of this paper are: 1) an heuristic-based alternative for performance evaluation of low-level motion controllers is provided. Unlike classical time-domain methods, our approach is based on geometrical characteristics of simple and real trajectories described by a robot, 2) design and implementation of two embedded feedback controllers, one for each motor, that receive velocity references and make that motors achieve and maintain these values until new references are received, and 3) implementation of an identification process in which the robot acts as a drive unit and a data acquisition system. For this, a C program for applying PWM signals

[⋆] The authors thank their institutions of origin, Politécnico Colombiano Jaime Isaza Cadavid and Universidad Nacional de Colombia, for the support provided in the development of this work.

to the motors, obtaining data from encoders and wireless transmission of these data was developed.

This article is organized as follows. In this first section, we highlight the importance of motor control in mobile robotics. In section 2 our heuristic geometric-based approach for controller performance evaluation is presented. In section 3 the modelling of robot and effectors is detailed. In section 4, the design procedure of embedded controllers is discussed. In section 5, experiments are conducted and results are presented and, finally, conclusions are drawn in section 6.

2. A GEOMETRIC-BASED APPROACH FOR PERFORMANCE CONTROLLER EVALUATION

In a control loop, the error is defined as the deviation of the controlled variable with respect to a reference value. A way to evaluate the Quality of Control (QoC) of a system is quantifying the cumulative error, for which, in the case of discrete-time controllers, is essential to know the error $e(nT)$ at each sampling instant $T = 1/f_s$. Some performance indices based on cumulative error for QoC evaluation are: Integral of Absolute Error (IAE) and Integral of Squared Error (ISE). However these metrics are difficult or impossible to apply in the field of mobile robotics when motors control is carried out by sealed units that only receive reference values, but do not send actual motor speed data.

In this paper we propose an heuristic method for performance evaluation of controllers based on geometric characteristics of two simple trajectories: linear and circular displacements. This approach can be applied in those cases where it is difficult, if not impossible, to establish the error $e(nT)$.

Metric based on the linear displacements Linear movement of differential robots is obtained when the speeds of its driving wheels (left and right) are exactly equals at all time, this is, when $v_{_L}(t) = v_{_R}(t)$. However, the uncertainty inherent to any real control system, causes, at least for finite intervals of time, that these speeds will be different. Thus, the actual robot path differs from the purely linear. This fact suggests that we could use the deviations of the actual path of the robot with respect to ideal straight-line path, to establish a metric for performance evaluation of motor control schemes on differential robots. Relative to Figure **??**, we define the metric $Cumulative\ Absolute\ Distance - CAD$ such as

$$CAD = \sum_{k=k_0}^{k_f} |d_k|,\ k\ \epsilon\ \mathbb{Z}. \tag{1}$$

where d_k is the *k-th* perpendicular distance between the actual trajectory and the ideal straight-line trajectory.

Metric based on the circular displacements Otherwise, when $v_{_L}(t) \neq v_{_R}(t)$ the robot describes curved paths. An special case is when $v_{_L}(t) = k_1$, $v_{_R}(t) = k_2$, and $k_1 \neq k_2$, in which the path obtained should be to a perfect circle of radius R. But as previously mentioned, due to the uncertainty associated to real control systems, is possible to obtain open circular paths such as shown in **??**. In this figure, ΔC corresponds to *aperture distance* or distance

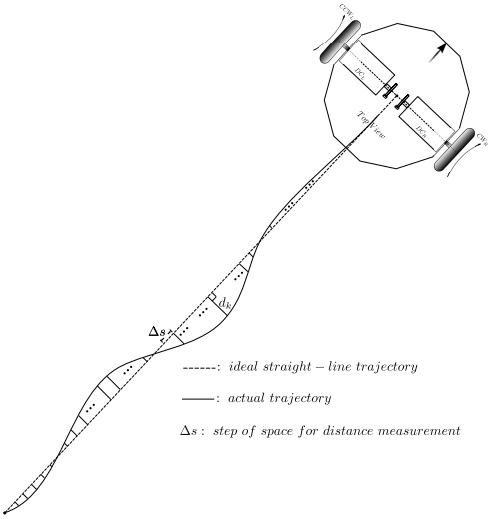


Fig. 1. Linear path and distances for CAD definition

between initial point A and final point B of the described path. Because the actual path does not correspond to a perfect circle, then we define a *circle-base* with center in O and ϕ_r diameter passing through the midpoint P of ΔC.

The diameter of the *circle-base* is set as follows: with center in P, draw an arc that cuts the circular trajectory at C and D points to get the chord $\overline{CD}$. Once the midpoint M of $\overline{CD}$ is set, the $\overline{MP}$ line segment is drawn and extends until cuts the circular path at Q point. In this way we establish as the diameter ϕ_r, of the $circle - base$, the length of $\overline{PQ}$ segment and its midpoint O as its center.

Now we can define the metric $Circular\ Trajectory\ Error - CTE$ such as

$$CTE = |\phi_r - \phi_t|. \tag{2}$$

where the theoretical diameter ϕ_t of the ideal circular trajectory is calculated using Equation **??**.

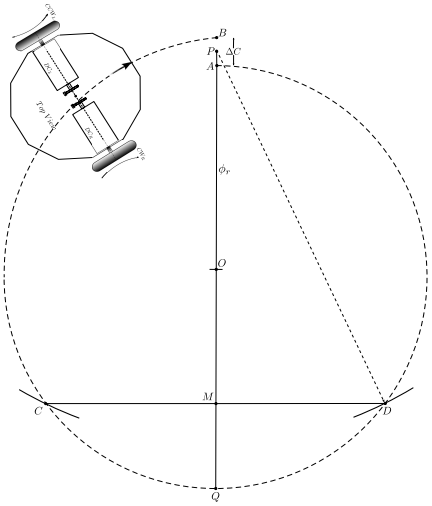


Fig. 2. Open circular path and geometry for CTE definition

3. SYSTEM MODELING

Kinematic Modeling The kinematic modeling refers to the study of motion of a mechanical system without

considering the forces and torques involved. For a 2WMR the Kinematic model allows to express its velocities as functions of velocities of its wheels and its geometric parameters [**?**, **?**].

For 2WMRs a pure rolling motion without slipping is obtained when the robot rotates around an external point located over the common axis of both driving wheels. This point is known as the instantaneous center of curvature (ICC) or instantaneous center of rotation (ICR). The ICC will move by changing velocities of the two driving wheels, allowing the robot carry out different paths. At each time instant, right and left wheels follow paths around the ICC at the same angular velocity $\omega = d\psi/dt$ as is shown in Figure **??**. Table **??** shows the symbols used in this section, with $L = 22.1$ cm. The linear velocities of the driving wheels are given by

$$v_{_L} = \omega(R + \frac{L}{2}). \tag{3}$$

$$v_{_R} = \omega(R - \frac{L}{2}). \tag{4}$$

where R is the distance between the ICC and the middle point C_R between the two wheels. The angular velocity of the robot is obtained subtracting (**??**) from (**??**) and is expressed by

$$\omega = \frac{v_{_L} - v_{_R}}{L}. \tag{5}$$

By adding (**??**) and (**??**) and substituting (**??**) in the result we obtain

$$R = \frac{v_{_L} + v_{_R}}{v_{_L} - v_{_R}} \cdot \frac{L}{2}. \tag{6}$$

Equations (**??**) and (**??**) allow to establish the angular velocity of the robot and the instantaneous radius of curvature as functions of linear velocities of the wheels and their separation L. Finally, the linear velocity of the midpoint $C_{_R}$ is given by the average of the wheel velocities

$$v_{_{CR}} = \omega R = \frac{v_{_L} + v_{_R}}{2}. \tag{7}$$

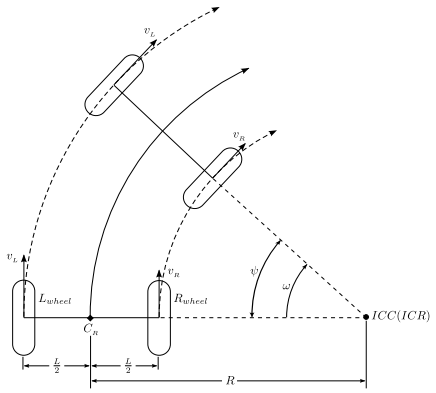


Fig. 3. Differential drive motion of a mobile robot

Effectors Modeling In a previous work [**?**] a teaching and research **Car-li**ke Au**to**nomou**s** vehicle (*Carlitos*) was designed and constructed. This 2WRM is used here for experimental validation of our approach. The block diagram of *Carlitos* is shown in Fig. **??**. Actuation system is composed of two DC motors with coupled gearboxes and a dual H-bridge motor driver for PWM control signals amplification. The proprioceptive sensory system integrates quadrature encoders mounted over the axes of both DC motors, and a CMPS03 digital compass for estimation of robot head orientation. Encoders provide 1125 counts per revolution.

The exteroceptive system is made up of a sonars ring with twelve SRF02 devices for distance measurement to surrounding obstacles. The top side of sonars ring has a 802.15.4 radio for wireless command reception and data transmission. Processing unit correspond to a 32-bit microcontroller capable of operate up 50 MHz. We use the Freescale MCF51QE128 microcontroller to process pulses from encoders and generate PWM control signal for each motor. Control algorithms were coded in C language and programmed in microcontroller's internal flash memory.

Carlitos robot was conceived and designed having in account modularity and scalability criteria.

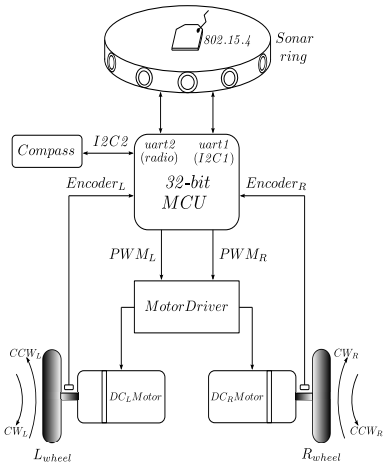


Fig. 4. Block diagram and a picture of the experimental robot

To take advantage of the robot's resources, a C language algorithm was developed to support the effectors modelling task. Initially the robot was placed in an obstacles free indoor environment awaiting the start command sent from a PC via a 802.15.4 radio link. As is shown in Figure **??**, once received the *start* order, an step PWM signal with a duty cycle (DC) of 50% it is generated by the microcontroller to drive both motors. This signal was maintained for 1.8 seconds. Subsequently, the duty cycle of the PWM signal was incremented to 60%. This step was also maintained by 1.8 seconds. During the total

Table 1. Kinematic model variables

Parameter	Description
L	Distance between support points of wheels (cm)
R	Instantaneous radius of curvature (cm)
$C_{_R}$	Midpoint between driver wheels (cm)
ψ	Robot orientation angle (rad)
ω	Angular velocity of 2WMR ($rad \cdot s^{-1}$)
$v_{_L}$, $v_{_R}$	Velocities of wheels along the ground ($cm \cdot s^{-1}$)
$v_{_{CR}}$	Linear velocity of $C_{_R}$ ($cm \cdot s^{-1}$)

displacement (3.6 seconds), the number of pulses per sampling period $T = 1/f_s = 0.1\ s$ was captured from the encoders and sent to a PC via the wireless link. Anything additional resources to those available in mobile robotic platform were required for the actuators modeling.

The databases obtained from the test (Figure **??**-a) were used, in combination with the system identification toolbox$^{\text{TM}}$ of Matlab®, to get a mathematical model of the actuators. We choose an approach of lower-order as the first-order-plus dead-time (FOPDT) to model the dynamic behavior of DC motors. The form of a FOPDT model is given by

$$G(s) = \frac{Ke^{-\theta s}}{\tau s + 1}. \tag{8}$$

With the model parameters $K = 2.63$ (process gain), $\tau = 0.2222$ (time constant), and $\theta = 0.1159$ (dead-time), we obtain

$$G(s) = \frac{2.63e^{-0.1159s}}{0.2222s + 1}. \tag{9}$$

The correlation factor given by the system identification toolbox$^{\text{TM}}$ for the models of both motors was of 94.02%.

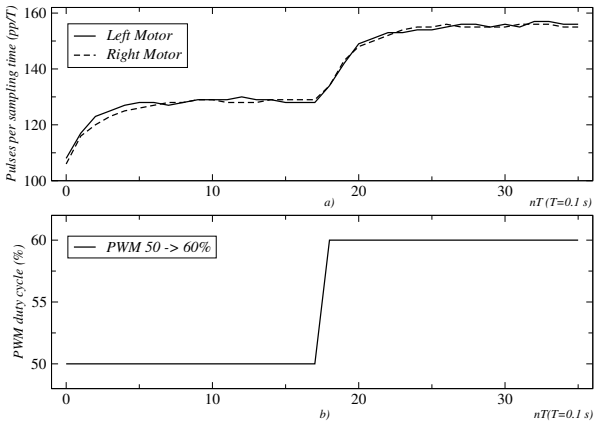


Fig. 5. Step test: a) response of actuators, and b) PWM signal of stimulus

4. EMBEDDED CONTROLLERS DESIGN

Two proportional-integral (PI) feedback controllers were implemented in the robot microcontroller, one for each motor and two model-based tuning methods were applied: Ciancone-Marlin and Cohen-Coon. For a PI discrete-time controller the transfer function is given by

$$D(z) = \frac{q_0 z + q_1}{z - 1}. \tag{10}$$

Next, we present in detail the methods used to determine the q_0 and q_1 parameters.

4.1 Ciancone-Marlin Method

To determine the controller parameters K_c (controller gain) and T_i (integral time), Ciancone and Marlin defined the parametric relationship $fractionaldeadtimeT_f$ such as

$$T_f = \frac{\theta}{\theta + \tau} = \frac{0.1159}{0.1159 + 0.2222} = 0.3428. \tag{11}$$

Additional relationships such as dimensionless gain G_x and dimensionless reset time τ_y were defined by

$$G_x = K_c K. \tag{12}$$

$$\tau_y = \frac{\tau_i}{\theta + \tau}. \tag{13}$$

both expressions correlated with T_f. According to correlation tuning data presented in Table **??** [**?**], $G_x = 1.032$ and $\tau_y = 0.881$.

Table 2. Ciancone-Marlin tunning table for PI controllers

T_f	G_x	τ_y
0.0	1.417	0.748
0.1	1.417	0.748
0.2	1.193	0.964
0.3	1.032	0.881
0.4	0.918	0.818
0.5	0.861	0.756
0.6	0.722	0.693
0.7	0.464	0.631
0.8	0.608	0.568
0.9	0.594	0.506
1.0	0.558	0.443

For discrete-time systems θ_d is calculated as

$$\theta_d = \theta + \frac{T}{2} = 0.1159 + \frac{0.1}{2} = 0.1659. \tag{14}$$

Thus, the parameters of the PI controller are calculated by

$$\tau_i = \tau_y(\theta_d + \tau) = 0.881(0.1659 + 0.2222) = 0.3419. \tag{15}$$

$$K_c = \frac{G_x}{K} = \frac{1.032}{2.63} = 0.3924. \tag{16}$$

with these values q_0 and q_1 are calculated as

$$\begin{aligned} q_0 &= K_c(1 + \frac{T}{2\tau_i}) \\ &= 0.3924(1 + \frac{0.1}{2 \times 0.3419}) = 0.4498. \end{aligned} \tag{17}$$

$$\begin{aligned} q_1 &= -K_c(1 - \frac{T}{2\tau_i}) \\ &= -0.3924(1 - \frac{0.1}{2 \times 0.3419}) = -0.335. \end{aligned} \tag{18}$$

4.2 Cohen-Coon Method

Cohen and Coon noted that the response of many processing units to a change in input had sigmoidal form, which could be approximated to the response of a FOPDT system. The controller parameters could be determined from the parameters of the plant by the following semi-empirical relationships

$$K_c = \frac{\tau}{K\theta}(0.9 + \frac{\theta}{12\tau}) = \frac{0.2222}{2.63 \times 0.1659}(0.9 + \frac{0.1659}{12 \times 0.2222}) = 0.49. \quad (19)$$

$$\tau_i = \frac{\theta(30\tau + 3\theta)}{9\tau + 20\theta} = \frac{0.1659(30 \times 0.2222 + 3 \times 0.1659)}{9 \times 0.2222 + 20 \times 0.1659} = 0.2235. \quad (20)$$

As in the above method q_0 and q_1 are calculated by

$$q_0 = K_c(1 + \frac{T}{2\tau_i}) = 0.49(1 + \frac{0.1}{2 \times 0.2235}) = 0.5996. \quad (21)$$

$$q_1 = -K_c(1 - \frac{T}{2\tau_i}) = -0.49(1 - \frac{0.1}{2 \times 0.2235}) = -0.3804. \quad (22)$$

Figure **??** shows the response of the two controllers to a setpoint speed of 100 pp/T. Note that the wheel speeds are expressed in pulses per sampling time (pp/T). At the top, the response of Ciancone-Marlin controller and, at the bottom, the response of Cohen-Coon controller. A C language program for effectors control, data acquisition and its wireless transmission, was developed and programmed in the internal flash memory of the robot microcontroller. Figure **??** was drawn with the data received. The difference equation for PI controller implementation is given by

$$m(k) = q_0 e(k) + q_1 e(k-1) + m(k-1). \quad (23)$$

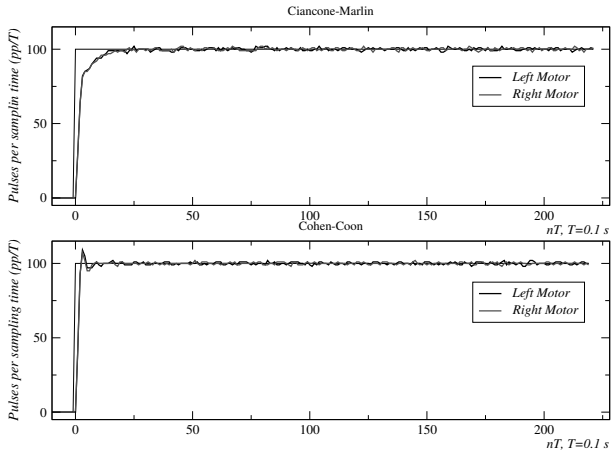


Fig. 6. Step responses of controllers, Ciancone-Marlin (top), and Cohen-Coon (bottom)

5. VALIDATION EXPERIMENTS

Two simple trajectories and its geometrical characteristics were considered for controllers performance evaluation. Equations (**??**) and (**??**) shows how to synthesize the two paths of interest for this study:

i. *Circular path.* The path is a circle of radius R with center in ICC. For locate the ICC outside the line segment that connects both wheels, must meet $|R| = L/2$ condition. The robot moves clockwise when $v_{_L} > v_{_R}$.
ii. *Linear path.* The robot moves straight line when $\omega = 0$, this is met for $v_{_L} = v_{_R}$.

Should be noted that the actual trajectories of the robot differ from the theoretical because of uncertainty in the measurements of the sensors and uncertainty in control commands.

Circular displacement. For validation of CTE metric defined by Equation **??**, three circular paths C_1, C_2 and C_3 were conducted for each controller. The reference velocities for each path were established as, C_1: $v_{_L} = 80$, $v_{_R} = 60$, C_2: $v_{_L} = 100, v_{_R} = 80$, and C_3: $v_{_L} = 120, v_{_R} = 100$. Table **??** presents the experimental results obtained for the Ciancone-Marlin controller and table **??** for Cohen-Coon controller.

Table 3. Results of CTE metric for Ciancone-Marlin controller

Path	ϕ_r (cm)	ϕ_t (cm)	CTE (cm)
C_1	153	148.4	4.6
C_2	193.4	190.8	2.6
C_3	235	233.5	1.5
Mean			2.90

Table 4. Results of CTE metric for Cohen-Coon controller

Path	ϕ_r (cm)	ϕ_t (cm)	CTE (cm)
C_1	151.5	148.4	3.1
C_2	192.6	190.8	1.8
C_3	234.7	233.5	1.2
Mean			2.03

Linear displacement Also, experiments of linear displacement for validation of the CAD metric, defined by Equation **??**, were conducted. This time the set-points of velocity were $v_{_L} = v_{_R} = 100pp/T$ and real trajectories were of approximately 430 centimetres. Step of space Δs (Figure **??**), for distance measurement was of 30 cm. The results obtained are given in table **??**.

Table 5. CAD metrics for embedded controllers

Metric	Ciancone-Marlin	Cohen-Coon
CAD	55.6	50.3

For comparative purposes, two metrics for quality of control (QoC) evaluation were considered, these are: IAE and ISE. These two metrics express QoC in terms of the error $e(t)$, which is defined as the difference between the set-point $r(t)$ and the system output $y(t)$. Continuous-time and discrete-time forms of IAE and ISE are given by

$$IAE = \int_{t_0}^{t_f} |e(t)|dt \approx \sum_{k=k_0}^{k_f} |r(kT) - y(kT)|. \quad (24)$$

$$ISE = \int_{t_0}^{t_f} e(t)^2 dt \approx \sum_{k=k_0}^{k_f} (r(kT) - y(kT))^2. \quad (25)$$

where $t_0(k_0)$ and $t_f(k_f)$ are the initial and final continuous (discrete) times of the evaluation interval and $T = 0.1\ s$. Table **??** shows the results obtained by applying both metrics to the controllers considered.

Table 6. IAE & ISE metrics for embedded controllers

	Ciancone-Marlin		Cohen-Coon	
Metrics	DC_L	DC_R	DC_L	DC_R
IAE	450	453	317	286
ISE	17584	17693	13613	13430

Results analysis Comparing results obtained from the circular path experiments (Tables **??** and **??**), we can observe that the CTE metric was better for Cohen-Coon controller than for Ciancone-Marlin for the three considered trajectories. Similarly, for the linear paths, the best CAD metric (Table **??**) was obtained with the Cohen-Coon controller. In this way, we can conclude that the Cohen-Coon controller offers a better performance for simple trajectories in a 2WMR than the Ciancone-Marlin controller. This result is consistent when we consider the QoC metrics calculated in table **??** for the two controllers. Both, IAE and ISE values, say us that the better performance was obtained with the Cohen-Coon controller.

Our performance evaluation approach of low level motion controllers in 2WMRs is useful in situations where is difficult or impossible to obtain the error from embedded controllers to calculate standard QoC metrics. In that sense, our approach constitute a performance evaluation alternative based on real graphics drawn for the robot during its displacement.

6. CONCLUSION

In this paper, an heuristic geometric-based alternative for performance evaluation of effectors controllers in mobile robots has been presented. Two embedded PI controllers, Ciancone-Marlin and Cohen-Coon, were designed and implemented for motion control of a 2WMR for experimental validation of our approach. Geometric characteristics of circular and linear paths were considered for performance evaluation of both controllers based on two defined metrics: CAD and CTE. Validation experiments and a comparative with IAE and ISE metrics showed that our approach constitutes a valid alternative for performance evaluation of embedded controllers of a mobile robot.

REFERENCES

K. Kanjanawanishkul. Motion control of a wheeled mobile robot using model predictive control: A survey. *KKU Research Journal*, volume 17, issue 5, pages 811–837, 2012.

P. Morin, and C. Samson. Motion control of wheeled mobile robots. In B. Siciliano, O. Khatib, editors, *Handbook of Robotics*, pages 799–826. Springer, 2008.

R. Comasolivas, J. Quevedo, T. Escobet and A. Escobet. Low level control of an omnidirectional mobile robot. *23rd Mediterranean Conference on Control and Automation (MED)*, pages 1160–1166. Torremolinos, Spain, 2015.

A.A. Mahfouz, A.A. Aly, and F.A. Salem. Mechatronics design of a mobile robot system. *I.J. Intelligent Systems and Applications*, volume 5, issue 3, pages 23–36, 2013.

S.K. Malu, and J. Majumdar. Kinematics, localization and control of differential drive mobile robot. *Global Journal of Researches in Engineering: H Robotics & Nano-Tech*, volume 14, issue 1, 2014.

T.S. Hong, B. Karasfi, and D. Nakhaeinia. Application of fuzzy logic in mobile robot navigation. In E. Dadios, editor, *Fuzzy Logic-Controls, Concepts, Theories and Applications*, pages 21–36. InTech, 2012.

J.-X. Xu, Z.-Q. Guo, and T.H. Lee. Design and implementation of a Takagi-Sugeno-type fuzzy logic controller on a two-wheeled mobile robot. *IEEE Trans. Ind. Electron.*, volume 60, issue 12, pages 5717–5728, 2013.

D. Hanafi, Y.M. Abueejela, and M.F. Zakaria. Wall follower autonomous robot development applying fuzzy incremental controller. *Intelligent Control and Automation*, volume 4, issue 1, pages 18–25, 2013.

G.A. Acosta, J.C. Herrera, and O. Zapata. Control de navegación basado en comportamientos para pequeños robots móviles. *Revista Politécnica*, volume 10, issue 18, pages 125–134, 2014.

L.E. García, and M. Arroyave. Desarrollo e implementacion de un controlador con ganancia programable para un autoclave. *Revista Politecnica*, volume 9, issue 1, pages 21–30, 2009.

R. Dhaouadi, and A. Abu Hatab. Dynamic modelling of differential-drive mobile robots using Lagrange and Newton-Euler Methodologies: a unified framework. *Advance in Robotics & Automation*, volume 2, issue 2, pages 1–7, 2013.

S.F.R. Alves, J.M. Rosario, H. Ferasoli Filho, L.K.A. Rincon, and R.A.T. Yamasaki. Conceptual bases of robot navigation modeling, control and applications. In A. Barrera, editor, *Advances in Robot Navigation*, InTech, 2011.

R. Shamshiri, and W.I.W. Ismail. Design and simulation of control systems for a field survey mobile robot platform. *Research Journal of Applied Sciences, Engineering and Technology*, volume 6, issue 23, pages 2307–2315, 2013.

A.A. Bature, S. Buyamin, M. N. Ahmad, and M. Muhammad. A comparison of controllers for balancing two wheeled inverted pendulum robot. *International Journal of Mechanical & Mechatronics Engineering*, volume 14, pages 62–68, 2014.

A.N.K. Nasir, M.A. Ahmad, R. Ghazali, and N.S. Pakheri. Performance comparison between fuzzy logic controller (FLC) and PID controller for a highly nonlinear two-wheels balancing robot. *First International Conference on Informatics and Computational Intelligence*, pages 176–181, 2011.

A Soft Sensor for Biomass in a Batch Process with Delayed Measurements

Jhon A. Isaza * **Juan Diego Sánchez-Torres** **
Esteban Jiménez-Rodríguez *** **Héctor A. Botero** ****

* *Faculty of Engineering and Architecture, Universidad Nacional de Colombia, Sede Manizales, Carrera 27 No 64-60, Colombia (e-mail: jaisazah@unal.edu.co).*
** *Department of Mathematics and Physics, ITESO, Periférico Sur Manuel Gómez Morín 8585 C.P. 45604, Tlaquepaque, Jalisco, México. (e-mail: dsanchez@iteso.mx).*
*** *Department of Electrical Engineering, CINVESTAV-IPN Guadalajara, Av. del Bosque 1145 Col. El Bajío CP 45019, México (e-mail:ejimenezr@gdl.cinvestav.mx).*
**** *Department of Electrical Energy and Automatica, Universidad Nacional de Colombia, Sede Medellín, Carrera 80 No 65-223, Colombia (e-mail: habotero@unal.edu.co)*

Abstract: This paper presents a soft sensor to estimate the biomass concentration in a batch bioprocess used in production of δ-endotoxins of *Bacillus thuringiensis*, subject to delayed measurements. The soft sensor proposed is based on a cascade observer-predictor algorithm. The observer stage is based on a class of second order sliding mode algorithms, allowing a fixed-time estimation of the biomass. Additionally, the prediction stage offsets the effect of the delay in measurements. Simulations show the feasibility of the proposed observer.

Keywords: cascade observer-predictor, delayed measurements, δ-endotoxins production of *Bacillus thuringiensis*, fixed-time observer, Smith predictor.

1. INTRODUCTION

Measuring variables in industrial processes, such as bioprocess, is necessary to carry out tasks of control, diagnosis and fault detection, identification and monitoring (Walcott et al., 1987; Dochain, 2003). For some variables, the work of measurement is hard, costly and difficult to perform due to the unavailability of reliable devices, time delays, errors in the measurement system, high costs of devices and hostile environments for primary measuring devices (Bequette, 2002). Therefore, in order to make estimates by measurements of other variables related directly or indirectly to the variable difficult to measure has been used the state estimators. This dynamic systems are applied to a specific process, with a combination of software and hardware, and they are commonly named as virtual sensors or soft sensors.

However, the soft sensors technology transfer to industrial bioprocesses require to solve some problems such as observer schemes that allowing the use of delayed measurements. To overcome such problem, some authors have developed different methods to incorporate nonuniform and delayed information in state estimation techniques. In (Gopalakrishnan et al., 2011; Guo and Huang, 2015; Guo et al., 2014) have incorporated asynchronous and delayed information to stochastic estimation techniques (Kalman filter and its modifications) but these only apply to discrete systems. Other authors present deterministic estimation techniques with asynchronous and delayed measurement for hybrid systems, with a continuous model for the process and a discrete model for the effects of sensor and sampling. These observers are grouped into three types: Piecewise (Wang et al., 2015), Cascade (Khosravian et al., 2015b,a) and distributed (Zeng and Liu, 2015). This deterministic techniques can to solve the problems of estimating independently or in stages. This feature allows adaptation and extension to solving future problems in state estimation. For example, a mathematical application of a high gain observer in cascade with a predictor was proposed in (Khosravian et al., 2015a). However, a few papers show applications in state estimation in bioprocess with delayed measurements (Zhao et al., 2015).

Therefore, in this paper a cascade observer-predictor for the process of δ-endotoxins production process of *Bt* with fixed time convergence and delayed measurements is considered. The cascade observer-predictor structure is based on the observer presented in (Khosravian et al., 2015b,a) and the Sliding Mode Observer (SMO) proposed in (Sánchez et al., 2015). The proposed observer allows the exact and fixed-time reconstruction of the biomass (vegetative cells and sporulated cells) in the reactor when measurements are delayed.

In the following, the Section 2 presents the mathematical model δ-endotoxins production process of *Bt* with Delayed Measurement. The cascade observer-predictor is presented in Section 3 and presents some mathematical

preliminaries in order to introduce the basics of fixed time stability and predictor stability. The Section 4 presents simulation results of the cascade observer-predictor for the δ-endotoxins production process of *Bt*. Finally, the conclusions of this paper are exposed in the Section 5.

2. BATCH PROCESS MODEL WITH DELAYED MEASUREMENT

The model of the δ-endotoxins production of *Bt* proposed on (Amicarelli et al., 2010; Rómoli et al., 2016) is used. In this paper the block-wise form of the equations is modified to allow a straightforward design of a second order sliding mode observer. The model equations are

$$\begin{aligned}\dot{s}_p &= -\left(\frac{\mu(s_p,o_d)}{y_{x/s}}+m_s\right)x_v\\ \dot{o}_d &= K_3Q_A\left(o_d^*-o_d\right)-K_1\left(\mu(s_p,o_d)-k_e(t)\right)x_v\\ &\quad -K_2\left(x_v+x_s\right)\\ \dot{x}_v &= \left(\mu-k_s(s_p)-k_e(t)\right)x_v\\ \dot{x}_s &= k_s x_v\end{aligned} \tag{1}$$

where s_p is the substrate concentration, o_d is the dissolved oxygen concentration, x_v is the vegetative cells concentration, x_s is the sporulated cells concentration, μ is the specific growth rate, $y_{x/s}$ is the growth yield, m_s is the maintenance constant, Q_A is the airflow that enters the bio-reactor, o_d^* is the oxygen saturation concentration, K_1 is the oxygen consumption dimensionless constant by growth, K_2 is the oxygen consumption constant for maintenance, K_3 is the ventilation constant, k_s is the spore formation kinetics and $k_e(t)$ is the specific cell death rate. Furthermore, the constitutive equations for $\mu(s_p,o_d)$ (Monod-based), $k_s(s_p)$ and $k_e(t)$ are given by:

$$\begin{aligned}\mu(s_p,o_d) &= \mu_{\max}\frac{s_p}{K_s+s_p}\frac{o_d}{K_o+o_d}\\ k_s(s_p) &= k_{s,\max}\left(\frac{1}{1+e^{G_s(s_p-P_s)}}-\frac{1}{1+e^{G_s(s_{p,\mathrm{ini}}-P_s)}}\right)\\ k_e(t) &= k_{e,\max}\left(\frac{1}{1+e^{-G_e(t-P_e)}}-\frac{1}{1+e^{-G_e(t_{\mathrm{ini}}-P_e)}}\right)\end{aligned} \tag{2}$$

where $\mu_{\max}$ is the maximum specific growth rate, K_s is the substrate saturation constant, K_o is the oxygen saturation constant,$k_{s,\max}$ is the maximum spore formation, $k_{e,\max}$ is the maximum specific cell death rate, G_s is the gain constant of the sigmoid equation for spore formation rate, G_e is the gain constant of the sigmoid equation for specific cell death rate, P_s is the position constant of the sigmoid equation for spore formation rate, P_e is the position constant of the sigmoid equation for specific cell death rate, $s_{p,\mathrm{ini}}$ is the initial glucose concentration and t_{ini} is the initial fermentation time.

Assumption 2.1. It is assumed that the measurements of the outputs s_p and o_d are continuously measured with a delay time $\tau > 0$. The delay τ is considered to be known and constant.

Defining $x_1 = s_p$, $x_2 = o_d$, $x_3 = x_v$, $x_4 = x_s$ and considering the Assumption 2.1, the model (1) can be written as:

$$\begin{aligned}\dot{x}_1(t) &= b_1(x_1(t),x_2(t))x_3(t)\\ \dot{x}_2(t) &= b_{21}(x_1(t),x_2(t))x_3(t)\\ &\quad + f_2(x_2(t)) + b_{22}x_4(t)\\ \dot{x}_3(t) &= b_3(x_1(t),x_2(t))x_3(t)\\ \dot{x}_4(t) &= b_4(x_1(t))x_3(t)\end{aligned} \tag{3}$$

where

$$\begin{aligned}b_1(x_1(t),x_2(t)) &= -\left(\frac{\mu(x_1(t),x_2(t))}{y_{x/s}}+m_s\right)\\ f_2(x_2(t)) &= K_3Q_A\left(o_d^*-x_2(t)\right)\\ b_{21}(x_1(t),x_2(t)) &= -K_1(\mu(x_1(t),x_2(t))-k_e(t))-K_2\\ b_{22} &= -K_2\\ b_3(x_1(t),x_2(t)) &= \mu(x_1(t),x_2(t))-k_s(x_1(t))-k_e(t)\\ b_4(x_1(t)) &= k_s(x_1(t))\end{aligned} \tag{4}$$

and with the measurements

$$y(t) = \left[x_1(t-\tau)\ x_2(t-\tau)\right]^T \tag{5}$$

The block-wise form (3)-(5) allows a straightforward design of a second order sliding mode observer. The nominal parameters for the system (3) are given in Table 1.

Table 1. Nominal Parameters of the BT model.

Parameter	Values	Unit
$\mu_{\max}$	0.65	h^{-1}
$y_{x/s}$	0.37	$\mathrm{g}\cdot\mathrm{g}^{-1}$
K_s	3	$\mathrm{g}\cdot\mathrm{L}^{-1}$
K_o	1×10^{-4}	$\mathrm{g}\cdot\mathrm{L}^{-1}$
m_s	5×10^{-3}	$\mathrm{g}\cdot\mathrm{g}^{-1}\cdot\mathrm{h}^{-1}$
$k_{s,\max}$	0.5	h^{-1}
G_s	1	$\mathrm{g}\cdot\mathrm{L}^{-1}$
P_s	1	$\mathrm{g}\cdot\mathrm{L}^{-1}$
$k_{e,\max}$	0.1	h^{-1}
G_e	5	h
P_e	4.9	h
K_1	3.795×10^{-3}	dimensionless
K_2	0.729×10^{-3}	h^{-1}
K_3	2.114×10^{-3}	L^{-1}
Q_A	1320	$\mathrm{L}\cdot\mathrm{h}^{-1}$
o_d^*	0.00759	$\mathrm{g}\cdot\mathrm{L}^{-1}$
t_{ini}	0	h
$s_{p,\mathrm{ini}}$	32	$\mathrm{g}\cdot\mathrm{L}^{-1}$

3. PROPOSED SOFT SENSOR SCHEME

3.1 Observability Analysis

Let the vector $\mathcal{H}$ which contains the measured outputs of the system (3), $x_1(t-\tau)$, $x_2(t-\tau)$ and their derivatives be defined as

$$\mathcal{H} = \left[\,x_1(t-\tau)\ x_2(t-\tau)\ \dot{x}_1(t-\tau)\ \dot{x}_2(t-\tau)\,\right]^T \tag{6}$$

Similarly to the analysis presented in Sánchez et al. (2015), the observability analysis for the system (3) determines the existence of a diffeomorphism between the vector $\mathcal{H}$ and the delayed state vector $x = \left[\,x_1(t-\tau)\ x_2(t-\tau)\ x_3(t-\tau)\ x_4(t-\tau)\,\right]^T$.

The existence of this diffeomorphism can be evaluated, at least locally, by checking if the observability matrix defined as $\mathcal{O} = \frac{\partial\mathcal{H}}{\partial x(t-\tau)}$ is invertible. For the system (3), the observability matrix is calculated from (6) and is given by

$$\mathcal{O} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ * & * & b_1(x_1(t-\tau), x_2(t-\tau)) & 0 \\ * & * & b_{21}(x_1(t-\tau), x_2(t-\tau)) & b_{22} \end{bmatrix} \quad (7)$$

where it follows that the determinant of (7) is $\det(\mathcal{O}) = b_{22} b_1(x_1(t-\tau), x_2(t-\tau))$. Therefore, this system is observable for $t \geq \tau$. However, it can be shown that $|\det(\mathcal{O})|$ achieves a very small value (about 1×10^{-9}), which compromises the numerical invertibility of the observability matrix $\mathcal{O}$ (Sánchez et al., 2015).

To overcome this numerical drawback, the following scaling transformation of the state is proposed:

$$\begin{aligned} x_{1s}(t-\tau) &= \beta_1 x_1(t-\tau) \\ x_{2s}(t-\tau) &= \beta_2 x_2(t-\tau) \end{aligned} \quad (8)$$

with β_1 and β_2 real positive constants to be defined thereafter.

Thus, using the notation $x_i^\tau = x_i(t-\tau)$ for $i = 1, \ldots, 4$, the system (3) under the scaling (8) becomes:

$$\begin{aligned} \dot{x}_{1s}^\tau &= b_1^s(x_1^\tau, x_2^\tau) x_3^\tau \\ \dot{x}_{2s}^\tau &= f_2^s(x_2^\tau) + b_{21}^s(x_1^\tau, x_2^\tau) x_3 + b_{22}^s x_4^\tau \\ \dot{x}_3^\tau &= b_3(x_1^\tau, x_2^\tau) x_3^\tau \\ \dot{x}_4^\tau &= b_4(x_1^\tau) x_3^\tau \end{aligned} \quad (9)$$

where $b_1^s(x_1^\tau, x_2^\tau) = \beta_1 b_1(x_1^\tau, x_2^\tau)$, $f_2^s(x_2^\tau) = \beta_2 f_2(x_2^\tau)$, $b_{21}^s(x_1^\tau, x_2^\tau) = \beta_2 b_{21}(x_1^\tau, x_2^\tau)$ and $b_{22}^s = \beta_2 b_{22}$.

3.2 Observer-Predictor Scheme

In this section a cascade observer-predictor scheme is represented. Based in the structure presented in Khosravian et al. (2015b), the proposed scheme is composed for a SMO and Smith predictor. A block diagram of this proposal is shown in Figure 1. In this figure the sensor block separately block process is proposed to clarify, in this paper, the problem of delay occurs in the dynamics of the sensor.

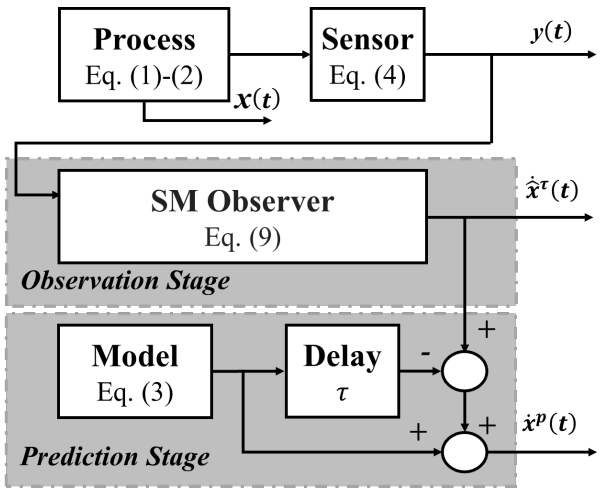


Figure 1. Observer-Predictor scheme

An explanation of the scheme of Figure 1 is as follows.

Observation Stage (SM Observer): First, from (8)-(9) the following Sliding Mode Observer is proposed in order to provide an estimation of the delayed state variables:

$$\begin{aligned} \hat{x}_1^\tau &= \beta_1^{-1} \hat{x}_{1s}^\tau \\ \hat{x}_2^\tau &= \beta_2^{-1} \hat{x}_{2s}^\tau \\ \dot{\hat{x}}_{1s}^\tau &= b_1^s(\hat{x}_1^\tau, \hat{x}_2^\tau) \hat{x}_3^\tau + k_{11} \phi_1(\tilde{x}_{1s}^\tau) \\ \dot{\hat{x}}_{2s}^\tau &= f_2^s(\hat{x}_2^\tau) + b_{21}^s(\hat{x}_1^\tau, \hat{x}_2^\tau) \hat{x}_3^\tau + b_{22}^s \hat{x}_4^\tau + k_{21} \phi_1(\tilde{x}_{2s}^\tau) \\ \dot{\hat{x}}_3^\tau &= b_3(\hat{x}_1^\tau, \hat{x}_2^\tau) \hat{x}_3^\tau + k_{12} \left[b_1^s(\hat{x}_1^\tau, \hat{x}_2^\tau)\right]^{-1} \phi_2(\tilde{x}_{1s}^\tau) \\ \dot{\hat{x}}_4^\tau &= b_4(\hat{x}_1^\tau) \hat{x}_3^\tau + k_{22} \left[b_{22}^s\right]^{-1} \phi_2(\tilde{x}_{2s}^\tau) \end{aligned} \quad (10)$$

where $\hat{x}_1^\tau$, $\hat{x}_2^\tau$, $\hat{x}_{1s}^\tau$, $\hat{x}_{2s}^\tau$, $\hat{x}_3^\tau$ and $\hat{x}_4^\tau$ are the estimates of x_1^τ, x_2^τ, x_{1s}^τ, x_{2s}^τ, x_3^τ and x_4^τ, respectively; $\tilde{x}_{1s}^\tau = x_{1s}^\tau - \hat{x}_{1s}^\tau$ and $\tilde{x}_{2s}^\tau = x_{2s}^\tau - \hat{x}_{2s}^\tau$ are the error variables; the observer input injections $\phi_1(\cdot)$ and $\phi_2(\cdot)$ are of the form $\phi_1(\cdot) = \lfloor \cdot \rceil^{\frac{1}{2}} + \theta \lfloor \cdot \rceil^{\frac{3}{2}}$ and $\phi_2(\cdot) = \frac{1}{2} \lfloor \cdot \rceil^0 + 2\theta \cdot + \frac{3}{2} \theta^2 \lfloor \cdot \rceil^2$, with the parameter $\theta \geq 0$, the function $\lfloor \cdot \rceil^\alpha = |\cdot|^\alpha \text{sign}(\cdot)$ is defined for $\alpha \geq 0$, where $\text{sign}(x) = 1$ for $x > 0$, $\text{sign}(x) = -1$ for $x < 0$ and $\text{sign}(0) \in \{-1, 1\}$; and $\lambda_1,\ \lambda_2 > 0$, and k_{11}, k_{12}, k_{21}, k_{22} are the observer positive gains.

The SMO (10) was proposed in a previous paper (Sánchez et al., 2015). This observer is fixed-time convergent and also has time-invariance property, according to the definition of Khosravian et al. (2015a). A detailed stability test of observer (10) without delay in measurements has been previously published (Sánchez et al., 2015). However, the problem considered in this paper is to estimate the current state $x(t)$ when the measurements of the output are delayed such that the output measurement at time t is $y(t) = h(x(t-\tau))$ for some know constant delay $\tau \geq 0$. In this sense a prediction stage it is proposed to offset the effect of the delay in the measurement.

Prediction Stage (Model + Delay): Second, based on (Khosravian et al., 2015a) a Smith predictor compensating the delay may be considered as

$$\dot{x}^p(t) = \dot{\hat{x}}^\tau(t) + f(x^p(t)) - f(x^p(t-\tau)) \quad (11)$$

where the prediction of the current state is denoted by $x^p \in \mathbb{R}^n$ and $\hat{x}^\tau$ is the estimate x subject to delayed output measurements (5). Moreover, with the system model (3) and the known delay τ for output measurement (5), it is possible to know the dynamics of the predicted states without delay $f(x^p(t))$ and delayed $f(x^p(t-\tau))$.

The stability of the Observer-Predictor structure is such that the estimate state converge asymptotically/exponentially to the system trajectories (1)-(2), if the estimates provided by the Observer (SMO) converge asymptotically/exponentially to the delayed system state (Khosravian et al., 2015a). In this sense the definition of finite time convergent include asymptotically/exponentially convergent and fixed-time convergent of SMO (10) is a stronger form of finite time (Polyakov, 2012). In the next section the simulation results are presented.

4. SIMULATION RESULTS

This section presents the numerical simulation results of the proposed estimation structure. The simulations parameters were:

- Fundamental step size of $1 \times 10^{-5}[h]$. This time is small due to requirement of robust differentiation in the estimation scheme.
- Model parameters like shown on Table 1.

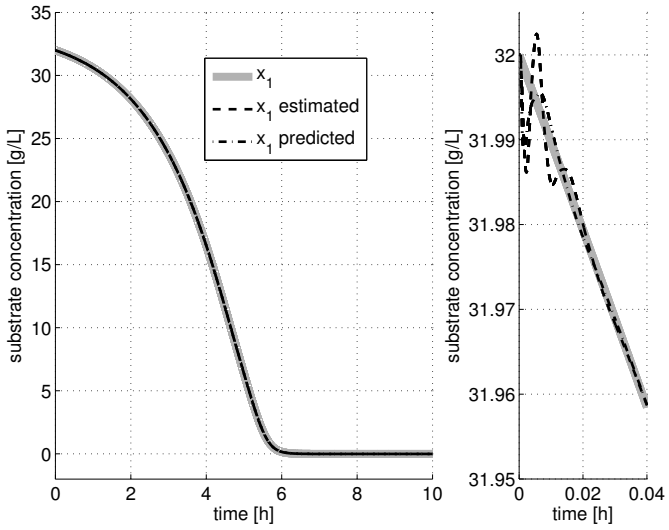


Figure 2. Substrate concentration s_p with $\tau = 5 \times 10^{-5}[h]$.

- The parameters shown in this table were taken according to the range to $20\,[\mathrm{g}\cdot\mathrm{L}^{-1}] < s_{p,\max} < 32\,[\mathrm{g}\cdot\mathrm{L}^{-1}]$.
- The value $s_{p,\max}$ corresponds to the initial condition of s_p since $\dot{s}_p \leq 0$.
- The substrate concentration $s_p = x_1$ and the dissolved oxygen concentration $o_d = x_2$ are assumed to be measured, noiseless and delayed, and the initial conditions $\hat{x}^\tau_{1s}$ and $\hat{x}^\tau_{2s}$ were taken as the scaled initial conditions of x_1 and x_2 respectively
- The delay is a known constant $\tau \geq 0$. However, since the vegetative cells concentration $x_v = x_3$ and the sporulated cells concentration $x_s = x_4$ aren't measured, the initial conditions $\hat{x}^\tau_3$ and $\hat{x}^\tau_4$ were taken different from x_3 and x_4, respectively.
- Another thing that should be noted is that with the selected values of β_1 and β_2, the minimum value of $|\det(\mathcal{O}_s)|$ is around 2.

Figures 2, 3, 4 and 5 show the comparison between the actual x, estimated $\hat{x}^\tau$ (SMO without prediction) and predicted x^p (SMO with prediction) variables corresponding to substrate concentration s_p, dissolved oxygen concentration o_d, vegetative cell concentration x_v and sporulated cells concentration x_s when the delay measurement is $\tau = 5 \times 10^{-5}[h]$. It can be noticed that, despite initial estimation error $x(0) = \left[32,\ 0.74 \times 10^{-2},\ 0.645,\ 1 \times 10{-}5\right]^T$, and $\hat{x}^\tau(0) = x^p(0) = \left[32,\ 0.74 \times 10^{-2},\ 6.45,\ 1\right]^T$ the fixed time convergence of the estimated variables is achieved.

Figures 6 and 7 show the comparison between the actual and estimated variables corresponding to x_v and x_s when measurements of s_p and o_d are delayed with $\tau = 1 \times 10^{-1}[h]$ with SMO (SMO without prediction) and predicted x^p (SMO with prediction). Based on the presented results, it can be observed a good performance of the observer-predictor scheme proposed while the only SMO does not converge. A correct and fast estimation of x_v and x_s using the cascade observer-predictor is achieved making the proposed system suitable for observer-based control applications.

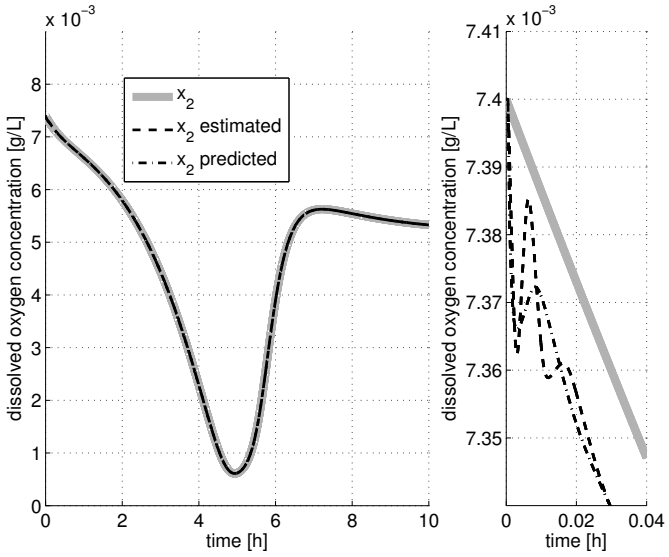


Figure 3. Dissolved oxygen concentration o_d with $\tau = 5 \times 10^{-5}[h]$.

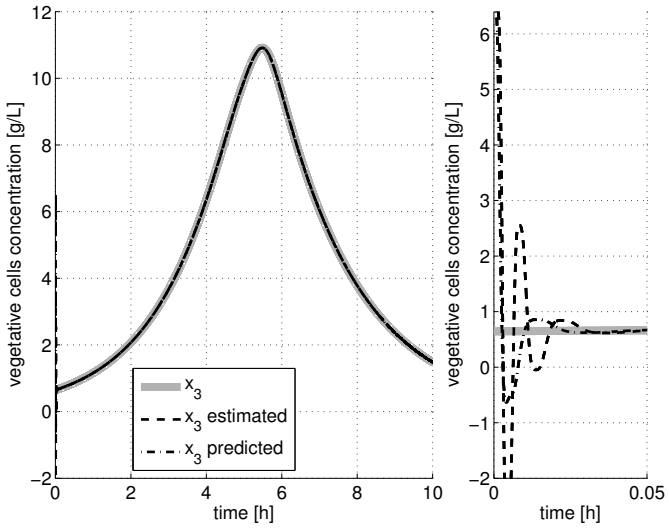


Figure 4. Vegetative cells concentration x_v with $\tau = 5 \times 10^{-5}[h]$.

Finally, Figures 8 and 9 show the effect of increased the delay measurement in s_p and o_d for estimation of x_v and x_s respectively in booth cases only SMO and SMO-predictor. The Integral Time Absolute Error (ITAE) of only SMO tends to infinity for delays in measuring higher than $\tau = 6 \times 10^{-5}[h]$, while the cascade observer-predictor scheme keep the convergence of error when the delay increase.

5. CONCLUSIONS

In this paper was presented a soft sensor to estimate the biomass in a batch bioprocess subject to delayed measurements. The soft sensor proposed is based on a cascade sliding mode observer-predictor. The observer stage is based on a class of second order sliding mode algorithms, allowing a fixed-time estimation of the biomass. The prediction stage offsets the effect of the delay in measurements. Convergence proof and numerical

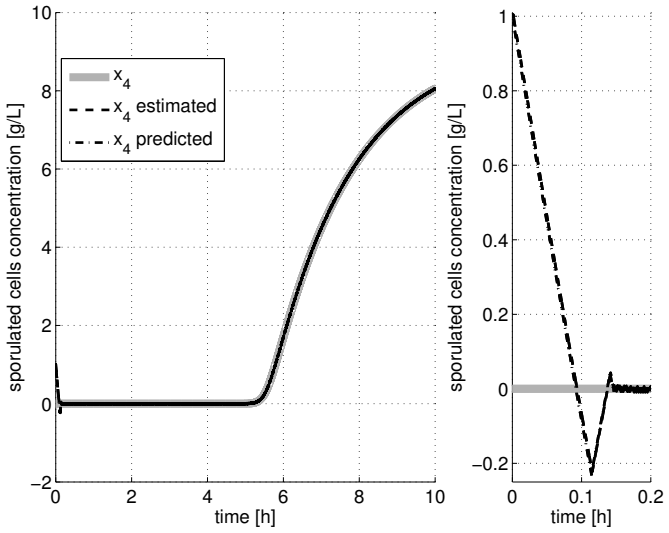


Figure 5. Sporulated cells concentration x_s with $\tau = 5 \times 10^{-5}[h]$.

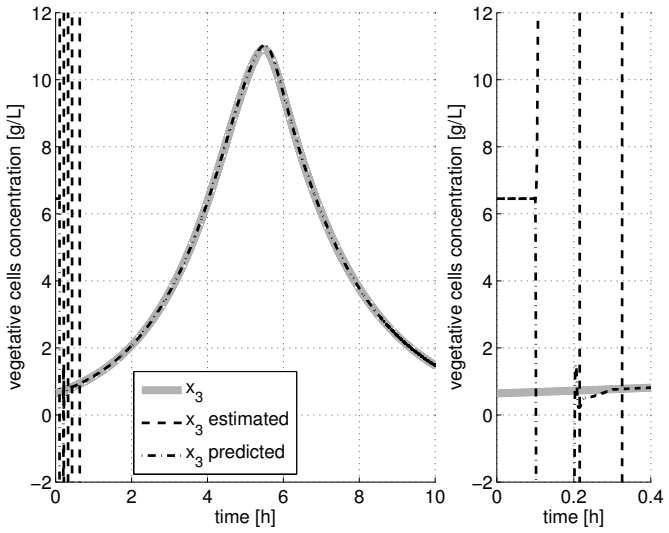


Figure 6. Vegetative cells concentration x_v with $\tau = 0.1[h]$.

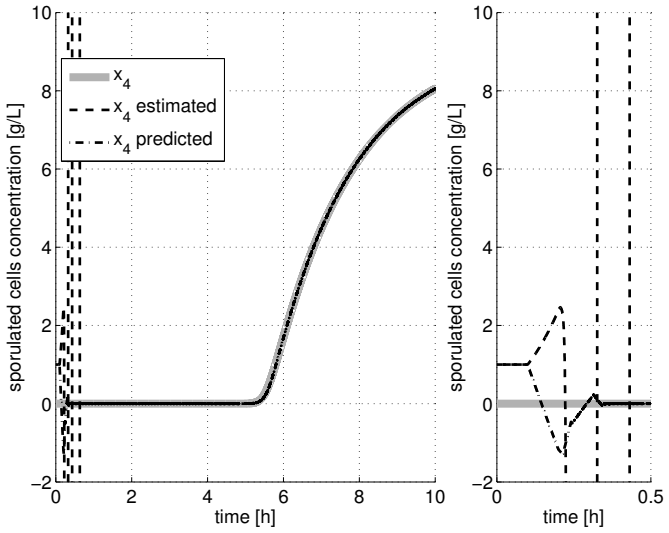


Figure 7. Sporulated cells concentration x_s with $\tau = 0.1[h]$.

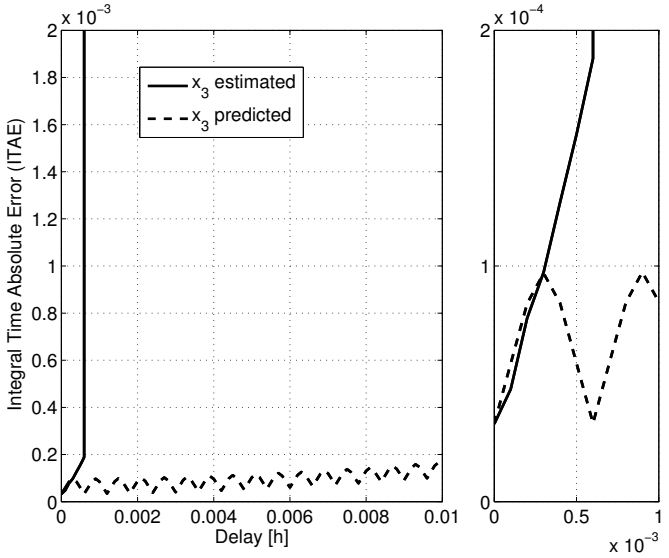


Figure 8. Delay effect to estimate x_v.

simulations shown the feasibility of the cascade observer-predictor proposed.

REFERENCES

Amicarelli, A., di Sciascio, F., Toibero, J.M., and Alvarez, H. (2010). Including Dissolved Oxygen Dynamics into the Bt delta-Endotoxins Production Process Model and its Application to Process Control. *Brazilian Journal of Chemical Engineering*, 27(1), 41–62.

Bequette, B. (2002). Behavior of a CSTR with a recirculating jacket heat transfer system. In *Proceedings of the American Control Conference.*, volume 4, 3275 – 3280.

Dochain, D. (2003). State and parameter estimation in chemical and biochemical processes: a tutorial. *Journal of Process Control*, 13(8), 801 – 818.

Gopalakrishnan, A., Kaisare, N.S., and Narasimhan, S. (2011). Incorporating delayed and infrequent measurements in Extended Kalman Filter based nonlinear state estimation. *Journal of Process Control*, 21(1), 119–129.

Guo, Y. and Huang, B. (2015). State estimation incorporating infrequent, delayed and integral measurements. *Automatica*, 58, 32–38. doi: 10.1016/j.automatica.2015.05.001.

Guo, Y., Zhao, Y., and Huang, B. (2014). Development of soft sensor by incorporating the delayed infrequent and irregular measurements. *Journal of Process Control*, 24(11), 1733–1739.

Khosravian, A., Trumpf, J., and Mahony, R. (2015a). State Estimation for Nonlinear Systems with Delayed Output Measurements. (Cdc), 6330–6335. doi: 10.1016/j.automatica.2016.01.024.

Khosravian, A., Trumpf, J., Mahony, R., and Hamel, T. (2015b). State Estimation for Invariant Systems on Lie Groups with Delayed Output. *Automatica*, 68, 254–265.

Polyakov, A. (2012). Nonlinear feedback design for fixed-time stabilization of linear control systems. *IEEE Transactions on Automatic Control*, 57(8), 2106–2110.

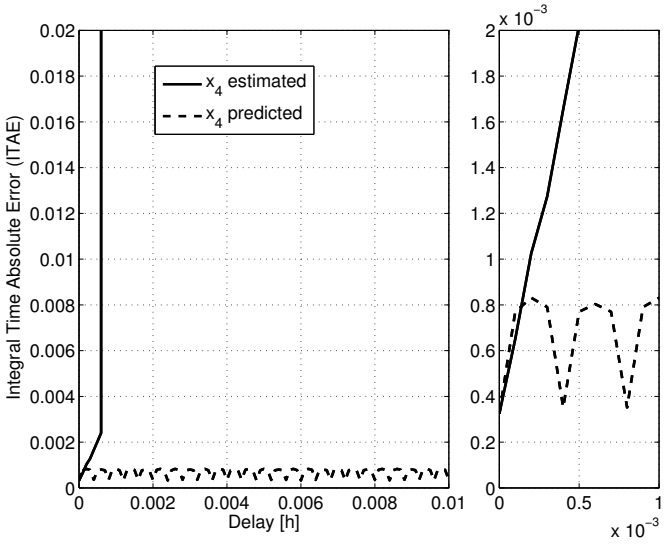


Figure 9. Delay effect to estimate x_s.

Rómoli, S., Amicarelli, A.N., Ortiz, O.A., Scaglia, G.J.E., and di Sciascio, F. (2016). Nonlinear control of the dissolved oxygen concentration integrated with a biomass estimator for production of Bacillus thuringiensis δ-endotoxins. *Computers & Chemical Engineering*, 93, 13–24. doi:10.1016/j.compchemeng.2016.05.017.

Sánchez, J.D., Isaza, J.A., Jaramillo, O., Jimenez, E., and Botero, H. (2015). A Fixed-Time Convergent Observer for Biomass in a Batch Process. In *IEEE CCAC 2015 Conference Proceedings*, volume 1, 1–4.

Walcott, B.L., Corless, M.J., and Zak, S.H. (1987). Comparative study of nonlinear state observation techniques. *Int. J. Control*, 45, 2109–2132.

Wang, H.P., Tian, Y., and Vasseur, C. (2015). Piecewise continuous hybrid systems based observer design for linear systems with variable sampling periods and delay output. *Signal Processing*, 114, 75–84. doi: 10.1016/j.sigpro.2015.01.009.

Zeng, J. and Liu, J. (2015). Distributed moving horizon state estimation: Simultaneously handling communication delays and data losses. *Systems & Control Letters*, 75, 56–68. doi:10.1016/j.sysconle.2014.11.007.

Zhao, L., Wang, J., Yu, T., Chen, K., and Liu, T. (2015). Nonlinear state estimation for fermentation process using cubature Kalman filter to incorporate delayed measurements. *Chinese Journal of Chemical Engineering*, 23(11), 1801–1810. doi:10.1016/j.cjche.2015.09.005.

Adaptive Trajectory Tracking Control of a Boiler-Turbine Adopting an Algebra Approach

Sebastián Godoy Bordes, Mario E. Serrano, Santiago Rómoli, Gustavo Scaglia.

Instituto de Ingeniería Química, Universidad Nacional de San Juan, CONICET, Argentina.
(e-mail: sgodoy@unsj.edu.ar, serranoemanuel84@gmail.com, sromoli@unsj.edu.ar, gscaglia@unsj.edu.ar).

Abstract: This paper presents the development of a novel control law with parameter updating for trajectory tracking in a boiler-turbine system, which is a MIMO system where all state variables are strongly coupled. The proposed controller is simple and based on comprehensible concepts. Its design consists in representing the mathematical model using numerical methods, and then the control actions are computed through a linear algebra technique to accomplish the target of trajectory tracking control, by last, the parameter updating is based in the Lyapunov theory. The advantages of this method are the condition for the tracking error tends to zero, and the calculation of control actions, are obtained simply by solving a system of linear equations. Besides, the control technique has the ability to follow trajectories for two outputs simultaneously.

Keywords: Control System Design, Linear Algebra, Adaptive Control, Nonlinear Model, Tracking Trajectory Control.

1. INTRODUCTION

A boiler-turbine system is an energy conversion device that consists of a steam boiler and a turbine. The steam boiler part generates the thermal energy that is transferred into the turbine part. The boiler-turbine systems are able to supply electricity faster than other traditional systems, due to this systems can remain in operation, despite being disconnected of the electric grid. Then the system can be connected immediately when the grid requires it. The main control requirement of a boiler-turbine is about meeting the load demand quickly maintaining internal variables such as temperature and drum water level within the desired range to prevent the overheating or flooding in the system. In addition, the drum pressure must be controlled in order to remain bounded over a range of values according to variable operating conditions and the load demand. Also, the inputs and outputs constraints are imposed by physical limits such as the magnitude and saturation of the internal control valves. When these restrictions are added, the preliminary system becomes a multi-input multi-output (MIMO) nonlinear system (Bell et al., 1987), with state variables strongly coupled. Tracking control becomes a bigger challenge when the parametric model uncertainties that affect substantially the performance of the control system are considered. In this paper a simple control solution that addresses all the aspects mentioned above is proposed.

In general, the control structure used in the boiler-turbine must achieve the following challenges in order to be successfully applied.

1) The design method must be applied to a family of boiler turbine units. In general for each unit must be found the identification of the model and then the design of the controller. Each boiler-turbine unit has a specific model but in general, all of them show similar dynamics. For this reason, a generic control structure can be used for a general class of boiler-turbine units.

2) The control algorithms must be easy to implement and maintain. The application of advanced control methods generates, commonly, controllers that are complex, therefore it is necessary find techniques that guarantees a good performance with a simple control structure. First and second items are satisfied through techniques such as tuning method (Tan et al., 2004).

3) The performance for a wide-range load variation should be guaranteed for a single controller. In general, a variation in the load demand generates different operation points with different linearized models. Therefore, a single controller that ensure the performance in all operation points is a challenging task. Techniques such as gain schooled (Chen et al., 2004) and multi-model control (Tan et al., 2004) are used to achieve this objective.

The control process becomes even more interesting due to the severe nonlinearity in many variables over a wide operation range, tight operating constraints and the fact that the boiler-turbine plant take part, more frequently, in the grid power regulation. Furthermore the impossibility of having a accurate model without uncertainty difficult the process control (Toodeshki et al., 2008). Considering all the reasons mentioned above, the task of designing an advanced

controller for the boiler-turbine, that should be simple and easy to implement, is a challenging assignment.

The control of boiler-turbine has attracted the attention of many researches in the last years. Different control strategies have been proposed in the literature, where some of them use linearization techniques (Chen et al., 2004; Tan et al., 2005; Zheng et al., 2006; Tan et al., 2008; Sarailoo et al., 2012). Also in (Dimeo et al., 1995; Fang et al., 2004; Wu et al., 2009; Wu et al., 2010) different control methods based on Hinf are applied. In (Fang et al., 2004) the loop-shaping H\inf method is used to design the feedback controller and the final controller is reduced to a multivariable PI form. In (Wu et al., 2009; Wu et al., 2010), the nonlinear boiler-turbine dynamics were represented by the Takagi and Sugeno fuzzy model. Then, a fuzzy Hinf state feedback control law was synthesized in terms of linear matrix inequalities (LMIs). In (Thangavelusamy et al., 2013) the authors propose a Fuzzy logic based Sliding Mode Control (FSMC) to a drum-type boiler–turbine system with a proportional integral controller. The drawback of these methods lies in the need of linearizing the model for a certain range of operation.

Several authors suggest the combined application of different control techniques such as genetic algorithm (GA), fuzzy control, gain scheduling, sliding mode, for the control of boiler-turbine. In (Wu et al., 2012) the authors uses several techniques jointly, first, a nonlinear predictive control based on an online Recording Horizon Control (RHC) algorithm with genetic algorithm is designed. Second, an Hinf fuzzy controller is implemented. Finally a switching control strategy is applied to choose between the two described controllers. In (Wu et al., 2013) is presented a data-driven fuzzy modelling strategy and predictive controller for boiler-turbine. In this work the behavoir of the boiler-turbine is dived into a number of local regions taking into account the measurement data, then a multimodel-based model predictive control (MMPC) is developed. Finally a Data-driven Direct Predictive Controller (DDPC) is designed with an online update of the predictor. In (Ghabraei et al, 2015) a robust adaptive sliding mode controller (RASMC) for a multivariable nonlinear model of boiler– turbine is presented. First the authors develop an input–output feedback linearization with the purpose of overcome the coupled nonlinearities. Then a new decoupled inputs model is generated. Finally suitable sliding surface for the RASMC is considered. In these works, the complexity of the controller is incremented by the number of stages included in the control structure. Furthermore, in general, an approximation or transformation of the nonlinear system is required. In addition, most of the mentioned works focused on boiler–turbine dynamics without modeling imprecisions.

This work provides a simple solution to the above challenging problem. A new control approach originally developed for robotic systems (Scaglia et al., 2009) is applied successfully in trajectory tracking of a boiler-turbine in presence of parametric uncertainties. The technique uses numerical methods and linear algebra to solve the problem. This structure has been used successfully in different nonlinear multivariable models, (Rómoli et al., 2014, Serrano et al., 2014, Cheein et al.,2016). This simple approach suggests that knowing the value of the desired state, a value for the control action which forces the system to move from its current state to the desired one may be found. The theoretical fundamentals are based on easily understandable concepts and there is no need of complex calculations to attain the control signal. The adaptive scheme is obtained analyzing the system stability and the asymptotic stability of the complete controllers is proven through Lyapunov theory.

The paper is organized as follows: in Section 2, the dynamic model of a boiler-turbine is presented. The methodology of the controller design is shown in Section 3. The development of the parameter updating is detailed in Section 4. In Section 5, the theoretical results are validated with simulation results of the control algorithm. Finally, Section 6 presents the conclusions.

2. DYNAMIC MODEL OF THE MOBILE ROBOT

A 160MW oil-fired electrical power plant model is used in this paper constituted with a drum type boiler and a turbine. The model is based on the P16/G16 at the Sydvenska Kraft AB plant in Malmö, Sweden (Bell et al., 1987),. This model is widely used in the specialized literature (Tan et al., 2008), (Tan et al., 2005), (Fang et al., 2004; Sarailoo et al., 2012), . The boiler dynamic model is provided by both physical and empirical methods based on data acquired from a series of experiments and identifications which capture all the relevant characteristics of the process. The nonlinear boiler-turbine dynamic model is given by the following equations:

$$\begin{cases} \dot{x}_1 = -0.0018u_2x_1^{9/8} + 0.9u_1 - 0.15u_3 \\ \dot{x}_2 = (0.073u_2 - 0.016)x_1^{9/8} - 0.1x_2 \\ \dot{x}_3 = (141u_3 - (1.1u_2 - 0.19)x_1)/85 \end{cases} \quad (1)$$

$$\begin{cases} y_1 = x_1 \\ y_2 = x_2 \\ y_3 = 0.05(0.13073x_3 + 100\alpha + q_e/9 - 67.975) \end{cases} \quad (2)$$

In (1) and (2), the state variables x_1, x_2, and x_3 denote drum pressure (kg/cm^2), electric output (MW), and fluid density (kg/m^3), respectively. The inputs u_1, u_2, and u_3 are the valve positions for fuel flow, steam control, and feedwater flow, respectively. The output y_3 is the drum water level (m), and α and q_e are steam quality and evaporation rate (kg/s), respectively and are given by:

$$\alpha = \frac{(1 - 0.001538x_3)(0.8x_1 - 25.6)}{x_3(1.0394 - 0.0012304x_1)} \quad (3)$$

$$q_e = (0.854u_2 - 0.147)x_1 + 45.59u_1 - 2.514u_3 - 2.096$$

The system presents the following physical restrictions in the control actions:

$$
\begin{aligned}
&-0.007/sec \le \frac{du_1}{dt} \le 0.007/sec \\
&-1/sec \le \frac{du_2}{dt} \le 0.1/sec \\
&-0.05/sec \le \frac{du_3}{dt} \le 0.05/sec \\
&0 \le u_i \le 1 \qquad \forall i \in (1,2,3)
\end{aligned}
\tag{4}
$$

The Figure 1 shows a simplified scheme of the plant used. The boiler part has as inputs the control action u_1, u_3 and the output y_1, y_3, them the steam generated is directed to the Turbine part by means of u_2 generating the required electricity output y_2

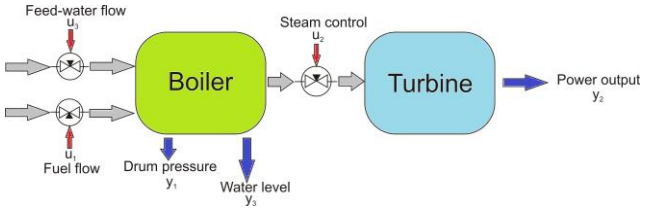


Fig.1: Simplified scheme of the plant used

Some typical operating points of the boiler-turbine model (1), are tabulated in Table I. The literature, in works such as (Bell et al., 1987; Fang et al., 2004; Tan et al., 2004) usually takes the operating point #4 as the nominal point. For more details about the points of operation see (Bell et al., 1987; Fang et al., 2004) .

TABLE I
Typical operating points of Bell and Astrom model

	#1	#2	#3	#4	#5	#6	#7
x_1^0	75.60	86.40	97.20	108	118.8	129.6	140.4
x_2^0	15.27	36.65	50.52	66.65	85.06	105.8	128.9
x_3^0	299.60	342.40	385.20	428	470.8	513.6	556.4
u_1^0	0.156	0.209	0.271	0.34	0.418	0.505	0.6
u_2^0	0.483	0.552	0.621	0.69	0.759	0.828	0.897
u_3^0	0.183	0.256	0.340	0.433	0.543	0.663	0.793
y_3^0	-0.97	-0.65	-0.32	0	0.32	0.64	0.98

3. CONTROL DESIGN

The basic objective of any design for boiler-turbine control system is to make the electrical output y_2 follows the load demand rapidly while the water level y_3 and drum steam pressure y_1 within the allowed limits are maintained. Besides fulfilling the previous objective, the proposed controller is capable of following the allowable trajectory for the outputs y_1 and y_2, holding y_3 constant within certain limits/bounds.

Remark 1: Allowable path is that one which may be physically realizable, considering the dynamics and constraints of the plant.

Remark 2: It is considered that the values of the state variable are available in every time

In this section, the control law capable of generating the signals $u_1(t)$, $u_2(t)$, $u_3(t)$, with the objective that the outputs $y_1(t)$, $y_2(t)$, $y_3(t)$, follow the desired trajectories, previously determined, $y_{1d}(t)$, $y_{2d}(t)$, $y_{3d}(t)$, is designed. Where $y_{1d}(t)$ and $y_{2d}(t)$ will be variable trajectories while $y_{3d}(t)$ will be a maintained constant. These requirements result in order to get a stipulated output power, varying as desired the pressure generated in the boiler and maintaining the water level in the boiler within a safe level, avoiding in this way that overheating occurs (Wu et al., 2010; Chen, 2013).

According to (2) the output $y_1(t)$, $y_2(t)$ are manipulated by x_1, x_2 respectively, therefore modifying the states x_1, x_2, the desired output $y_{1d}(t)$, $y_{2d}(t)$ are achieved. Then, the state variable x_3 is formulated to attain the desired value of the $y_{3d}(t)$. Once all the state variables are available, in order to achieve the tracking control objective, it is necessary calculate the control actions which lead the system to the desired trajectories, by solving a system of linear equations.

The first step of the proposed methodology is to write (1) in a matrix form:

$$
\begin{bmatrix} \theta_1 & -\theta_2 x_1^{9/8} & -\theta_3 \\ 0 & \theta_4 x_1^{9/8} & 0 \\ 0 & -\theta_5 x_1 & \theta_6 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} \ldots
\tag{5}
$$

$$
\ldots + \begin{bmatrix} 0 & 0 & 0 \\ \theta_7 x_1^{1/8} & \theta_8 & 0 \\ -\theta_9 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}
$$

Where the parameters θ are the same ones that (1)

$$
\theta_1 = 0.9 \quad \theta_2 = 0.0018 \quad \theta_3 = 0.15 \quad \theta_4 = 0.073 \quad \theta_5 = 0.012
$$
$$
\theta_6 = 1.658 \quad \theta_7 = 0.016 \quad \theta_8 = 0.1 \quad \theta_9 = 0.0022
$$

Writing (5) in symbolic form, we have:

$$
\mathbf{Au} = \dot{\mathbf{x}} + \mathbf{Bx} \tag{6}
$$

Where (6) represent the real dynamic model of the boiler-turbine. It will use to find the expressions of control actions for tracking trajectories.

*Remark 3:*The real parameters of the system boiler-turbine are $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_9]^T$, while the estimated parameters used by

the proposed controller will be $\hat{\boldsymbol{\theta}} = \left[\hat{\theta}_1, \ldots, \hat{\theta}_9\right]^T$. Finally the error in the parameters election will be $\tilde{\boldsymbol{\theta}} = (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) = \left[\tilde{\theta}_1, \ldots, \tilde{\theta}_9\right]^T$.

Remark 4: When accurate knowledge of the model is mentioned, this means that the plant parameters are known exactly $(\boldsymbol{\theta} = \hat{\boldsymbol{\theta}})$, otherwise, parametric uncertainty is considered $\tilde{\boldsymbol{\theta}} = (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})$.

Considering exact knowledge of the model and based on the inverse dynamic, it is proposed the following control law which considers (6) as a purely mathematical system

$$\mathbf{A}\mathbf{u} = \mathbf{b} \tag{7}$$

Where

$$\mathbf{b} = \boldsymbol{\sigma} + \mathbf{B}\mathbf{x}$$

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \end{bmatrix} = \begin{bmatrix} \dot{x}_{1d} + k_1 e_{x_1} \\ \dot{x}_{2d} + k_2 e_{x_2} \\ \dot{x}_{3ez} + k_3 e_{x_3} \end{bmatrix}$$

The variables x_{1d} y x_{2d} represent the desired values of the states to be achieved, x_{3ez} represents the required value that must have x_3 to attain the desired value of the y_3. The expression of the x_{3ez} will be found in the next paragraph. The constants (k_1,k_2,k_3 > *0*) represent the parameter of the controller. Finally e_{x1}, e_{x2}, e_{x3} represent the individual tracking errors.

Remark 5: The value of the difference between the desired and real trajectory will be called tracking error. It is given by $e_{x_1} = x_{1d} - x_1, e_{x_2} = x_{2d} - x_2, e_{x_3} = x_{3ez} - x_3$. The tracking error is represented by $\|e_n\| = \sqrt{\left(e_{x_1}{}^2 + e_{x_2}{}^2 + e_{x_3}{}^2\right)}$.

The value of x_{3ez} is found using the latest expression of (2). It aims to force y_3 for achieving the desired output value y_{3d} by varying x_{3ez}

$$y_3 = 0.05(0.13073x_3 + 100\alpha_{(x_1 \cdot x_3)} + q_{e(x_1 \cdot u)} / 9 - 67.975)$$

$$y_{3d} = 0.05\left(0.13073x_{3ez}\right) + \left(\frac{(1 - 0.001538x_{3ez})(0.8x_1 - 25.6)}{x_{3ez}(1.0394 - 0.0012304x_1)}\right) + \cdots$$

$$\cdots + 0.05\left(\frac{(0.854u_2 - 0.147)x_1}{9} + \frac{45.59u_1}{9} - \frac{2.514u_3}{9} - \frac{2.096}{9} - 67.975\right)$$

$$y_3 \to y_{3d} \quad when \quad x_3 \to x_{3ez}$$

. (8)

Where the values y_{3d}, x_1, u_1, u_2, u_3 are known. The value of x_{3ez} is obtained using algorithms that find minimum of unconstrained multivariable function using derivative-free method (Brent, 2013).

Remark 6: When the algorithm is implemented in t=0 the values $u_1(0)$, $u_2(0)$, $u_3(0)$ and $x_{3ez}(0)=x_3(0)$ are obtained depending on the chosen operating point according to Table 1.

The system (7) can be considered as a system of linear equations, where $\mathbf{u}$ represents the unknown matrix, $\mathbf{A}$ the coefficient matrix, and $\mathbf{b} = \boldsymbol{\sigma} + \mathbf{B}\mathbf{x}$ the matrix of independent term. The solution of the system (7) is resolved through least square theory (Strang, 2006). Analyzing $\mathbf{A}$ is noted that it is always linearly independent, because x_1 is never zero in normal operation, therefore, the inverse of the matrix $\mathbf{A}$ exist.

$$\mathbf{u} = \mathbf{A}^{-1}\boldsymbol{\sigma} + \mathbf{A}^{-1}\mathbf{B}\mathbf{x} \tag{9}$$

Using the process of Gaussian elimination (Strang, 2006) the analytic expressions of the control actions are found

$$u_1 = \frac{1}{\theta_1}\sigma_1 + \frac{\theta_3}{\theta_1\theta_6}\left(\sigma_3 - \theta_9 x_1\right) + \ldots$$
$$\ldots\left(\frac{\theta_2}{\theta_1\theta_4} + \frac{\theta_3\theta_5}{\theta_1\theta_4\theta_6}x_1^{-1/8}\right)\left(\sigma_2 + \theta_7 x_1^{9/8} + \theta_8 x_2\right)$$

$$u_2 = \frac{1}{\theta_4}\left(\sigma_2 + \theta_7 x_1^{9/8} + \theta_8 x_2\right)x_1^{-9/8} \tag{10}$$

$$u_3 = \frac{1}{\theta_6}\left(\sigma_3 - \theta_9 x_1\right) + \frac{\theta_5}{\theta_4\theta_6}x_1^{-1/8}\left(\sigma_2 + \theta_7 x_1^{9/8} + \theta_8 x_2\right)$$

The proposed controller design is completely finished by using the relationships given in (8), which will be used to generate the control actions in (10).

Theorem 1: If the system behaviour is ruled by (6) and the controller is designed by (10), then $e_n \to 0$ with $t \to \infty$ when trajectory tracking problems are considered. The proof of Theorem 1 and the convergence to zero of tracking errors can be seen in Appendix A.

4. ADAPTIVE LAW

In this section the existence of parametric uncertainty is considered, where the estimated parameters used by the controller are far from the real values of the plant. This uncertainty, when is not taken into consideration, affects the performance of the developed controller. To compensate this effect a methodology is proposed to adapt the internal

parameters of the controller and compensate the influence of the parametric uncertainly in the trajectory tracking.

When parametric uncertainty is considered, the control action designed in (9) is modified using the estimated parameters of the model instead of real parameters

$$\mathbf{u} = \hat{\mathbf{A}}^{-1}\boldsymbol{\sigma} + \hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x} \tag{11}$$

Where

$$\hat{\mathbf{A}}^{-1} = \begin{bmatrix} \frac{1}{\hat{\theta}_1} & \left(\frac{\hat{\theta}_2}{\hat{\theta}_1\hat{\theta}_4} + \frac{\hat{\theta}_3\hat{\theta}_5}{\hat{\theta}_1\hat{\theta}_4\hat{\theta}_6}x_1^{-1/8}\right) & \frac{\hat{\theta}_3}{\hat{\theta}_1\hat{\theta}_6} \\ 0 & \frac{1}{\hat{\theta}_4}x_1^{-9/8} & 0 \\ 0 & \frac{\hat{\theta}_5}{\hat{\theta}_4\hat{\theta}_6}x_1^{-1/8} & \frac{1}{\hat{\theta}_6} \end{bmatrix}$$

$$\hat{\mathbf{B}} = \begin{bmatrix} 0 & 0 & 0 \\ \hat{\theta}_7 x_1^{1/8} & \hat{\theta}_8 & 0 \\ -\hat{\theta}_9 & 0 & 0 \end{bmatrix}$$

Besides, the real boiler-turbine model (6), can be descomposed considering the *Remark 3* as follows:

$$\left(\hat{\mathbf{A}} - \tilde{\mathbf{A}}\right)\mathbf{u} = \dot{\mathbf{x}} + \left(\hat{\mathbf{B}} - \tilde{\mathbf{B}}\right)\mathbf{x} \tag{12}$$

Where

$$\tilde{\mathbf{A}} = \begin{bmatrix} \tilde{\theta}_1 & -\tilde{\theta}_2 x_1^{9/8} & -\tilde{\theta}_3 \\ 0 & \tilde{\theta}_4 x_1^{9/8} & 0 \\ 0 & -\tilde{\theta}_5 x_1 & \tilde{\theta}_6 \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} 0 & 0 & 0 \\ \tilde{\theta}_7 x_1^{1/8} & \tilde{\theta}_8 & 0 \\ -\tilde{\theta}_9 & 0 & 0 \end{bmatrix}$$

The error equation of the control system when there is parametric uncertainty is obtained replacing (11) in (12).

$$\left(\hat{\mathbf{A}} - \tilde{\mathbf{A}}\right)\left(\hat{\mathbf{A}}^{-1}\boldsymbol{\sigma} + \hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x}\right) = \dot{\mathbf{x}} + \left(\hat{\mathbf{B}} - \tilde{\mathbf{B}}\right)\mathbf{x}$$

which is equivalent to

$$\boldsymbol{\sigma} - \dot{\mathbf{x}} = \tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\boldsymbol{\sigma} + \tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x} - \tilde{\mathbf{B}}\mathbf{x}$$

Analyzing the left side of the equation and considering (20) and (21) in the proof of Theorem 1 we have:

$$\dot{\mathbf{e}}_x + \mathbf{K}\mathbf{e}_x = \tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\boldsymbol{\sigma} + \tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x} - \tilde{\mathbf{B}}\mathbf{x} \tag{13}$$

Analyzing the right side of the equation, we see that all terms are pre-multiplied by matrices that contain terms which depend on the errors in the parameters. Therefore each term can be written as a function of $\tilde{\boldsymbol{\theta}}$ as follows:

$$\tilde{\mathbf{B}}\mathbf{x} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x_1^{9/8} & x_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -x_1 \end{bmatrix}\tilde{\boldsymbol{\theta}}$$

$$\tilde{\mathbf{B}}\mathbf{x} = \mathbf{G}_1\tilde{\boldsymbol{\theta}}$$

Where $\mathbf{G_1}$ is a 3x9 matrix. In the same way the remaining terms are rewritten

$$\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\boldsymbol{\sigma} = \mathbf{G}_2\tilde{\boldsymbol{\theta}}$$
$$\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x} = \mathbf{G}_3\tilde{\boldsymbol{\theta}}$$

Where $\mathbf{G_2}$ and $\mathbf{G_3}$ are a 3x9 matrix. Then (13) can be replaced by

$$\dot{\mathbf{e}}_x + \mathbf{K}\mathbf{e}_x = \mathbf{G}_2\tilde{\boldsymbol{\theta}} + \mathbf{G}_3\tilde{\boldsymbol{\theta}} - \mathbf{G}_1\tilde{\boldsymbol{\theta}}$$

$$\dot{\mathbf{e}}_x = -\mathbf{K}\mathbf{e}_x + \mathbf{G}\tilde{\boldsymbol{\theta}} \tag{14}$$

Where $\mathbf{G=G_2+G_3-G_1}$. The expression (14) represents the error equation of the control system when exist parametric uncertainties, where the term $\mathbf{G}\tilde{\boldsymbol{\theta}}$ represents the influence of the parametric uncertainties in the tracking errors. The full development of the matrices $\mathbf{G_1}$, $\mathbf{G_2}$, $\mathbf{G_3}$ and $\mathbf{G}$ are detailed in Appendix B.

Analyzing (14) and considering the following Lyapunov candidate function (Khalil et al., 1996)

$$V = \mathbf{e}_x^{\mathrm{T}}\mathbf{e}_x + \tilde{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\theta} \tag{15}$$

Such that $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$ his time derivate can be considered as $\dot{\tilde{\boldsymbol{\theta}}} = \dot{\hat{\boldsymbol{\theta}}}$ due to the vector of real parameters $\boldsymbol{\theta}$ is considered constant. Then the time derivate of (15) can be written as:

$$\dot{V} = -\mathbf{e}_x^{\mathrm{T}}\mathbf{K}\,\mathbf{e}_x + \mathbf{e}_x^{\mathrm{T}}\mathbf{G}\,\tilde{\boldsymbol{\theta}} + \tilde{\boldsymbol{\theta}}^{\mathrm{T}}\dot{\hat{\boldsymbol{\theta}}} \tag{16}$$

Analyzing (16) is defined the following parameter-updating laws for the proposed adaptive controller.

$$\dot{\hat{\boldsymbol{\theta}}} = -\mathbf{G}^{T}\,\mathbf{e}_x \tag{17}$$

Substituting this expression in (16) we have

$$\dot{V} = -\mathbf{e}_x^{\,T}\mathbf{K}\,\mathbf{e}_x < 0 \qquad (18)$$

Where $\mathbf{K} = diag\left(k_1, k_2, k_3\right) > 0$ and $\mathbf{e}_x$ equals to $\left[x_{1d} - x_1, x_{2d} - x_2, x_{3ez} - x_3\right]^T$, therefore, (18) is negative definite and consequently the system with the proposed adaptive controller is asymptotically stable, which means that $\mathbf{e}_x(t) \to 0$ when $t \to \infty$.

An illustrative scheme shown in Fig. 2, where it is observed that the adaptation block is continuously sensing the state variables of the boiler-turbine and the desired states to be followed of the controller. The adaptation block updating the values of the $\hat{\boldsymbol{\theta}}$ according (17) make that the tracking error $\mathbf{e}_x(t)$ tends to zero (18).

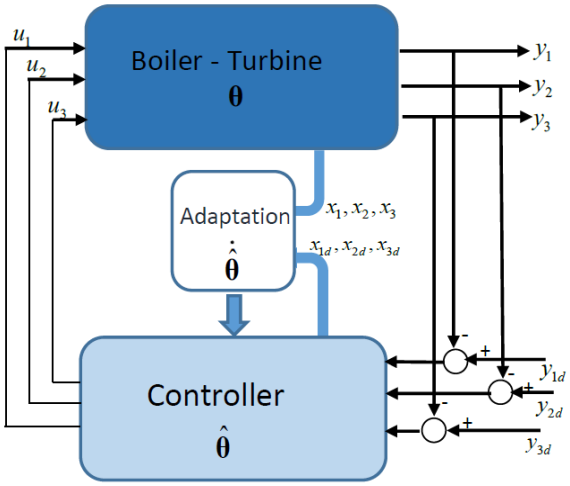


Fig. 2. Scheme. Adaptive Control Law.

5. SIMULATION RESULTS

To demonstrate the theoretical result obtained in the previous section, the proposed control system was tested in three different simulations. In the first one, the Monte Carlo method is applied to select an optimal set of controller parameters (Tempo et al., 2007); secondly, the controller developed is compared to different controllers proposed in the literature, for demanding tracking requirements; in third place, the controller is tested considering parametric uncertainty in the model. The added uncertainty is a 20% of the nominal value in all parameters of the model (Bell et al., 1987). In this simulation, the performance of the developed adaptation law is shown.

The goal of the simulations is to confirm the good performance of the controller obtained in the previous section. The simulations are performed using a simulator developed in the Matlab© platform, which considers an accurate model of the boiler-turbine. The control approach is applied on the original time-continuous system. The outputs $y_{1d}(t)$ and $y_{2d}(t)$ will be variable trajectories while $y_{3d}(t)$ will be maintained constant and equal to zero (Wu et al., 2010; Chen, 2013), .

5.1 Monte Carlo Experiment (MCE)

In this section the developed controller for boiler-turbine plant is subjected to an Monte Carlo Experimentation. The MCE aims to find the optimal parameters controller (k_1, k_2, k_3) optimizing a defined cost function. An idea widely used in the literature is to consider the cost incurred by the tracking error (Cheein et al., 2013). Let Φ be a desired trajectory, $C_{y_i}^{\Phi} = \frac{1}{2}\int_0^{t_f}\left(y_{d_i}(t) - y_i(t)\right)^2 dt$ the quadratic error of the output y_i with $i \in \{1,2,3\}$. Thus, the cost function can be represented by the combination of all quadratic errors,

$$C^{\Phi} = \frac{1}{2}\left(C_{y_1}^{\Phi} + C_{y_2}^{\Phi} + C_{y_3}^{\Phi}\right) \qquad (19)$$

In this work the MCE is carried out considering N=1000 simulations. This number was calculated according to (Tempo et al., 2007) in order to have a high accuracy in the solution found. The MC experiment allows to find empirically the parameter values minimizing the cost function (19).

The system begins balanced in the operating point #4 shown in Table 1. In *t=100s* is requested a change of operating point suddenly, from #4 at #2 (see Table 1). Then it is asked to follow these desired trajectories for y_{1d} and y_{2d}:

$$\underbrace{\begin{aligned} y_{1d} &= 9\sin(0.02(t-100)) + 0.02(t-100) + 86.4 \\ y_{2d} &= 9\sin(0.03(t-100)) + 0.05(t-100) + 36.65 \end{aligned}}_{100 \le t < 800}$$

The desired value of y_{3d} equals to zero for all the experimentation. This prevents any overheating of the boiler. The desired trajectory for the electrical output and drum pressure are out of a real operating requirement. However, the goal of proposing such paths is, to show the good performance of the proposed controller to follow any demanding and difficult requirement.

Experimental considerations of MCE:

- The simulations are performed using MatLab software platform.
- All simulations are implemented with the same desired trajectory Φ .

- For each simulation, the controller parameters are chosen in a random way, as specified in (Tempo et al., 2007).

The Fig. 3 shows the temporal response obtained for the specified desired path to N=1000 simulations. As it can be seen, the controller presents a good performance, despite the demanding requirement. The Fig. 4 shows the tracking errors outputs for each simulation. The convergence time of the errors is a function of the selected value of the controller´s parameters. The Fig. 5 shows the values taken by the cost function for each simulation. The minimum cost occurs in the simulation number 593, corresponding to the values of the controller´s parameters [k_1 k_2 k_3]=[1.2, 1.12, 1.05]. This values will be used in the next simulations as the optimal parameters of the controller.

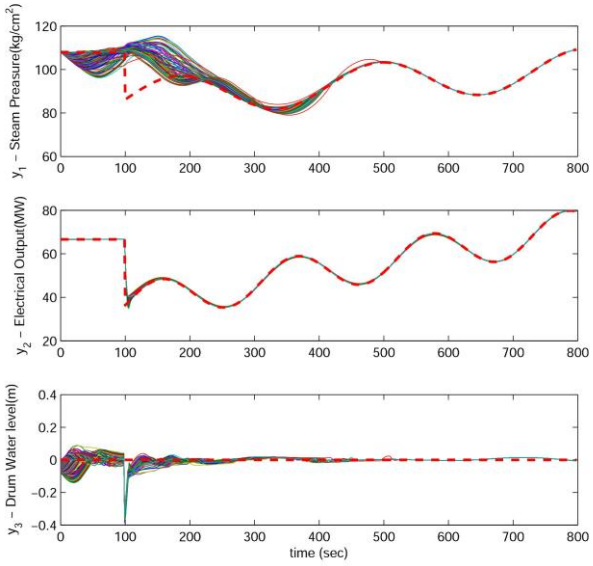


Fig. 3. Monte Carlo experimentation.

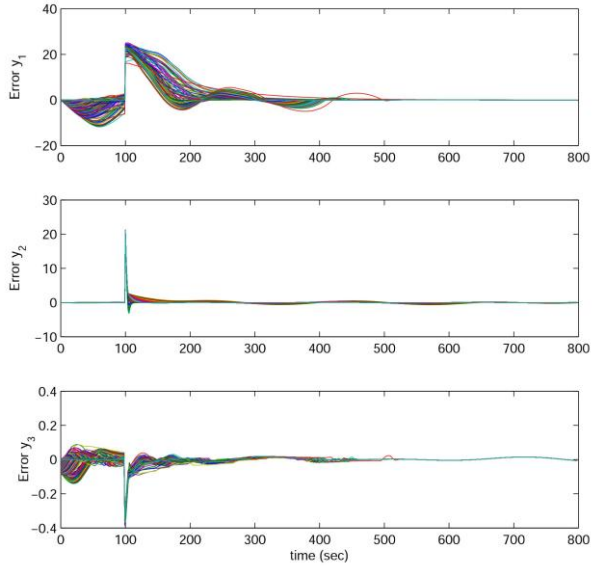


Fig. 4. Monte Carlo experiment. Outputs Errors.

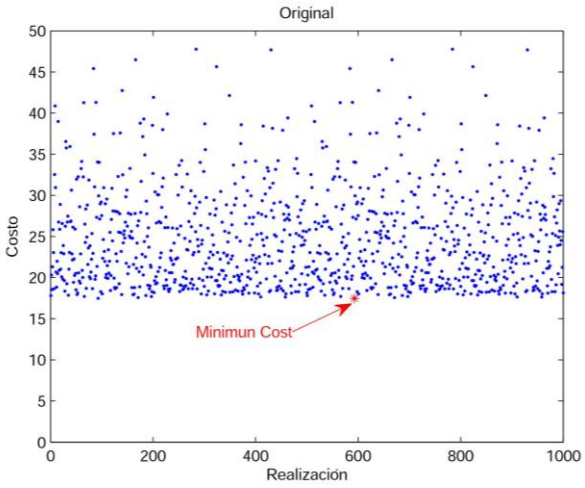


Fig. 5. Cost of Monte Carlo Experiment.

5.2 Controllers' Comparison

The performance to the proposed controller, when it does not considere uncertainties, is compared with two controllers developed in (Wu et al., 2010). The specification of the desired trajectory are (Wu et al., 2010):

- The desired path y_{1d} starts to be stabilized in 75,6 Kg/cm^2, then in *t=500s* an instant change is requested, taking a value equals to 135,4 Kg/cm^2, finally in *t=1500s* the output y_1 returns to the original value.

- The request y_{2d} suffers a set-point change from 15,27 MW to 127 MW in *t=500s*, then it returns to the original value in *t=1500s*.

- The desired trajectory y_{3d} remains constant and equals zero for the whole experiment.

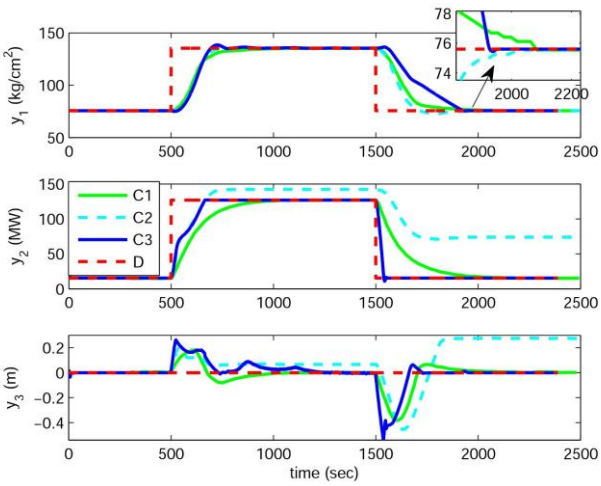


Fig. 6. Tracking trajectory. Comparison Controllers.

In this simulation the compared controller are C1= TS Control, C2= Linear Control and C3 = our proposal controller, where C1 and C2 are developed in(Wu et al., 2010). Fig. 6 shows the outputs, where in *D* represents the desired values for each output, while the Fig. 7 shows the control action. As it can be seen the response of the controller C3 have a better performance for the stipulated trajectory specially for y_2 that is the main variable to follow. In this one it can be observed that the C3 follows more quickly the desired requirements.

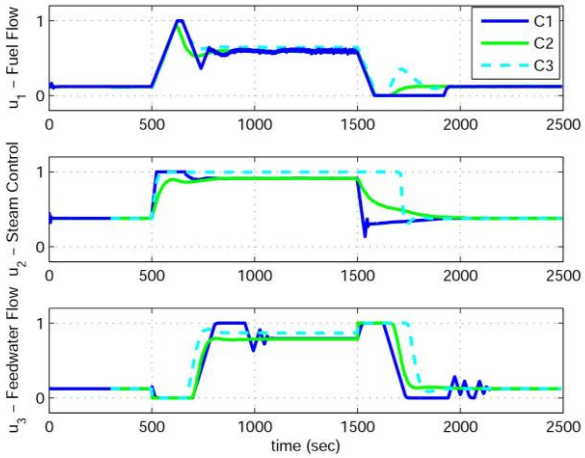


Fig. 7. Control Actions. Comparison Controllers.

5.3 Parametric Uncertain

In this section the performance of the controller incorporating the proposed adaptation law is tested. For this purpose, a comparison is performed between the controller with and without parameter updating development. The desired trajectories are formed by simultaneous set-points changes for the outputs y_1 and y_2. In this simulation it is considered a 20% of the parametric uncertainties in all parameters of the system.

The Fig. 8 shows the response of the proposed controller with and without parametric updating when the tracking trajectory is performed. In the Fig. 9 shows the tracking errors for the outputs using the proposed controller. As it can be seen the tracking error decreases, especially for the main output y_2 when the parametric updating is commissioning.

The Fig. 10 shows the cost function defined in (19) for both cases, where the cost is reduced in a 59,89% due to the adaptation law proposed.

The evolution of the estimated parameters along the time is shown in Fig. 11, where it is explained how the parameters are adapted to each change of operating point.

Despite that, the parameters tend to a fixed value.

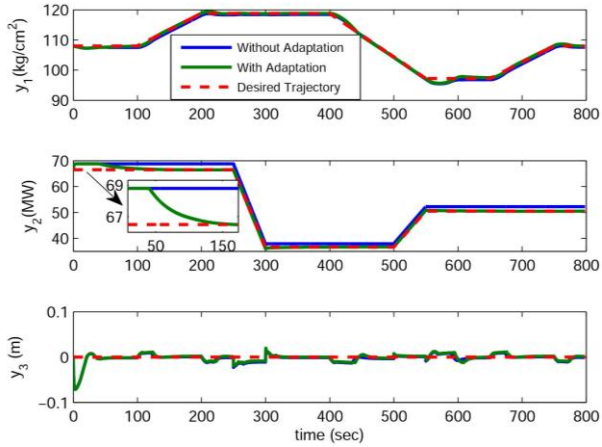


Fig. 8. Experiments with and without parameter updating.

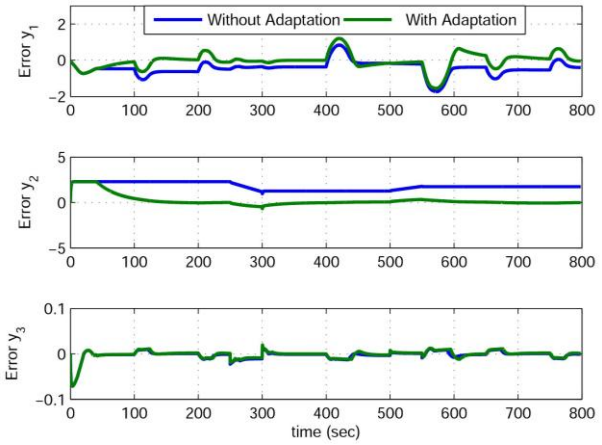


Fig. 9. Tracking errors with and without parameter updating.

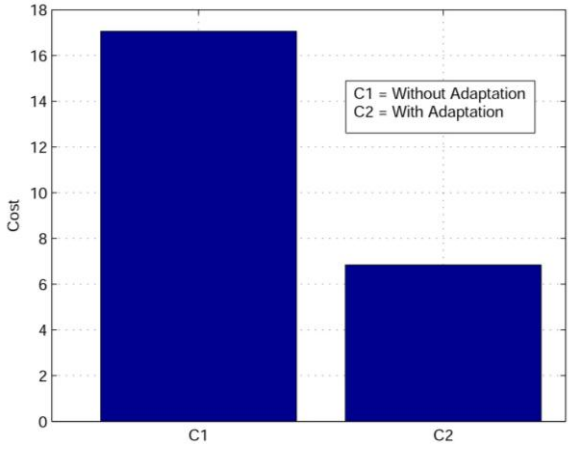


Fig. 10. Cost Function. Comparison.

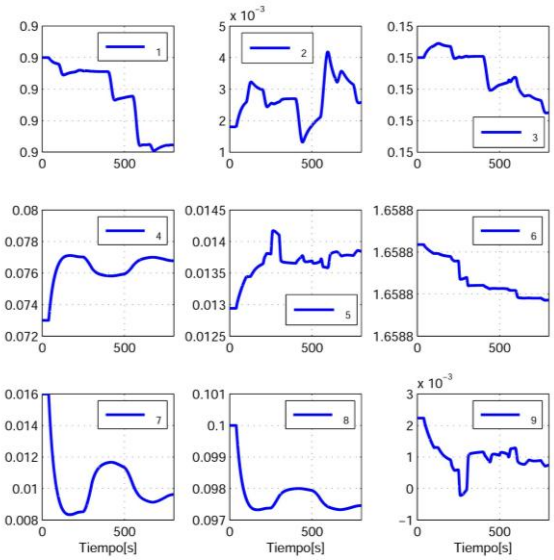


Fig. 11. Evolution of parameters estimates using parameter updating

6. ACKNOWLEDGEMENTS

This work was partially funded by the Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET – National Council for Scientific Research), Argentina and the Universidad Nacional de San Juan (UNSJ), San Juan, Argentina.

7. CONCLUSION

In this paper, a new control law with parameter updating for trajectory tracking in a boiler-turbine was proposed. The control actions for a zero-error trend, are obtained through linear algebra techniques by means of the resolution of a linear system. These techniques are based on simple methods and do not require the use of coordinate transformation, and complicated adaptive scheme; the process model and the values of y_{1d} and y_{2d} are the only variables needed to be known.

The main advantages of the proposed controller are the simplicity of the design procedure and the ability to follow any path allowable for y_1 and y_2 since, in normal operating conditions. The different tests carried out in this work prove the good performance of the proposed controller design procedure, even when they are compared with a controller of the literature. In fact, the system behavior was tested, reaching better performance than those ones obtained by (Wu et al., 2010). The parameter updating is obtained through the study of the stability analysis based on Lyapunov theory. The convergence to zero of the tracking errors for the controller and for the adaptation law were demonstrated by the Lyapunov theory. The results proved that the proposed controller is capable of tracking a desired trajectory with a small error when the dynamic parameters are adapted.

REFERENCES

Bell, R. and K. J. Åström (1987). Dynamic models for boiler-turbine-alternator units: data logs and parameter estimation for a 160 MW unit, Lund Institute of Technology, Department of Automatic Control.

Brent, R. P. (2013). Algorithms for minimization without derivatives, Courier Corporation.

Cheein, F. A. A., Scaglia, G., Torres-Torriti, M., Guivant, J., Prado, A. J., Arnò, J., ... & Rosell-Polo, J. R. (2016). Algebraic path tracking to aid the manual harvesting of olives using an automated service unit. Biosystems Engineering, 142, 117-132.

Cheein, F. A. and G. Scaglia (2013). "Trajectory Tracking Controller Design for Unmanned Vehicles: A New Methodology." Journal of Field Robotics: n/a-n/a.

Chen, P.-C. and J. S. Shamma (2004). "Gain-scheduled ℓ 1-optimal control for boiler-turbine dynamics with actuator saturation." Journal of Process Control 14(3): 263-277.

Fang, F., L. Jizhen, et al. (2004). Output tracking control of a nonlinear boiler-turbine unit. Decision and Control, 2004. CDC. 43rd IEEE Conference on.

Ghabraei, S., Moradi, H., & Vossoughi, G. (2015). Multivariable robust adaptive sliding mode control of an industrial boiler–turbine in the presence of modeling imprecisions and external disturbances: A comparison with type-I servo controller. ISA transactions, 58, 398-408.

Khalil, H. K. and J. Grizzle (1996). Nonlinear systems, Prentice hall New Jersey.

Romoli, S., Scaglia, G. J. E., Serrano, M. E., Godoy, S. A., Ortiz, O. A., & Vega, J. R. (2014). Control of a Fed-Batch Fermenter Based on a Linear Algebra Strategy. IEEE Latin America Transactions, 12(7), 1206-1213.

Sarailoo, M., B. Rezaie, et al. (2012). "MLD model of boiler-turbine system based on PWA linearization approach." International Journal of Control Science and Engineering 2(4): 88-92.

Scaglia, G., L. Q. Montoya, et al. (2009). "Numerical methods based controller design for mobile robots." Robotica 27(02): 269-279.

Serrano, M. E., Scaglia, G. J., Godoy, S. A., Mut, V., & Ortiz, O. A. (2014). Trajectory tracking of underactuated surface vessels: A linear algebra approach. IEEE Transactions on Control Systems Technology, 22(3), 1103-1111.

Strang, G. (2006). Linear Algebra and Its applications. USA.

Tan, W., F. Fang, et al. (2008). "Linear control of a boiler–turbine unit: Analysis and design." ISA transactions 47(2): 189-197.

Tan, W., H. J. Marquez, et al. (2004). "Multimodel analysis and controller design for nonlinear processes." Computers & chemical engineering 28(12): 2667-2675.

Tan, W., H. J. Marquez, et al. (2005). "Analysis and control of a nonlinear boiler-turbine unit." Journal of Process Control 15(8): 883-891.

Tan, W., J. Liu, et al. (2004). "Tuning of PID controllers for boiler-turbine units." ISA transactions 43(4): 571-583.

Tempo, R. and H. Ishii (2007). "Monte Carlo and Las Vegas Randomized Algorithms for Systems and Control*:

An Introduction." European journal of control 13(2): 189-203.

Thangavelusamy, D. and L. Ponnusamy (2013). "Elimination of chattering using fuzzy sliding mode controller for drum boiler turbine system." Journal of Control Engineering and Applied Informatics 15(2): 78-85.

Wu, J., J. Shen, et al. (2012). "GA-based nonlinear predictive switching control for a boiler-turbine system." Journal of control theory and applications 10(1): 100-106.

Wu, J., M. Krug, et al. (2009). H∞ fuzzy tracking control for boiler-turbine systems. Control and Automation, 2009. ICCA 2009. IEEE International Conference on, IEEE.

Wu, J., S. K. Nguang, et al. (2010). "Robust tracking control of boiler–turbine systems." ISA Transactions 49(3): 369-375.

Wu, X., J. Shen, et al. (2013). "Data-driven modeling and predictive control for boiler–turbine unit." Energy Conversion, IEEE Transactions on 28(3): 470-481.

Zheng, K., A.-H. Lee, et al. (2006). "Steady-state bumpless transfer under controller uncertainty using the state/output feedback topology." Control Systems Technology, IEEE Transactions on 14(1): 3-17.

Appendix A

Theorem 1: Considering accurate knowledge of the model according *Remark 4* and replacing the expression of the control actions (9) in the system (6) are obtained the error equation of the control system.

$$\begin{aligned}&\mathbf{A}\left(\mathbf{A}^{-1}\boldsymbol{\sigma}+\mathbf{A}^{-1}\mathbf{B}\mathbf{x}\right)=\dot{\mathbf{x}}+\mathbf{B}\mathbf{x}\\&\boldsymbol{\sigma}+\mathbf{B}\mathbf{x}=\dot{\mathbf{x}}+\mathbf{B}\mathbf{x}\\&\left(\boldsymbol{\sigma}-\dot{\mathbf{x}}\right)=0\end{aligned}\tag{20}$$

According to (7) we have $\boldsymbol{\sigma}-\dot{\mathbf{x}}=\dot{\mathbf{e}}_{\mathbf{x}}+\mathbf{K}\mathbf{e}_{\mathbf{x}}$, where $\mathbf{K}=diag\left(k_1,k_2,k_3\right)>0$ and $\mathbf{e}_{\mathbf{x}}=\left[x_{1d}-x_1,x_{2d}-x_2,x_{3ez}-x_3\right]^T$, thus (20) can be written as follows

$$\dot{\mathbf{e}}_{\mathbf{x}}=-\mathbf{K}\mathbf{e}_{\mathbf{x}}\tag{21}$$

The expression (21) represent the error equation of the control system when there is no presence of parameter uncertainty.

It is now considered the Lyapunov candidate function

$$V=\frac{1}{2}\mathbf{e}_{\mathbf{x}}{}^{T}\mathbf{e}_{\mathbf{x}}>0\tag{22}$$

The time derivative of the Lyapunov candidate function can be written as

$$\dot{V}=\mathbf{e}_{\mathbf{x}}{}^{T}\dot{\mathbf{e}}_{\mathbf{x}}=-\mathbf{e}_{\mathbf{x}}{}^{T}\mathbf{K}\,\mathbf{e}_{\mathbf{x}}<0\tag{23}$$

is negative definite.

Hence, one can straightforwardly conclude that the system controlled by proposed controller (10), when there is no presence of parameter uncertainty, has an asymptotically stable equilibrium at the origin, which means that $\mathbf{e}_{\mathbf{x}}(t)\to 0$ when $t\to\infty$. ■

Appendix B

Equation (13) represent the error equation of the control system in function of the system matrix, when there is parametric uncertainty

$$\dot{\mathbf{e}}_{\mathbf{x}}+\mathbf{K}\mathbf{e}_{\mathbf{x}}=\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\boldsymbol{\sigma}+\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x}-\tilde{\mathbf{B}}\mathbf{x}\tag{24}$$

The objective is replace the right side of the equation (24) by expressions that depend on $\tilde{\boldsymbol{\theta}}=\left[\tilde{\theta}_1,\ldots,\tilde{\theta}_9\right]^T$. Tacking the last term of the right side of (24):

$$\tilde{\mathbf{B}}\mathbf{x}=\begin{bmatrix}0\\ \tilde{\theta}_7x_1^{9/8}+\tilde{\theta}_8x_2\\ \tilde{\theta}_9x_1\end{bmatrix}$$

Therefore

$$\tilde{\mathbf{B}}\mathbf{x}=\begin{bmatrix}0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&x_1^{9/8}&x_2&0\\0&0&0&0&0&0&0&0&-x_1\end{bmatrix}\tilde{\boldsymbol{\theta}}\tag{25}$$

$$\tilde{\mathbf{B}}\mathbf{x}=\mathbf{G}_1\tilde{\boldsymbol{\theta}}$$

Proceeding in the same way and using the Symbolic Math Toolbox of MatLab, the expression of the two remaining terms of (24) were obtained. The expression of the term $\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\boldsymbol{\sigma}$ is equivalent to:

$$\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\boldsymbol{\sigma}=\begin{bmatrix}a_{11}&\cdots&a_{19}\\ \vdots&\ddots&\vdots\\ a_{91}&\cdots&a_{99}\end{bmatrix}\tilde{\boldsymbol{\theta}}=\mathbf{G}_2\tilde{\boldsymbol{\theta}}\tag{26}$$

Where the elements a_{ij} are equal to zero excepting the positions $\left\{a_{11},a_{12},a_{13},a_{24},a_{35},a_{36}\right\}$, whose expression are

$$a_{11} = \left(\frac{\sigma_1}{\hat{\theta}_1} + \frac{\sigma_3 \hat{\theta}_3}{\hat{\theta}_1 \hat{\theta}_6} + \sigma_2 \left(\frac{\hat{\theta}_2}{\hat{\theta}_1 \hat{\theta}_4} + \frac{\hat{\theta}_3 \hat{\theta}_5 \, x_1^{-1/8}}{\hat{\theta}_1 \hat{\theta}_4 \hat{\theta}_6} \right) \right)$$

$$a_{12} = -\frac{\sigma_2}{\hat{\theta}_4}$$

$$a_{13} = \left(-\frac{\sigma_3}{\hat{\theta}_6} - \frac{\sigma_2 \hat{\theta}_3 \, x_1^{-1/8}}{\hat{\theta}_4 \hat{\theta}_6} \right)$$

$$a_{24} = \frac{\sigma_2}{\hat{\theta}_4}$$

$$a_{35} = -\frac{\sigma_2 \, x_1^{-1/8}}{\hat{\theta}_4}$$

$$a_{36} = \left(\frac{\sigma_3}{\hat{\theta}_6} + \frac{\sigma_2 \hat{\theta}_5 \, x_1^{-1/8}}{\hat{\theta}_4 \hat{\theta}_6} \right)$$

The expression of the $\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x}$ is equivalent to:

$$\tilde{\mathbf{A}}\hat{\mathbf{A}}^{-1}\hat{\mathbf{B}}\mathbf{x} = \begin{bmatrix} b_{11} & \cdots & b_{19} \\ \vdots & \ddots & \vdots \\ b_{91} & \cdots & b_{99} \end{bmatrix} \tilde{\boldsymbol{\theta}} = \mathbf{G}_3 \tilde{\boldsymbol{\theta}} \qquad (27)$$

Whose nonzero elements are:

$$b_{11} = x_1 \left(\hat{\theta}_3 \hat{\theta}_5 \hat{\theta}_7 - \hat{\theta}_3 \hat{\theta}_4 \hat{\theta}_9 \right) + \hat{\theta}_2 \hat{\theta}_6 \hat{\theta}_8 x_2 + \ldots$$

$$\ldots + \hat{\theta}_2 \hat{\theta}_6 \hat{\theta}_7 x_1^{9/8} + \frac{\hat{\theta}_3 \hat{\theta}_5 \hat{\theta}_8 \, x_2 x_1^{-1/8}}{\hat{\theta}_1 \hat{\theta}_4 \hat{\theta}_6}$$

$$b_{12} = \frac{\hat{\theta}_8 \, x_2}{\hat{\theta}_4} - \frac{\hat{\theta}_7 \, x_1^{9/8}}{\hat{\theta}_4}$$

$$b_{13} = -\frac{\hat{\theta}_5 \hat{\theta}_7 \, x_1}{\hat{\theta}_4 \hat{\theta}_6} + \frac{\hat{\theta}_9 \, x_1}{\hat{\theta}_6} - \frac{\hat{\theta}_5 \hat{\theta}_8 \, x_2 x_1^{-1/8}}{\hat{\theta}_4 \hat{\theta}_6}$$

$$b_{24} = \frac{\hat{\theta}_8 \, x_2 + \hat{\theta}_7 x_1^{9/8}}{\hat{\theta}_4}$$

$$b_{35} = -\frac{x_1^{-9/8}}{\hat{\theta}_4} \left(\hat{\theta}_8 x_2 + \hat{\theta}_7 x_1^{9/8} \right)$$

$$b_{36} = -\frac{\hat{\theta}_9 \, x_2}{\hat{\theta}_6} + \frac{\hat{\theta}_5 x_1^{-9/8}}{\hat{\theta}_4 \hat{\theta}_6} \left(\hat{\theta}_8 x_2 + \hat{\theta}_7 x_1^{9/8} \right)$$

The matrices $\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3$ are completely constituted by know values. The matrix $\mathbf{G}$ is equal to $\mathbf{G} = \mathbf{G}_2 + \mathbf{G}_3 - \mathbf{G}_1$ and your expression is

$$\mathbf{G}\tilde{\boldsymbol{\theta}} = \begin{bmatrix} a_{11}+b_{11} & 0 & 0 \\ a_{12}+b_{12} & 0 & 0 \\ a_{13}+b_{13} & 0 & 0 \\ 0 & a_{24}+b_{24} & 0 \\ 0 & 0 & a_{35}+b_{35} \\ 0 & 0 & a_{36}+b_{36} \\ 0 & -x_1^{9/8} & 0 \\ 0 & -x_2 & 0 \\ 0 & 0 & x_1 \end{bmatrix}^T \tilde{\boldsymbol{\theta}}$$

Adaptive Twisting Controller using Dynamic Gain ⋆

Raúl Santiesteban Cos *

* *Tecnológico Nacional de México, Campus Culiacán. Departamento de Metal-Mecánica. E-mail:raulsncos@gmail.com*

Abstract. In this paper, a new variable gain controller for the double integrator is proposed. A second order sliding mode, such as Twisting control synthesis is presented using variable gain. In order to keep properties such as finite time stability, a first order sliding mode is designed as variable gain. A nonsmooth strict Lyapunov function is identified to establish global finite time stability of the nominal system. In order to support theoretical results, a one-link pendulum is considered as a test bed considering the presence of bounded, persistent and state space variable dependant disturbances.

Keywords: Sliding mode control; Stability analysis; Lyapunov methods.

1. INTRODUCTION

In the last decades, the sliding mode control become a popular strategy for control of nonlinear uncertain systems, with a very large frame of applications fields, particularly with electromechanical systems (Orlov et al. [2003], H. Sira-Ramirez and Rodriguez-Angeles [2010], Shtessel et al. [2007]). Considering a discontinuous function and using high control gain, the main features of this strategy are the robustness of closed-loop system and the finite-time convergence to the point of interest. One of the first second order sliding modes controllers (SOSM), the twisting algorithm, became very popular due its advantage to consider Coulomb friction of a mechanical system as part of the controller (S. V. Emelyanov and Levantovsky [1986]). Moreover, it is well known that this algorithm has properties such as finite time stability (FTS) and robustness against bounded external perturbations (see for example Orlov [2008]).

Then, the main drawback of the sliding mode control, the well-known chattering phenomenon (for its analysis, see Boiko and Fridman [2005], Fridman [2003]), is important because it could damage actuators and systems. In literature, several works on the literature have been devoted to modify SOSM controllers to reduce this phenomenon. A first way is using another kind of discontinuous functions (see Orlov et al. [2011], Bhat and Bernstein [1998], Qin and Duan [2012]) in order to design a smooth control law. However, there is a trade-off between the robustness of the closed-loop system and the chattering present in the control action. A second way to decrease the chattering phenomenon is the use of higher order sliding mode controller (G. Bartolini and Utkin [2000], Levant [2007], Fridman and Levant [2002]).

Adaptive sliding mode is an alternative way considered in the literature (see for example (T. Gonzales and Fridman [2012]), (Edwards and Shtessel [2014])). The main objective is to design a dynamical adaptation of the control gain in order to be as small as possible whereas sufficient to counteract the uncertainties and perturbations. Indeed, in (Lee and Utkin [2007]) and the references there in, the chattering phenomenon depends proportionally on the gain of the controller, *i.e.* a big gains or over-estimated gain, gives larger control magnitude and larger chattering (see for example (Y. J. Huang and Chang [2008]), (F. Plestan and Poznyak [2010]), (M. Dal [2011])). In this works a first order sliding mode is under study using a dynamic law for the gain of the controller. In (Y. J. Huang and Chang [2008]), the idea is based on use of equivalent control: once sliding mode occurs, disturbance magnitude is evaluated and allows an adequate tuning of control gain. In (F. Plestan and Poznyak [2010]) the gain is dynamically tuned in order to ensure the establishment of a sliding mode; once this sliding mode is achieved, the gain is adjusted in order to get a "sufficient" value in order to counteract the perturbations and uncertainties. In all this works a first order sliding mode is considered. In (A. Levant and Plestan [2011]) a dynamic adaptation law is used in the twisting algorithm where the uncertainty is bounded, so that one knows in advance the value of the gain controller which is sufficient to establish and keep the sliding mode.

This work is based on the idea of dynamic adaptation law, *i.e.* variable gain (Lee and Utkin [2007]) and a second order sliding mode, the well known Twisting algorithm. In practice, usually the constant perturbations are small, then the uncertainty term can be modeled as a constant term and a state space variable and/or time dependant term. With this in mind, a first order sliding mode is designed with respect to the gain of the controller, in order to follow an adaptation rule known *a priori*, *i.e.* save energy and reduce chattering phenomenon. Moreover, the idea of implement any adaptation rule for the gain can be considered. A nonsmooth strict Lyapunov function is proposed to analyze the stability of the nominal closed-loop system. The sta-

⋆ CONACYT grant 53869.

bility analysis shows that this algorithm preserves the well known properties of a sliding mode controller such as finite time stability. The robust algorithm will be considered in a separate paper, nonetheless numerical simulations are shown considering bounded and state space dependant disturbances, in order to support theoretical results.

In section 2 the problem statement is presented: the stabilization of an uncertain system is under study. In section 3, the closed loop system is analyzed using non-smooth Lyapunov functions, in section 4 to support theoretical results a numerical example is shown and in section 5 are the conclusions of this work.

2. PROBLEM STATEMENT

Consider an uncertain system that describes the dynamics of a mechanical system and it is well known that has relative degree two,

$$\begin{aligned} \dot{x} &= y \\ \dot{y} &= f(x,t) + \tau + \delta(x,y,t),\ x, y \in \mathbf{R}, \end{aligned} \tag{1}$$

The known part of the system dynamics is represented by function $f(x,t)$. The not known part of system dynamics (such uncertainties, external/parametric perturbations, noise, among others) are concentrated in $\delta(t,x,y)$ and the control signal is represented by variable τ. The uncertainty term is considered bounded by

$$|\delta(x,y,t)| < M(x,y,t) + \mu \tag{2}$$

where $M(x,y,t)$ is Lipschitz and μ is a positive constant. The solutions of all systems of differential equations are understood in the Filippov's sense Filippov [1988]. For system (1) the following control design is proposed

$$\tau = U - f(x,t) \tag{3}$$

where U is the proposed algorithm. In the next section an homogeneous control law is proposed in order to achieve finite time stability using a non smooth Lyapunov function.

3. NOMINAL SYSTEM

In this section, the stability of a control law of the unperturbed system (5) will be studied. Indeed, consider the nominal system (1) and the new proposal, Dynamic Gain Twisting Control Law (DGT),

$$\begin{aligned} U &= -\left(|\alpha| + \beta\right)\left(\mathrm{sgn}(x) - \gamma \mathrm{sgn}(y)\right) \\ \dot{\alpha} &= -k_1 \mathrm{sgn}(\alpha - M(x,y,t)) - k_2(\alpha - M(x,y,t)) \end{aligned} \tag{4}$$

where β, k_1 and k_2 are positive constants and $0 < \gamma < 1$ and $M(x,y,t)$ is as equation (2). Then, the closed loop system is as follows uncertainty term.

$$\begin{aligned} \dot{x} &= y \\ \dot{y} &= -\left(|\alpha| + \beta\right)\left(\mathrm{sgn}(x) + \gamma \mathrm{sgn}(y)\right) \\ \dot{\alpha} &= -k_1 \mathrm{sgn}(S) - k_2 S \end{aligned} \tag{5}$$

where the $S = \alpha - M(x,y,t)$.

Theorem 1. Let the parameters of the switched system (5) be such that conditions

$$\begin{aligned} k_1 > \frac{1}{1-\nu}\dot{M};\ \ k_2 > \nu;\ \ \nu\beta\gamma > \dot{M} \\ \nu > 1;\ \ 0 < \gamma < 1 \end{aligned} \tag{6}$$

are satisfied and ν, k_1 and k_2 are positive constants. Then, the system (5) is globally uniformly finite time stable around the set $(x,y,z) = (0,0,M)$.

proof:

In order to study the stability of the set $(x,y,\alpha) = (0,0,M(x,y,t))$ of system (5), a nonsmooth Lyapunov function is proposed.

$$\begin{aligned} V(x,y,\alpha) = &\Big(k_1 + \nu\beta\Big)|x| + \Big(k_2|S| + \nu|\alpha|\Big)|x| \\ &+ \frac{1}{2}\nu y^2 + |S| y \mathrm{sgn}(x) + \frac{1}{2}S^2 \end{aligned} \tag{7}$$

with ν as a positive constant.

First step. Lets show that $V(x,y,\alpha)$ is a positive definite function.

$$\begin{aligned} V(x,y,\alpha) = &\Big(k_1 + \nu\beta\Big)|x| + \Big(k_2|S| + \nu|\alpha|\Big)|x| \\ &+ \frac{1}{2}\rho^T P \rho \end{aligned} \tag{8}$$

where $\rho^T = [y\ \ S]$

$$P = \begin{pmatrix} \nu & 1 \\ 1 & 1 \end{pmatrix} \tag{9}$$

Now, if $det(P) > 0$ then function (7) is definite positive, then the following inequality must hold at all time: $\nu > 1$.

Moreover, a lower bound and an upper bound of $V(x,y,\alpha)$ are estimated using the eigenvalues of matrix P, i.e. $\lambda_{max}(P)$, as follows

$$\begin{aligned} \zeta_{min}\Big(|x| + |S||x| + |\alpha||x| + ||\rho||^2\Big) \leq \\ V(x,y,\alpha) \leq \zeta_{max}\Big(|x| + |S||x| + |\alpha||x| + ||\rho||^2\Big) \end{aligned} \tag{10}$$

where

$$\begin{aligned} \zeta_{min} &= \min\left\{k_1 + \nu\beta, k_2, \nu, \frac{1}{2}\lambda_{min}(P)\right\} \\ \zeta_{max} &= \max\left\{k_1 + \nu\beta, k_2, \nu, \frac{1}{2}\lambda_{max}(P)\right\} \end{aligned} \tag{11}$$

Since $xy \leq x^2 + y^2$, note that the terms $|S||x|$ and $|\alpha||x|$ can be described using quadratic functions as follows:

$$V(x,y,\alpha) \leq \zeta_{max}\Big(|x| + x^2 + \alpha^2 + y^2 + S^2\Big) \tag{12}$$

Now, let's calculate the time derivative of $V(x,y,\alpha)$. Note that $V(x,y,\alpha)$ is locally Lipschitz everywhere, and it is differentiable at any point except on the set defined by $S = \{(x,y,\alpha)|x = 0\}$. The set S does not contain any trajectory of system (5). This means that $\dot{V}(x,y,\alpha)$ computed along the trajectory $(x(t),y(t),\alpha(t))$ exists almost everywhere.

The time derivative of (7) along the trajectories of the perturbed system (5) is given by

$$\begin{aligned}\dot{V}(x,y,\alpha) = &(k_1+\nu\beta)\,\mathrm{sgn}(x)y\\ &+k_2|S|\mathrm{sgn}(x)y\\ &-k_2|x|\mathrm{sgn}(S)\left(k_1\mathrm{sgn}(S)+k_2S+\dot{M}\right)\\ &+\nu|\alpha|\mathrm{sgn}(x)y-\nu|x|\mathrm{sgn}(\alpha)\left(k_1\mathrm{sgn}(S)+k_2S\right)\\ &-\nu y\left(|\alpha|+\beta\right)\left(\mathrm{sgn}(x)+\gamma\mathrm{sgn}(y)\right)\\ &-y\mathrm{sgn}(x)\mathrm{sgn}(S)\left(k_1\mathrm{sgn}(S)+k_2S+\dot{M}\right)\\ &-|S|\mathrm{sgn}(x)\left(|\alpha|+\beta\right)\left(\mathrm{sgn}(x)+\gamma\mathrm{sgn}(y)\right)\\ &-S\left(k_1\mathrm{sgn}(S)+k_2S+\dot{M}\right)\end{aligned} \tag{13}$$

$$\begin{aligned}\dot{V}(x,y,\alpha) = &(k_1+\nu\beta)\,\mathrm{sgn}(x)y\\ &+k_2|S|\mathrm{sgn}(x)y\\ &-k_2|x|\left(k_1+k_2|S|+\dot{M}\mathrm{sgn}(S)\right)\\ &+\nu|\alpha|\mathrm{sgn}(x)y-\nu|x|\left(k_1\mathrm{sgn}(S\alpha)+k_2S\mathrm{sgn}(\alpha)\right)\\ &-\nu y|\alpha|\left(\mathrm{sgn}(x)+\gamma\mathrm{sgn}(y)\right)\\ &-\nu y\beta\left(\mathrm{sgn}(x)+\gamma\mathrm{sgn}(y)\right)\\ &-y\mathrm{sgn}(x)\left(k_1+k_2|S|+\dot{M}\mathrm{sgn}(S)\right)\\ &-|S|\left(|\alpha|+\beta\right)\left(1+\gamma\mathrm{sgn}(xy)\right)\\ &-|S|\left(k_1+k_2|S|+\dot{M}\mathrm{sgn}(S)\right)\end{aligned} \tag{14}$$

$$\begin{aligned}\dot{V}(x,y,\alpha) \le &-k_2\left(k_1-\dot{M}\mathrm{sgn}(S)\right)|x|-k_2^2|S||x|\\ &-\nu|x|\left(k_1\mathrm{sgn}(S\alpha)+k_2S\mathrm{sgn}(\alpha)\right)\\ &-\nu|\alpha|\gamma|y|-\nu\beta\gamma|y|\\ &-\dot{M}y\mathrm{sgn}(xS)\\ &-|S|\left(|\alpha|+\beta\right)\left(1+\gamma\mathrm{sgn}(xy)\right)\\ &-|S|\left(k_1+k_2|S|+\dot{M}\mathrm{sgn}(S)\right)\end{aligned} \tag{15}$$

$$\begin{aligned}\dot{V}(x,y,\alpha) \le &-k_2\left(k_1-\dot{M}-k_1\nu\right)|x|\\ &-k_2\left(k_2-\nu\right)|S||x|\\ &-\nu\gamma|\alpha||y|-\left(\nu\beta\gamma-\dot{M}\right)|y|\\ &-(1-\gamma)\,|\alpha||S|-(1-\gamma)\,\beta|S|\\ &-\left(k_1-\dot{M}\right)|S|+k_2S^2\end{aligned} \tag{16}$$

In order to proof that the function $\dot{V}(x,y,\alpha)$ is negative definite, the following inequalities must be satisfied at all time:

$$k_1 > \dot{M}+k_1\nu;\;\; k_2>\nu;\;\; \nu\beta\gamma>\dot{M} \tag{17}$$

Using the following inequality

$$\begin{aligned}\xi_{min} = \min\Bigg\{ &-k_2\left(k_1-\dot{M}-k_1\nu\right);\;\; k_2\left(k_2-\nu\right);\\ &\nu\gamma;\;\; \left(\nu\beta\gamma-\dot{M}\right);\;\; (1-\gamma);\\ &(1-\gamma)\,\beta;\;\; \left(k_1-\dot{M}\right)\Bigg\}\end{aligned} \tag{18}$$

an upper bound for the Lyapunov function $V(x,y,\alpha)$ is estimated as follows

$$\dot{V}(x,y,\alpha) \le -\xi\left(|x|+x^2+\alpha^2+y^2+S^2\right) \tag{19}$$

Equation (19) can be written in terms of function $V(x,y,\alpha)$ using inequality (12)

$$\dot{V}(x,y,\alpha) \le -\frac{\xi}{\zeta_{max}}V(x,y,\alpha) \tag{20}$$

The above relation ensure that function $V(x,y,\alpha)$ decays exponentially

$$V(x,y,\alpha) \le V(x(0),y(0),\alpha(0))e^{\left(-\frac{\xi}{\zeta_{max}}\right)} \tag{21}$$

on the solutions of (5), uniformly on the uncertainty (2) and the initial data. Clearly, this proves that the system (5) is locally uniformly asymptotically stable (see for example Orlov [2005]). Then the set $(x,y,\alpha) = (0,0,M(x,y,t))$ of system (5) is asymptotically stable if inequality (6) holds. Due to the piece-wise continuous term k_2S is locally uniformly bounded, whereas the right-hand side of the nominal model (5) is piece-wise continuous and globally homogeneous of degree $q=-1$ with respect to dilation $r=(2,1,1)$. Hence, the condition $q+r3\le 0$, required by Theorem 3.2 in Orlov [2005] is satisfied, and is applicable to the uniformly asymptotically stable system (5). By applying this Theorem, the system (5) is thus globally uniformly finite time stable. The proof of Theorem 1 is completed.

4. SIMULATIONS

4.1 Problem Statement

In this section, the control problem known as tracking is considered, using the one-link pendulum system as a test bed to illustrate the DGT controller performance. The state equation of a controlled one-link pendulum (see Fig. 1) is given by

$$(ml^2+J)\ddot{q} = mglsin(q) - F(\dot{q}) + \tau + \delta(t,q,\dot{q}) \tag{22}$$

where q is the angle made by the pendulum with the

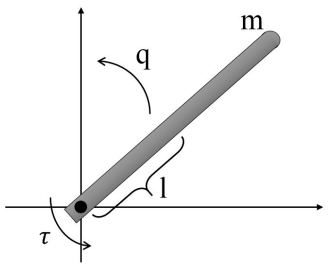


Figure 1. The one-link pendulum system.

vertical, m is the mass of the pendulum, l is the distance to the center of mass, J is the moment of inertia of the pendulum about the center of mass, g is the gravity acceleration, F is the friction force, τ is the control torque, and $\delta(t,q,\dot{q})$ is the external disturbance. The friction force F is described by

$$F(\dot{q}) = \rho_v\dot{q} + \rho_c sign(\dot{q}). \tag{23}$$

where $\rho_v > 0$ denotes the viscous friction coefficient and $\rho_c > 0$ denotes the Coulomb friction level. Subject

to (23), the right-hand side of the dynamic system (22) is piece-wise continuous. An uncertainty term $\delta(t, q, \dot{q})$ is introduced into the dynamic equation (22) and it is bounded by (2). *The control objective* is to drive the one-link pendulum to a known trajectory in exact finite time, *i.e.*

$$q(t) - x(t) = 0 \tag{24}$$

where $x(t) = sin(t)$ even in the presence of an admissible external disturbance (2). Let the tracking error be given by

$$y(t) = q(t) - x(t). \tag{25}$$

Using the DGT control in the form (4)

$$\begin{aligned} \tau &= -\left(|\alpha| + \beta\right)\left(\text{sgn}(x) + \gamma \text{sgn}(y)\right) \\ \dot{\alpha} &= -k_1 \text{sgn}(S) - k_2 S \end{aligned} \tag{26}$$

where $S = \alpha - M(x, y, t)$ and $M(x, y, t) = sin(q)$ the error dynamics can be written as follows

$$\begin{aligned} (ml^2 + J)\ddot{y} &= -\left(|\alpha| + \beta\right)\left(\text{sgn}(x) + \gamma \text{sgn}(y)\right) \\ &\quad + \delta(x, y, t) \\ \dot{\alpha} &= -k_1 \text{sgn}(S) - k_2 S \end{aligned} \tag{27}$$

In order to show the performance of the proposed algorithm, a comparison with the well known twisting controller is considered. In order to achieve the control objective, $\delta(x, y, t)$ is a perturbation bounded by a positive constant M in the case of twisting algorithm (see for example Orlov [2005]). Considering the new proposal, the uncertainty term is given by $\delta(x, y, t) = mglsin(q) - F(\dot{q})$. Then the DGT controller gain is design considering $M(x, y, t) = 4sin(q)$ (see equation (2)).

In order to make show the good performance of this new proposal, three cases are considered. In the first, second and third case, the compensation of the perturbations is the only criterion considered. Three sets of gains are considered, in order to give more information about the performance of both controllers. In the last case, parametric perturbations are considered.

Parameters of a real laboratory one-link pendulum system are considered: the mass of the pendulum is $m = 0.5234kg$, the length of the link $l = 0.108m$, and the inertia about the center of the mass $J = 0.006kg \cdot m^2$. The viscous friction is given by $\rho_v = 0.53N \cdot m \cdot s/rad$ and the Coulomb friction as $\rho_c = 0.05492N \cdot m$. The initial conditions for the pendulum, selected for all experiments, are fixed as $\theta(0) = \pi\ rad$ and $\dot{\theta}(0) = 0\ rad/seg$ for the position and velocity, respectively.

Case I. Figure 2 and figure 3 shows the dynamics of one-link pendulum system in closed loop affected by the bounded internal perturbations due to the design of the controllers. The numeric exercise use the constants $\alpha = 1$, $\beta = 0.5$, as gains for twisting algorithm and for the DGT controller the values $\alpha = 0.5$, $\beta = 0.25$ are considered. The applied torque to one-link pendulum is shown in figure 4. In figure 5, the behavior of the dynamic gain of the new proposal is presented.

Case II. Figure 6 shows the dynamics of one-link pendulum system in closed loop affected by the bounded internal perturbations due to the design of the controllers. The

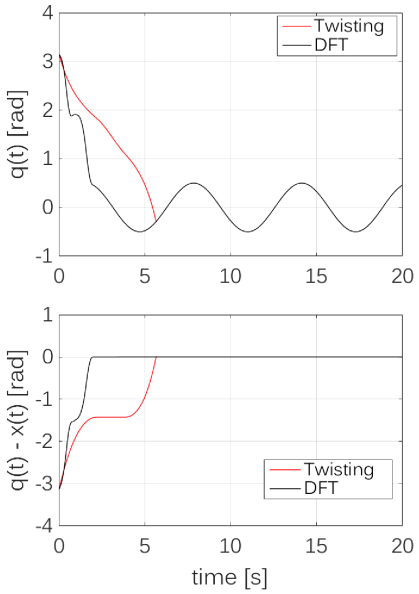


Figure 2. Tracking stabilization of the one-link pendulum (position).

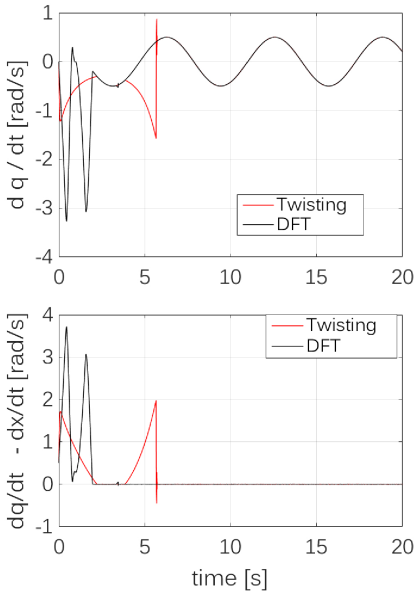


Figure 3. Tracking stabilization of the one-link pendulum (velocity).

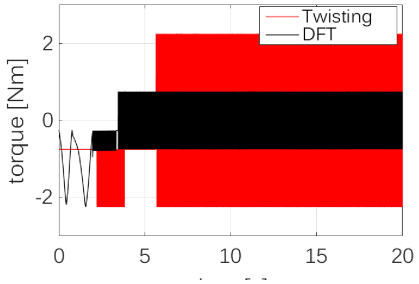


Figure 4. Applied torque to one-link pendulum.

numeric exercise use the constants $\alpha = 2$, $\beta = 1$, for twisting algorithm and for the DGT controller, the values

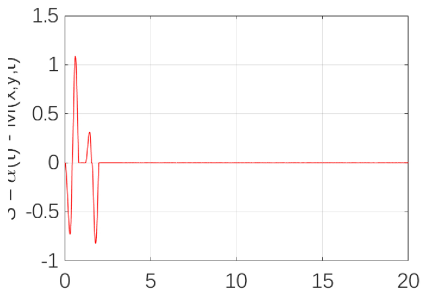


Figure 5. Dynamic Gain of DGT controller.

$\alpha = 1$, $\beta = 0.5$ are considered. The applied torque to the one-link pendulum and the behavior of the dynamic gain of the new proposal is shown in figure 7.

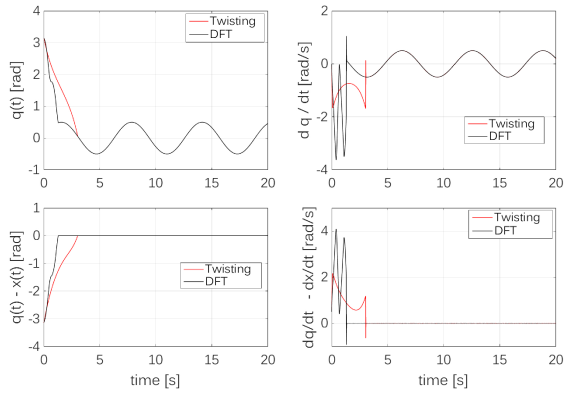


Figure 6. Tracking stabilization of the one-link pendulum.

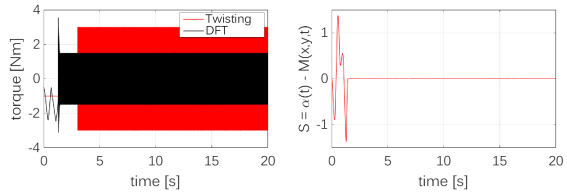


Figure 7. Applied torque and Dynamic Gain of DGT controller.

Case III. Figure 8 shows the dynamics of one-link pendulum system in closed loop affected by the bounded internal perturbations due to the design of the controllers. The numeric exercise use the constants $\alpha = 2.5$, $\beta = 1.25$, for twisting algorithm and for the DGT controller, the values $\alpha = 1.5$, $\beta = 0.75$ are considered. The applied torque to the one-link pendulum and the behavior of the dynamic gain of the new proposal is shown in figure 9.

Case IV. In figure 10 and figure 11, another experiment is shown considering parametric perturbations in the DGT controller, where $M(x, y, t) = 4sin(2q)$. It can be easily appreciated that the DGT controller has a nice performance in spite of parametric uncertainties.

5. DISCUSSION

In the last section, four experiments were presented using one-link pendulum as a test bed. The well performance of the new proposal is shown in spite of unknown but

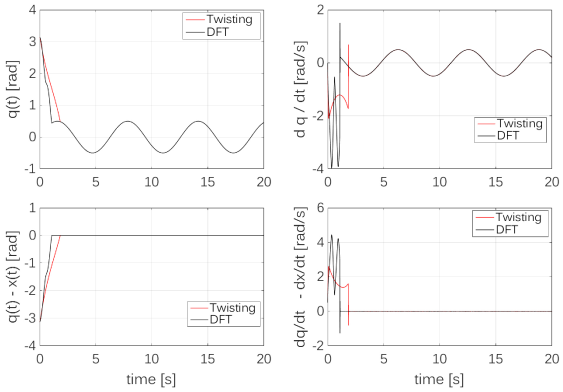


Figure 8. Tracking stabilization of the one-link pendulum.

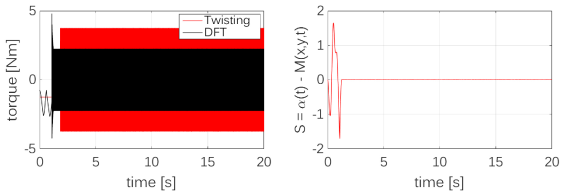


Figure 9. Applied torque and Dynamic Gain of DGT controller.

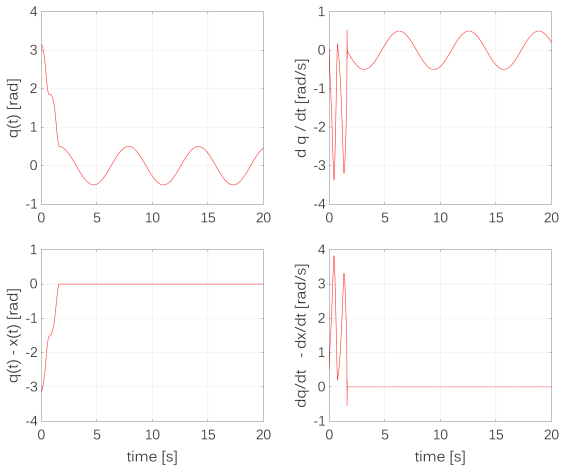


Figure 10. Tracking stabilization of the one-link pendulum (parametric perturbations).

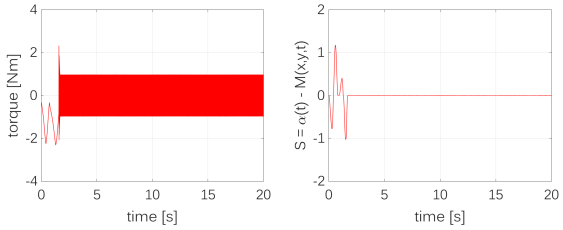


Figure 11. Applied torque and Dynamic Gain of DGT controller (parametric perturbations).

bounded uncertainties. Moreover, it is clear that the chattering phenomenon is reduced using an algorithm with adaptive gain.

As it can be seen, the chattering phenomenon depends on the gain of the controller. Note that using a positive

constant as an upper bound for the perturbation is better for implementation purposes. However, the chattering phenomenon will be present at all time. Considering an variable gain, *i.e.* a function as an upper bound for the uncertainty, vanishing perturbations could be compensated using variable gain.

6. CONCLUSIONS

Finite time Stability of the double integrator affected by bounded external perturbations is shown, using Twisting control synthesis using a dynamic algorithm as variable gain. With this aim a nonsmooth strict Lyapunov functions is proposed in order to show the stability of the nominal closed loop system. The performance of the algorithms were shown by numerical simulations, in spite of bounded external and parametric perturbations. Indeed, a one-link pendulum is considered as a test bed. The closed loop system showed to be robust and provide nice performance in spite of unknown but bounded uncertainties.

REFERENCES

A. Levant, M.T. and Plestan, F. (2011). Twisting controller gain adaptation. *50th Conference on Decision and Control and European Control Conference Proceedings*, 1, 7015–7020.

Bhat, S. and Bernstein, D. (1998). Continuous finite-time stabilization of the translational and rotationnal double integrator. *IEEE Trans. Autom. Control*, 43, 678–682.

Boiko, I. and Fridman, L. (2005). Analysis of chattering in continuous sliding-mode controllers. *IEEE Transaction on Autonatic Control*, 50(9), 1442–1446.

Edwards, C. and Shtessel, Y. (2014). Adaptive continuous higher order sliding mode control. *19th World Congress The International Federation of Automatic Control*, 10826–10831.

F. Plestan, Y. Shtessel, V.B. and Poznyak, A. (2010). New methodologes for adaptive sliding mode control. *International Journal of Control*, 83(9), 1907–1919.

Filippov, A.F. (1988). *Differential equations with discontinuous right-hand sides.* Kluwer Academic Publisher, Dordrecht.

Fridman, L. (2003). Chattering analysis in sliding mode systems with inertial sensors. *International Journal of Control*, 76(9-10), 906–912.

Fridman, L. and Levant, A. (2002). *Higher order sliding modes. In Sliding mode control in engineering.* Control Engineering Series. London: CRC Press.

G. Bartolini, A. Ferrara, E.U. and Utkin, V. (2000). On multi-input chattering-free second order sliding mode control. *IEEE Transactions on Automatic Control*, 45(9), 1711–1717.

H. Sira-Ramirez, M.R.N. and Rodriguez-Angeles, A. (2010). On the linear control of nonlinear mechanical systems. *49th IEEE Conference on Decision and Control (CDC)*, 1999–2004.

Lee, H. and Utkin, V.I. (2007). Chattering suppresion methods in sliding mode control systems. *Annual Revies in Control*, 31, 179–188.

Levant, A. (2007). Principles of 2-sliding mode design. *Automatica*, 43(4), 576–586.

M. Dal, R.T. (2011). Sliding mode controller gain adaptation and chattering reduction techniques for dsp-based pm dc motor drives. *Turk J. Electric Engineering and Computational Science*, 4, 531–549.

Orlov, Y. (2005). Finite-time stability and robust control synthesis of uncertain switched systems. *SIAM Journal on Control and Optimization*, 43, 1253–1271.

Orlov, Y. (2008). *Discontinuous Systems: Lyapunov Analysis and Robust Synthesis under Uncertainty Conditions.* Communications and Control Engineering. Springer, 2009 edition.

Orlov, Y., Alavarez, J., Acho, L., and Aguilar, L. (2003). Global position regulation of friction manipulators via switched chattering control. *Inter. J. control*, 76, 1446–1452.

Orlov, Y., Aoustin, Y., and Chevallereau, C. (2011). Finite time stabilization of a perturbed double integratorpart i: Continuous sliding mode-based output feedback synthesis. *IEEE Transactions on Automatic Control*, 56(3), 614–618.

Qin, G. and Duan, Z. (2012). Output chattering attenuation between two tracking controllers. *International Journal of Control, Automation and Systems*, 10(3), 651–658.

S. V. Emelyanov, S.K.K. and Levantovsky, L.V. (1986). Second order sliding modes in controlling uncertain systems. *Soviet Journal of Compute and System Science*, 24(4), 63–68.

Shtessel, Y.B., Shkolnikov, I.A., and Levant, A. (2007). Smooth second-order sliding modes: Missile guidance application. *Automatica*, 43, 1470–1476.

T. Gonzales, J.A.M. and Fridman, L. (2012). Variable gain super-twisting sliding mode control. *IEEE Transactions on Automatic Control*, 57(8), 2856–2861.

Y. J. Huang, T.C.K. and Chang, S.H. (2008). Adaptive sliding mode for nonlinear systems with uncertain parameters. *IEEE Transactions on Systems, Man and Cybernetics*, 38(2), 534–539.

Advanced Control of a fed-batch reaction system to increase the productivity in the polyhydroxyalkanoates production process.

Cesar García*, Alejandro Acosta** and Silvia Ochoa***

*SIDCOP Research Group, University of Antioquia-Colombia (tel: +57 2198483; cesar.garciae@udea.edu.co).
**Biotransformación Research Group, University of Antioquia-Colombia (tel: +57 2198483; alejandro.acosta@udea.edu.co).
***SIDCOP Research Group, University of Antioquia-Colombia (tel: +57 2198568; silvia.ochoa@udea.edu.co).

In this work, the optimizing control of the fed-batch process for the production of Polyhydroxyalkanoates is carried out by formulating and solving a dynamic optimization problem in order to maximize the process productivity. The optimization problem is subject to constraints on the feed flow rates, the final volume and the maximum concentrations able to be reached on the substrate and nitrogen-source in order to avoid inhibition. For solving the dynamic optimization problem, different parameterization strategies of the control vector were used in order to compare their effect on the dynamic behavior of the biological variables, and therefore, on the process productivity. Results have shown that sinusoidal parameterization of the control profiles lead to higher productivity values while avoiding abrupt changes on the microorganism environment. Furthermore, it is shown that coupling the optimizing control to a neural network soft-sensor developed for predicting the number average molecular weight of the biopolymer is a good strategy in order to fulfill the end-product specifications. In this case, the optimal solution leads to a productivity over 300 g, while keeping the number average molecular weight at common reported values for important applications (between $4x10^5$ - $2x10^6$ g/mol).

Keywords: Advanced Control, Optimization, State Estimation, Biotechnology, Simulation.

1. INTRODUCTION

Polyhydroxyalkanoates (PHAs) are polymers from biological origins, which are currently claimed to be an environmentally friendly option for replacing petroleum based plastic materials in a wide number of applications. However, production costs of these plastic materials are still higher than the petroleum based ones, which has prevented the expansion of the biopolymer industry, despite the fact of its innumerable environmental advantages. Therefore, in the last years, scientists have put many effort in improving the technical and economic feasibility of the process. Some of these works focused on using alternative low cost substrates (Lee & Na 2013), (Dietrich, Illman & Crooks 2013). Other works have focused on using tools from the Process Systems Engineering (PSE), in order to address the optimization and control of the process, towards increasing its productivity (Keshavarz & Roy 2010), (Khanna & Srivastava 2006) and (López , Bucalá & Villar 2010).

In this work, it is proposed to maximize the productivity of the PHA's production process by applying an optimizing control strategy based on a first principle model containing the most relevant dynamics for the process (substrates, biomass, polymer, dissolved Carbon Dioxide and Dissolved Oxygen). It is also expected to include characteristics of the desired product such as molecular weight distribution (MWD), the Number Average Molecular Weight (Mn), the Weight Average Molecular Weight (Mw) and/or the Polydispersity Index (PDI). Due to the importance of assuring some of these characteristics, the optimizing control strategy proposed in this work, is coupled to a soft-sensor developed for predicting the Number Average Molecular Weight (Mn).

2. MODELLING OF THE PHA's PROCESS

The developed model is based on the work by (Shahhosseini 2004), (Khanna & Srivastava 2006), (Amicarelli et al. 2008) and (Chatzidoukasa, Penlogloub & Kiparissides 2013). The model is described by equations (1-8), where the dynamic equations describing the behavior of the biomass (X), substrate (S), biopolymer (P), nitrogen-source (N), dissolved oxygen (O_2L) and dissolved carbon dioxide (CO_2) concentrations are given. The specific growth rate (μ) depends on the glucose concentration (S), nitrogen-source concentration (N) and Dissolved Oxygen concentration (O_2L), following a sigmoidal relationship. F_1, F_2 and F_3 correspond to the feed flow rates of the substrate, nitrogen-source and oxygen, respectively. S_{in}, N_{in} and O_{in} correspond to the concentrations of substrate, nitrogen-source and oxygen in the feed. Finally, V is the fermentation volume.

$$\mu = \mu_m \left(\frac{\left(\frac{N}{S}\right)}{\left(\frac{N}{S}\right)+K_{sr}} \right) * \left(1 - \left(\frac{\frac{N}{S}}{S_m} \right)^{n_k} \right) * \left(\left(\frac{O_2L}{K_{ox}*X+O_2L} \right) \right) \quad (1)$$

$$\dot{X} = \mu X - \frac{F_1+F_2}{V} X \quad (2)$$

$$\dot{S} = -\left(\left(C_{sx}\, \mu X \right) + \left(R_{csx}\, X \right) + C_{sp}\left((K_1 \mu X) + (K_2 X) \right) \right) + \frac{F_1}{V} S_{in} - \frac{F_1+F_2}{V} S \quad (3)$$

$$\dot{P} = (K_1 \mu X) + (K_2 X) - \frac{F_1+F_2}{V} P \quad (4)$$

$$\dot{N} = -\left((C_{nx}\ \mu\ X) + (\dot{R_{cnx}}\ X)\right) + \frac{F_2}{V}N_{in} - \frac{F_1+F_2}{V}N \quad (5)$$

$$\dot{O_2}L = ((KL(O_{2Leq} - O_2L)) - (K_3\mu X) - ((K_4K_1\mu X) + (K_4K_2X)))) + \frac{F_3}{V}O_{in} - \frac{F_1+F_2}{V}O_2L \quad (6)$$

$$\dot{CO_2} = (\alpha_1\mu + \alpha_2)X + \alpha_3 - \frac{F_1+F_2}{V}CO_2 \quad (7)$$

$$\dot{V} = F_1 + F_2 \quad (8)$$

The model contains 19 parameters. For identifying those parameters, a hybrid strategy combining the simulated annealing and the interior point method was used. The objective function for parameter identification is given by (9).

$$\text{SSWR} = \sum_{i=1}^{n}\sum_{j=1}^{m}\frac{\Delta_{ij}}{W_j^2} \quad (9)$$

Where SSWR represents the sum of the square weighed residuals, n and 'm' are the total number of experimental data points and variables, respectively. W_j is a normalization factor for each variable. Δ_{ij} is the difference between the predicted and experimental data. Experimental data for identification and validation was taken from (Khanna & Srivastava 2005). For finding an optimal set of parameters, the methodology by (Wu et al. 2013) was applied. Results showed that only 7 parameters are identifiable. Therefore, a re-optimization strategy was performed for finding a better value for these sensitive parameters while keeping fixed the remaining 12 parameters. Figure 1 shows the model results after parameter identification. It can be observed that model predictions are in good consent with experimental data.

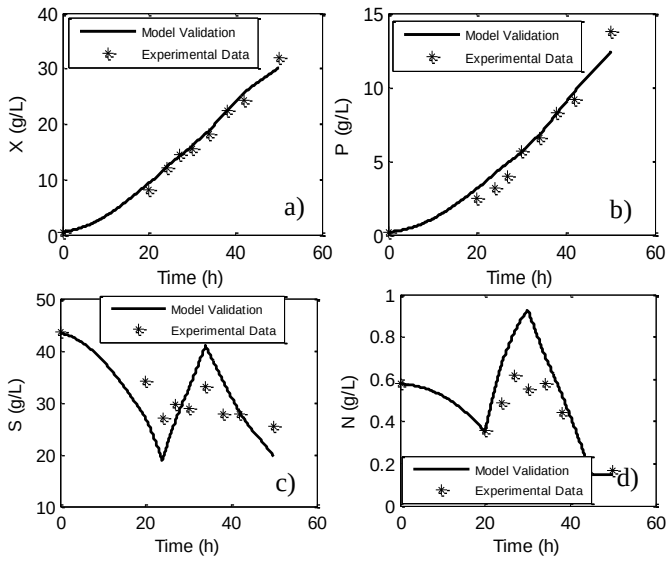


Figure 1. Model validation for fed-batch PHA production. Results for the main state variables: a) Biomass, b) Bio-polymer, c) Substrate, d) Nitrogen Source.

3. SOFT–SENSOR DEVELOPMENT FOR MOLECULAR WEIGHT DISTRIBUTION

In the case of polymer production, end-product properties are highly related to the Molecular Weight Distribution (MWD) achieved during the process, (Sudesh, Abe & Doi 2000). Therefore, it is important to control the MWD during the process operation. However, it has to be considered that one of the main limitations for controlling is the difficulty of getting cheap and reliable on-line measurements of the MWD. Previous works by (Penloglou et al. 2010) and (Penloglou et al. 2012) proposed the combination of a polymerization and a macroscopic model in order to determine the MWD in the PHA production. However, the mentioned works don´t take into account any control strategy in order to achieve a desired final molecular weight distribution. Based on the mentioned works, a structured macroscopic/polymerization kinetic model was simulated in order to obtain enough "in silico data" for building an Artificial Neural Network (ANN) capable of predicting the number average molecular weight (Mn). Such ANN is used as a soft-sensor inside the optimizing control strategy, in order to monitor the Mn during the process operation and to drive the optimization towards finding the conditions for assuring maximal productivity while keeping the Mn at a pre-established range (i.e. required for the biopolymer to fulfill certain mechanical/performance properties)

For simulating the macroscopic/polymerization kinetic model reported by (Penloglou et al. 2010), (Kumar & Ramkrishna 1996) and (Saliakas et al. 2007), a mathematical technique called fixed pivot was used in order to discretize the set of ordinary differential equations that describe the model. It is important to notice that the integration between the macroscopic and the polymerization models is the most important partof the MWD-prediction.. The polymerization model allows predicting Mn as a function of the monomer production rate ($J_M(t)$). Furthermore, $J_M(t)$ is determined by the way the microorganisms use the available substrate for producing the biopolymer. A simple approach that avoids the use of complex metabolic models) for estimating $J_M(t)$ is to calculate the monomer concentration via the consumption rate of the substrate (Penloglou 2015), which is predicted by the macroscopic model presented in section 2, specifically, by using equation (3).

Figures 2a-2b compare the results obtained for the polymer and substrate concentrations by using the described simulation (Model validation), against actual experimental data reported in (Penloglou et al. 2010). Furthermore, figure 2c presents a comparison between the simulation results for the number average molecular weight (Mn), the "real" experimental data and the results presented (Penloglou et al. 2010). Differences between simulation results at this work and the ones reported in the literature are caused by different kinetic parameters in each case, due to the fact that not all kinetic parameters were reported. Therefore, those missing kinetic parameters (the so-called adjusted parameters in Table 1) were estimated in this work through optimization (i.e. performing a similar procedure than the one explained in section 2 for parameter identification).

Table 1. Parameters of the polymerization model

Parameters	Reported by (Penloglou et al. 2012)	Adjusted Parameters
k_i (h^{-1})	0.62 ±9 x 10^4	0.64x10^4
k_P(h^{-1})	0.46±5x10^5	0.44x10^5
k_t^* (h^{-1})	0.14±1x10^1	0.101x10^1
k_{m1}^*(h^{-1})	0.11±2x10^{-3}	0.114x10^{-3}
k_{m2}(l/mol/h)	0.85±15x10^7	0.75x10^7
k_d^*(h^{-1})	0.83±6x10^2	0.25x10^2
$\frac{Y_M}{F}$	0.35±2x10^{-2}	0.23x10^{-2}

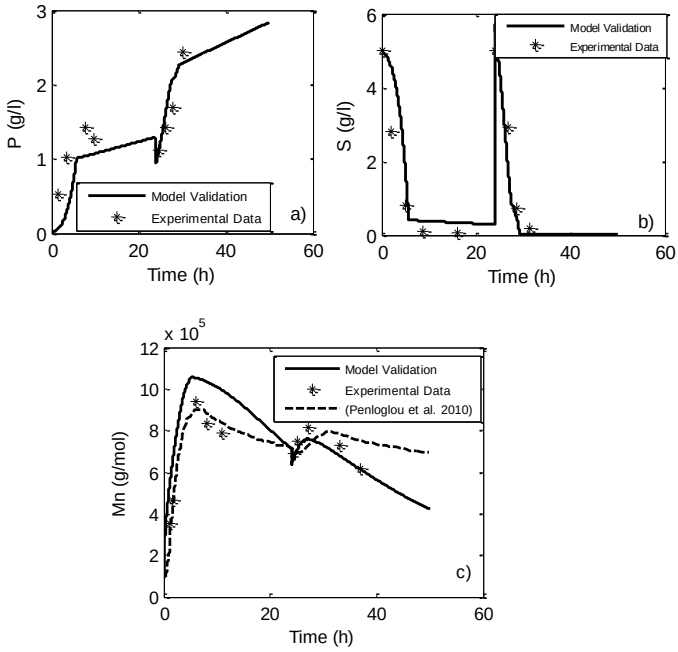


Figure 2. Simulation of the macroscopic/polymerization model: a) Polymer, b) Substrate, c) Number Average Molecular Weight (Mn).

After simulating and validating the mentioned macroscopic/polymerization kinetic model, it was possible to get the required "in silico data" for building an Artificial Neural Network for predicting the Mn. A feed-forward network was built using the following set consisting on input and "measured" variables as an regressor.(i.e. calculated by the first principles model described in section 2): [F_1(t-1); F_1(t-2); F_2(t-1); F_2(t-2); X(t-1); X(t-2); S(t-1);S(t-2); N(t-1); N(t-2); P(t-1); P(t-2); CO_2(t-1); CO_2(t-2)]. Finding such regressor was not a straightforward task. Therefore, the selection was carried out as suggested by (Amicarelli et al. 2014). The network was trained and validated by using respectively, 70% and 30% of the "in silico" data. The network training was carried out by the Levenberg-Marquardt back propagation algorithm. A trial and error approach was used to minimize the error in order to determine the optimal number of hidden layers and neurons. The ANN obtained is composed of 14 neurons at the input layer, and 12 and 1 in the first and second hidden layers, respectively. Results obtained for prediction of Mn by the ANN have shown a very good fit, with a mean absolute Error of 0.85% and R2 of 0.999. Figure 4 compares the ANN predictions against the validation data.

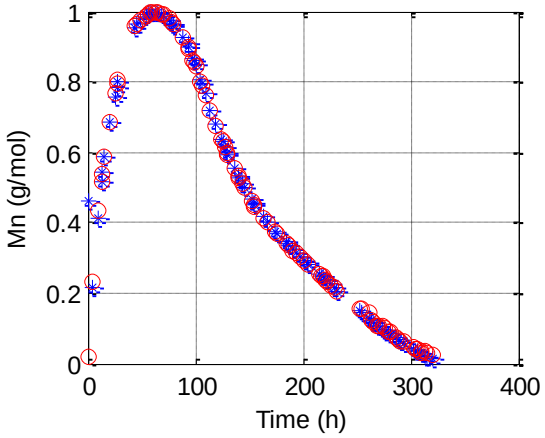


Figure 4. ANN predictions for Mn (red) vs. Validation Data (blue). Normalized data.

4. OPTIMIZING CONTROL

In this section, the development of the optimizing control strategy for maximizing the productivity in a fed-batch PHA process is explained. The objective of the optimizing control strategy is to keep the process operating at maximum productivity, while fulfilling the constraints. In order to assure the end-product specifications for the biopolymer, it is desirable to maintain the average number molecular weight inside a desired range of values. Therefore, the optimizing control is integrated with the ANN state estimator. Figure 5 shows the diagram of the control strategy, which includes the use of the developed ANN and the solution of the dynamic optimization problem through formulating the optimizing control problem. The manipulated variables in this process are the substrate and nitrogen-source feed flow rates.

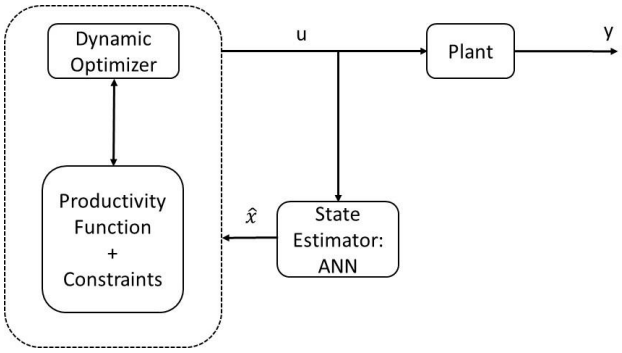


Figure 5. Advanced Control strategy: Optimizing Control + State Estimator

For solving the dynamic optimization problem, the so-called *control vector parameterization approach* was used (Banga et al. 2005). It is important to notice that different parameterizations of the control vector can be usedhaving different impact on the state variables, and therefore on the productivity. For this reason, four different feeding strategies were compared in this work: i) constant feed flow, ii) single

pulse, iii) piecewise constant, and iv) sinusoidal parameterization. .

For the piecewise constant parameterization, the feed flow rates are described by equations (10-11).

$$F_{1,2} = \sum_{j=1}^{m} a_{iok}\varphi(t_{i-1}, t_i)(u_{max} - u_{min}) + u_{min} \quad (10)$$

$$\varphi(t_{i-1}, t_i) = \begin{cases} 0, & t < t_{i-1} \\ 1, & t_{i-1} \le t < t_i \\ 0, & t \ge t_i \end{cases} \quad (11)$$

Where m=12 is the number of steps. u_{max}=0.3 and u_{min}=0 correspond to the maximum and minimum values for each step. The a_{iok} is the parameter that defines the control vector profile, and is therefore the decision variable of the dynamic optimization problem. For this specific case, this type of parameterization uses 12 parameters for each flow rate.

On the other hand, the sinusoidal parameterization of the feed flow rates was implemented as described by equation (12) (Ochoa 2016).

$$F_{1,2} = a_o + a_1 Cos\left(w_1\left(\frac{t-t_0}{t_f-t_0}\right) + \emptyset_1\right) + a_2 Cos\left(w_2\left(\frac{t-t_0}{t_f-t_0}\right) + \emptyset_2\right) \quad (12)$$

Where w_1, w_2 are the frequency, and $\emptyset_1$ and $\emptyset_2$ are the phase angle of the sinusoidal profile. For this specific case, this type of parameterization uses seven parameters for each flow rate.

The dynamic optimization problem is described in equation (15):

$$\underset{F_1(t), F_2(t),\ S_{in}, N_{in}}{Maximize} \ (P(t_f) * V(t_f)) \quad (15)$$

s.to. $F_1 \geq 0\left(\frac{L}{h}\right)$ (15a)

$0 \le F_2 \le 2\left(\frac{L}{h}\right)$ (15b)

$\max\left(S(t)\right) \le 90.11\left(\frac{g}{L}\right)$ (15c)

$\max\left(N(t)\right) \le 10.11\left(\frac{g}{L}\right)$ (15d)

$S_{in} \le 800\left(\frac{g}{L}\right)$ (15e)

$N_{in} \le 70\ \left(\frac{g}{L}\right)$ (15f)

$V \le 10L$ (15g)

Where t_f is the duration of the process. Constraints (15a-15b) take care of the maximum and minimum values allowable for F_1 and F_2, respectively. Max S and max N, are the maximum substrate and nitrogen-source concentrations allowed during the fermentation (i.e. for avoiding inhibition). S_{in} and N_{in} are the substrate and nitrogen-source concentrations in the feed flows F_1 and F_2, respectively. V is the fermentation volume.

The dynamic optimization problem expressed in (15) was solved at two different scenarios, in order to compare and to show the importance of including end-product specifications as part of the control problem during PHA production. The scenarios are, i) Optimizing control without constraints on Mn, and ii) Optimizing control with constraints on Mn.

4.1 Optimizing control without constraints on Mn

Figures 6 and 7 show the optimal F_1 and F_2 feeding profiles for each parameterization case, obtained by solving the dynamic optimization problem given by equation (15) where no constraints on Mn are used.

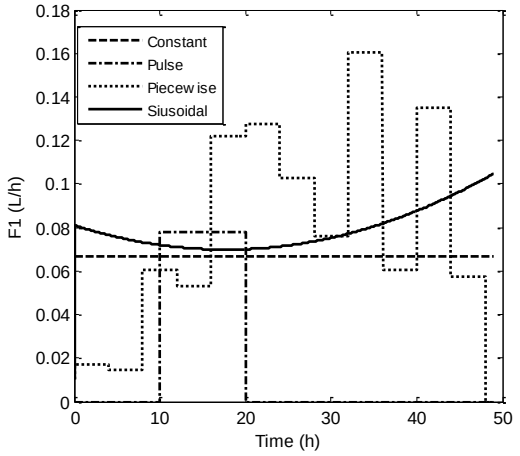


Figure 5. Optimal Feeding profiles for substrate (F_1)

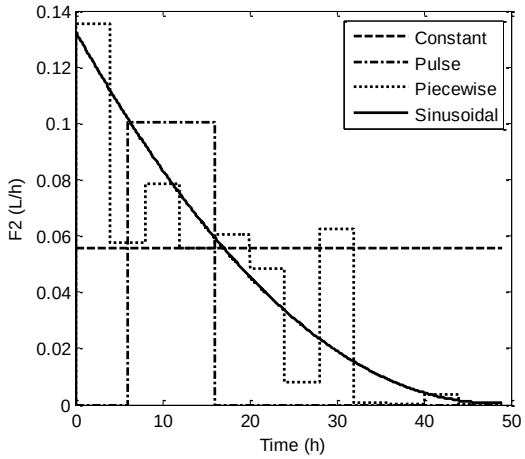


Figure 6. Optimal feeding profiles for nitrogen-source (F_2)

As it can be observed, the sinusoidal parameterization is a smooth strategy (i.e. avoids abrupt and sudden changes in the microorganisms environment), which helps reducing inhibitory effects on the microorganism. Inhibitory effects are more pronounced when the microorganism experiences a rapid increase on the substrate/ nutrient concentrations. Such rapid scenarios are more prompted to take place when step type policies (as for example, when the pulse or piecewise constant feeding policies are followed) are used.

Table 2 shows the productivity reached by applying the four kinds of profiles to the fed-batch PHA production.

Furthermore, the required nitrogen-source and substrate feed concentrations for keeping that productivity are shown. The constant and pulse feeding policies are usually used in practice due to the easiness of implementation. However, as results show in Table 2, these policies resulted in very low productivity values, when compared against the sinusoidal and piecewise constant policies. The sinusoidal feeding policy resulted in the higher productivity, followed by the piecewise constant. Furthermore, the computational time for solving the dynamic optimization by the sinusoidal parameterization was lower for the sinusoidal than for the piecewise constant (results not shown). This is due to the fact that the sinusoidal approach has a lower number of decision variables, which of course impacts the time for reaching an optimal solution. The lowest computational time was reached for the constant feeding policy, and the pulse strategy, respectively. The substrate and nitrogen-source concentrations on the feed are quite similar for all strategies, except for the single pulse.

Table 2. Feeding strategies comparison respect to productivity

Feeding Strategy	Productivity (g)	Nin (g/L) in F2	Sin(g/L) in F1
Constant	364.45	48.66	495.93
Pulse	183.61	38.44	790.59
Piecewise	402.90	43.37	448.52
Sinusoidal	405.17	42.35	463.61

As mentioned, the piecewise and sinusoidal feeding strategies reached higher productivity values by using a similar tendency in the feed rates. However, the abrupt changes on the feed flows calculated by the piecewise strategy could generate cellular stress due to the strong and rapid variations that take place on the microorganism environment, which is reflected on the substrate/nutrients concentrations on the culture media as shown Figure 7. Such stress will definitively affect the performance of the microorganism.

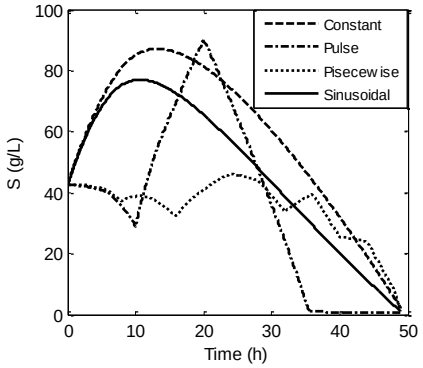


Figure 7. Substrate behavior due to the profiles F_1

Figure 8 shows the predicted number average molecular weight distribution (Mn) by the neural network (described in section 3), when the sinusoidal parameterization is used for the input flows F_1 and F_2 (profiles shown in figures 5 and 6)). As it is shown, the predicted Mn results in a finalvalue of $1x10^5$ if the optimizing control problem doesn´t consider constraints on the Mn which is a low value for industrial and commercial applications. Therefore, in the next section, implementation of the optimizing control including constraints on Mn is addressed.

4.2 Optimizing control coupled with constraints on Mn

Taking into account that the sinusoidal parameterization showed the best results in the previous case, such parameterization was used in the present scenario. The dynamic optimization problem is the same as in equation (15-15g) but including the following constraint on Mn (Gonzales García et al. 2013)

$$400000\left(\frac{g}{mol}\right) \leq Mn(t) \leq 2000000\ \left(\frac{g}{mol}\right) \qquad (16)$$

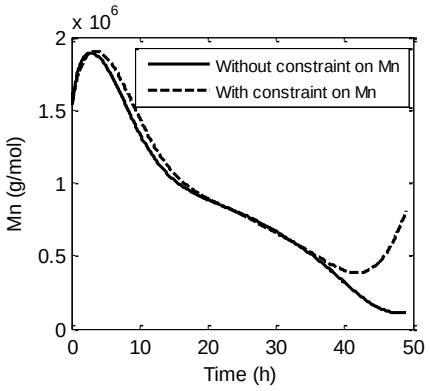


Figure 8. Optimizing control results for Mn: without and with constraint on Mn.

The objective is to keep Mn inside the desired range in order to obtain a polymer with adequate thermoplastic properties. Figure 9 show the feeding profiles that keep Mn at the desired range shown in Figure 8. It is important to notice that the final Mn is $8.07x10^5$ g/mol, which corresponds to values reported in the literature for PHA applications. Finally, the productivity reached was 369.13 g which is lower than in the previous case, but still fulfilling the required constraint on Mn.

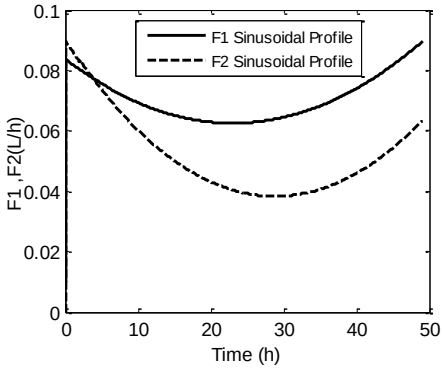


Figure 9. Carbon Source Profile F_1 and Nitrogen Source Profile F_2 with constrint in Mn

4. CONCLUSIONS

Advanced control strategies, as optimizing control, are essential tools that can be implemented in order to achieve maximal productivity and profitability in bioprocess applications. By applying the optimizing control concept coupled to a soft-sensor for the number average molecular weight it was possible to reach high profitability while keeping desired end-product specifications..

Sinusoidal parameterization has shown to provide higher productivity through the use of smooth feeding profiles that are suitable for avoiding cellular stress due to substrate shock. Furthermore, as such parameterization uses a lower number of parameters; the dynamic optimization problem was solved in a faster way.

Further work is now directed towards applying the optimizing control-ANN strategy developed here in a 500L pilot plant, for producing polyhydroxyalkanoates by using a mixture of vinasses/molasses as carbon source.

ACKNOWLEDGMENTS

Financial support from COLCIENCIAS through the project grant 669-2014 is gratefully acknowledged.

REFERENCES

Amicarelli, AN, Quintero, OL, di Sciascio, FA & Álvarez, HD 2008, 'Estudio Comparativo de técnicas para estimación de biomasa en bioproceso tipo batch', *XXIº Congreso Argentino de Control Automático*, Buenos Aires-Argentina.

Amicarelli, A, Quintero , & di Sciascio, 2014, 'Behavior comparison for biomass observers in batch processes', *ASIA-PACIFIC JOURNAL OF CHEMICAL ENGINEERING*, vol 94, pp. 81–92.

Banga, JR, Balsa-Canto, , Moles, CG & Alonso, AA 2005, 'Dynamic optimization of bioprocesses: Efficient and robust numerical strategies', *Journal of Biotechnology*, vol 117, no. 4, pp. 407–419.

Chatzidoukasa, C, Penlogloub, G & Kiparissides, 2013, 'Development of a structured dynamic model for the production of polyhydroxybutyrate (PHB) in Azohydromonas lata cultures', *Biochemical Engineering Journal*, vol 71, pp. 72– 80.

Dietrich, , Illman, & Crooks, 2013, 'Differential sensitivity of polyhydroxyalkanoate producing bacteria to fermentation inhibitors and comparison of polyhydroxybutyrate production from Burkholderia cepacia and Pseudomonas pseudoflava', *BMC Research Notes*, pp. 6-219.

Gonzales García, , MEZA CONTRERAS, JC, GONZÁLEZ REYNOSO, & CÓRDOVA LÓPEZ, JA 2013, 'Síntesis y biodegradación de Polihidroxialcanoatos: Plásticos de Origen Microbiano', *Rev. Int. Contam. Ambie*, vol 29, no. 1, pp. 77-115.

Keshavarz, T & Roy, I 2010, 'Polyhydroxyalkanoates: bioplastics with a green agenda', *Current Opinion in Microbiology*, vol 13, pp. 321-326.

Khanna , S & Srivastava, 2006, 'Optimization of nutrient feed concentration and addition time for production of poly(-hydroxybutyrate', *Enzyme and Microbial Technology 39*, vol 39, pp. 1145–1151.

Khanna, & Srivastava, AK 2005, 'A Simple Structured Mathematical Model for Biopolymer (PHB) Production', *Biotechnology Progress*, vol 21, no. 3, pp. 830-838.

Kumar, S & Ramkrishna, D 1996, 'On the solution of population balance equations by discretization--I. A Fixed Pivot Technique', *Chemical Engineering Science*, vol 51, no. 8, pp. 1311-1332.

Lee, G & Na, J 2013, 'Future of microbial polyesters', *Microbial Cell Factories*, pp. 12-54.

López , A, Bucalá , V & Villar, 2010, 'Application of Dynamic Optimization Techniques for Poly(β-hydroxybutyrate) Production in a Fed-Batch Bioreactor.', *Ind. Eng. Chem. Res.*, vol 49, pp. 1762–1769.

Ochoa, S 2016, 'A new approach for finding smooth optimal feeding profiles in fed-batch fermentations', *Biochemical Engineering Journal* , vol 105 , pp. 177–188.

Penloglou, G 2015, 'Personal Communication'.

Penloglou, , Roussos, A, Chatzidoukas, C & Kiparissides, C 2010, 'A combined metabolic/polymerization:kinetic model on the microbial production of poly(3-hydroxybutyrate)', *New Biotechnology*, vol 27, no. 4, pp. 1-10.

Penloglou, G, Chatzidoukas, C, Kiparissides, & xxx, X 2012, 'Microbial production of polyhydroxybutyrate with tailor-made properties: an integrated modelling approach and experimental validation.', *Biotechnology advances*, vol 1, no. 30, pp. 329-337.

Saliakas, , Chatzidoukas, C, Krallis, , Meimaroglou, & Kiparissides, 2007, 'Dynamic Optimization of Molecular Weight Distribution Using Orthogonal Collocation on Finite Elements and Fixed Pivot Methods: An Experimental and Theoretical Investigation', *Macromolecular Reaction Engineer*, vol 1, pp. 119–136.

Shahhosseini, 2004, 'Simulation and optimisation of PHB production in fed-batch culture of Ralstonia eutropha', *Process Biochemistry*, vol 39, pp. 963–969.

Sudesh, K, Abe, H & Doi, Y 2000, 'Synthesis, structure and properties of polyhydroxyalkanoates: Biological polyesters', *Progress in Polymer Science*, vol 25, pp. 1503-1555.

Wu, W, Lai, SY, Jang, F & Chou, 2013, 'Optimal adaptive control schemes for PHB production in fed-batch fermentation of< i> Ralstonia eutropha', *Journal of Process Control*, vol 8, no. 23, pp. 1159-1168.

An Approach of a Numerical Methods Controller for Nonlinear Chemical Processes

Javier Guevara*, Leonardo Guevara*, Oscar Camacho*, Gustavo Scaglia, Andrés Rosales***

** Departamento de Automatización y Control Industrial, Escuela Politécnica Nacional*
Quito, Ecuador (e-mail: dario.guevara, cesar.guevara01, oscar.camacho, andres.rosales}@epn.edu.ec)
*** Instituto de Ingeniería Química, CONICET, Universidad Nacional de San Juan*
San Juan, Argentina (e-mail: gscaglia@unsj.edu.ar)

Abstract: In this paper is presented an approach of Numerical Methods Controller (NMCr) based on an empirical linear model of the processes. The controller is developed for self-regulating processes with an open loop response similar as a first order plus dead time (FOPDT) model, and can be tuned using the characteristic parameters taken from the reaction curve. The performance of the proposed controller is tested in two nonlinear chemical processes and the results are compared, by simulations, against a PID controller using the ISE performance index to measure it.

Keywords: numerical methods, chemical processes, nonlinear processes, model approximation, FOPDT, characteristic parameters.

1. INTRODUCTION

The methodology based on linear algebra and numerical methods concepts to design control algorithms is a simple and relatively new method that allows the control of highly nonlinear systems. It have been applied to the design of different class of control systems such as trajectory tracking of mobile robots and UAV's (Scaglia, G. et al. (2006, 2009), Rosales, A. et al. (2010), Guerrero, F. et al. (2013)), chemical plants and bioprocesses (Quintero, O. L. et al. (2008), Scaglia, G. et al. (2014), Suvire, R. et al. (2013), Godoy, S. et al. (2013)) just to name some applications. Its main advantage is that the conditions for the tracking error tends to zero and control actions are obtained with low computational cost which makes it easy to implement in a microcontroller. These conditions are found by solving a system of linear equations and finding the conditions for the system to have exact solution.

For all the previous applications, the controllers were designed based on the complete process model. Therefore, for each different process a different control law is produced and its application is just for the process under analysis. Besides, the development of a complete model for industrial processes is difficult mainly due to the complexity of the process, the lack of knowledge of some process parameters and the possible higher order of the manipulated variables. Therefore, the methodology of numerical controller method using complete models can produce more complex controllers. An efficient alternative modelling method for process control is the use of empirical models, most times FOPDT models can work satisfactory for analysis and design of process control (Camacho O. and Smith C, 2000).

In a previous work, Guevara, L. et al. (2016) presented an approach of numerical methods controller (NMCr) based on a FOPDT model, also tuning equations are presented based on the characteristic parameters of the process, which are obtained from the reaction curve method. The results obtained, using the proposed approach, in several higher order linear systems showed a good performance.

This work shows the application of the proposed Numerical Method Controller (NMCr), based on a FOPDT model of the actual process, to nonlinear chemical processes. The performance and robustness of the proposed controller are tested through simulations in two nonlinear chemical processes. The first process is a mixing tank with variable dead time, and the second process is a continuous stirred tank reactor. In both cases the results are compared against a PID controller.

This paper is organized as follows: Section II presents the basic concepts required to controller design, section III presents the development of the proposed controller. Then in section IV the simulation results are presented divided in two cases of study and finally conclusions and future works are presented in section V.

2. BASIC CONCEPTS

2.1 First Order Plus Dead Time System (FOPDT)

Introducing a step change at the process input and then recording the transmitter output, produces a reaction curve that represent the dynamic behaviour of actual process (Seborg, D. E. et al. (2011)). The typical open loop response is overdamped as is shown in Fig. 1, and it is called a FOPDT system. The transfer function of a First Order Plus Dead Time system is:

$$G(s) = \frac{y(s)}{u(s)} = \frac{k\, e^{-t_0 s}}{\tau s + 1} \qquad (1)$$

Where, k is the gain of the process, τ is the time constant of the process and t_0 is the dead time or delay.

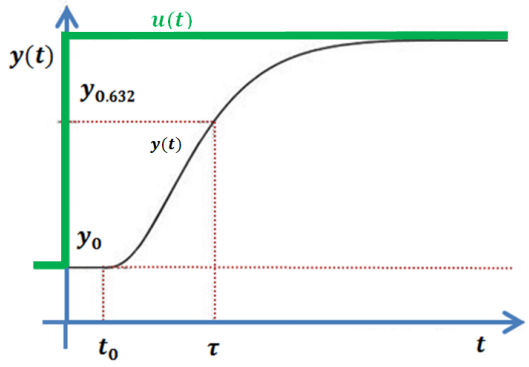


Fig. 1. Typical Response FOPDT.

2.2 Taylor Approximation for Dead Time

Dead time is not specified in the methodology based on linear algebra or numerical methods, therefore, an approximation to replace the dead time must be considered. For instance, in this article the Taylor approximation is used, similar to Camacho, O. et al. (1997). Hence, the dead time expression is substituted as follows:

$$e^{-t_0 s} \approx \frac{1}{1+t_0 s} \tag{2}$$

Replacing (2) in (1), it is obtained:

$$G(s) = \frac{k}{(\tau s+1)(t_0 s+1)} \tag{3}$$

2.3 Euler Approximation

This approximation is used for discretization of an integral term, replacing it by a finite difference. Thus, if the slope value is known in a time period T, an estimate value of the next state y_{n+1} could be obtained using a linear approximation (Chaves, E. (2010)).

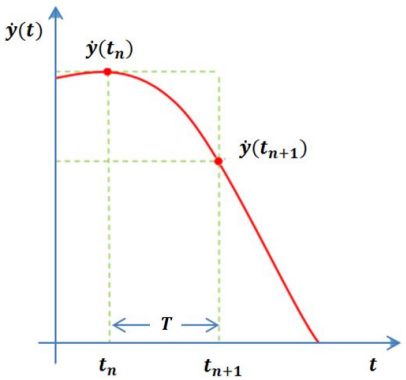


Fig. 2. Euler Approximation.

The derivate approximation is represented by:

$$\dot{y}_n \approx \frac{y_{n+1}-y_n}{T} \tag{4}$$

3. CONTROLLER DESIGN

In this section, it is shown the development of the proposed numerical methods controller based on a FOPDT model. The transfer function with the Taylor approximation is considered for design purposes. Simplifying some terms in (3) and reorganizing for ease of calculation, two parameters (K_A and K_B) are included, they contain the characteristic parameters of the system (t_0 and τ). Consecuently, The transfer function can be represented as follows:

$$G(s) = \frac{y(s)}{u(s)} = \frac{k\,K_B}{s^2+K_A s+K_B} \tag{5}$$

Where:

$$K_A = \frac{t_0+\tau}{t_0\,\tau}[=][\text{time}]^{-1} \tag{6}$$

$$K_B = \frac{1}{t_0\tau}[=][\text{time}]^{-2} \tag{7}$$

Then, solving (5) a second order differential equation is obtained:

$$\ddot{y} + K_A\,\dot{y} + K_B y = k K_B\,u \tag{8}$$

Representing (8) in matrix form, it is found:

$$\begin{bmatrix}\dot{y}_1\\ \dot{y}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ -K_B & -K_A\end{bmatrix}\begin{bmatrix}y_1\\ y_2\end{bmatrix} + \begin{bmatrix}0\\ k\,K_B\end{bmatrix}[u] \tag{9}$$

Where, y_1 and y_2 are the state variables of the system and u the controller output. Then, the system can be described by the following differential equations:

$$\dot{y}_1 = y_2 \tag{10}$$

$$K\,K_B\,u = \dot{y}_2 + K_A\,y_2 + K_B\,y_1 \tag{11}$$

Using the Euler approximation in the previous equations, the following equations are obtained:

$$y_{2\,n} = \frac{y_{1\,n+1}-y_{1\,n}}{T} \tag{12}$$

$$K\,K_B\,u_n = \frac{y_{2\,n+1}-y_{2\,n}}{T} + K_A\,y_{2\,n} + K_B\,y_{1\,n} \tag{13}$$

To make the variation of the error decrease slowly, a first tuning parameter to calibrate the response of the controller is added, it is named K_x and takes values from 0 to 1. If a faster response is required, the value of K_x should be close to 0 and if a slower response is required, the value should be close to 1. This tuning parameter is introduced in (12), and it affects the difference between reference value and output value, the error. It can be considered as a proportional tuning parameter. The resulting expression is:

$$y_{2\,n} = \frac{y_{1\,ref\,n+1}-y_{1\,n}-K_x(y_{1\,ref\,n}-y_{1\,n})}{T} \tag{14}$$

Where, $y_{1\,ref\,n}$ is the reference and $y_{1\,n}$ is the measured output. To reduce the resulting expressions, the difference between the reference $y_{1\,ref\,n}$ and the actual $y_{1\,n}$ can be represented by e_n.

To ensure that the controller compensates disturbances, an integral term is added, therefore it eliminates the steady state error, it is called (NMCr+I). This integral term is formed by

the integral of error multiplied by the second tuning parameter K_i. For implement it, the integral term is represent by:

$$Integral\ term = K_i\ T\ e_{T\ n} \tag{15}$$

Where, $e_{T\ n}$ is the total error accumulated represented by (16) and T is the sampling period:

$$e_{T\ n} = e_{\ n} + e_{T\ n-1} \tag{16}$$

Adding this integral term into (14), the following equation is obtained:

$$y_{2\ n} = \frac{y_{1\ ref\ n+1} - y_{1\ n} - K_x e_{\ n}}{T} + K_i\ T\ e_{T\ n} \tag{17}$$

Assuming that the current value of variable $y_{2\ n}$ is the same required for a next state $y_{2\ n+1}$. Then, replacing (16) in (13) and solving the output controller $u_{\ n}$, the control law is obtained:

$$u_{\ n} = \frac{1}{k\ K_B\ T^2}[y_{1\ ref\ n+1} - y_{1\ n} + K_x e_{\ n} - T\ y_{2\ n} + T^2 K_i e_{T\ n}] + \frac{y_{1\ n}}{k} + \frac{K_A y_{2\ n}}{k K_B} \tag{18}$$

Where:

$$y_{2\ n} = \frac{y_{1\ n} - y_{1\ n-1}}{T} \tag{19}$$

Equation (19) represents the derivative term of the controlled variable. The sampling time value (T) is recommended to be in the range τ /10 < T < τ /4.

The discrete approximation of the derivative, works well if the process does not present noise, but if noise is present, it is necessary to add a low-pass filter, as is done in a PID controller when the derivative term is present.

The integral term eliminates the steady state error, but a higher value of K_i produces a more oscillatory response. Therefore, it should be found a relationship between the two tuning parameters. Therefore, several tests in systems with different characteristics parameters were done, and minimizing the Integral Squared Error (ISE), a criterion for getting an initial value of K_i was obtained. K_x close to 1 is a good starting value, and substituting it into (20), the integral tuning parameter can be obtained.

$$K_i = \frac{0.2\ K_x}{\tau}\ [=][\text{time}]^{-1} \tag{20}$$

4. STUDY CASES

The proposed controller was tested in two very common chemical processes whose models have already been used in previous works.

4.1 Mixing Tank

The mixing tank is a nonlinear chemical process. The tank receives two streams, a hot stream $W_1(t)$, and a cold stream $W_2(t)$. The outlet temperature is measured at a point 125 ft downstream from the tank. The complete model (Camacho, O. and Smith C., 2000), it is described by the following equations.

Energy balance around tank:

$$W_1(t)Cp_1(t)T_1(t) + W_2(t)Cp_2(t)T_2(t) - (W_1(t) + W_2(t))Cp_3(t)T_3(t) = V\rho Cv_3\frac{dT_3(t)}{dt} \tag{21}$$

Pipe delay between the tank and the sensor location:

$$T_4(t) = T_3(t - t_0) \tag{22}$$

Transportation lag (Variable delay time):

$$t_0 = \frac{L\ A\rho}{W_1(t) + W_2(t)} \tag{23}$$

Temperature transmitter:

$$\frac{dTO(t)}{dt} = \frac{1}{\tau_T}\left[\frac{T_4(t) - 100}{100} - TO(t)\right] \tag{24}$$

Valve position:

$$\frac{dV_p(t)}{dt} = \frac{1}{\tau_{V_p}}\left[m(t) - V_p(t)\right] \tag{25}$$

Valve equation:

$$W_2(t) = \frac{500}{60} C_{VL} V_p(t)\sqrt{G_f \Delta P_V} \tag{26}$$

Where:

$W_1(t)$ = mass flow of hot stream, lb/min

$W_2(t)$ = mass flow of cold stream, lb/min

Cp = liquid heat capacity at constant pressure, Btu/lb-°F

Cv = liquid heat capacity at constant volume, Btu/lb-°F

$T_1(t)$ = hot flow temperature, °F

$T_2(t)$ = cold flow temperature, °F

$T_3(t)$ = liquid temperature in the mixing tank, °F

$T_4(t)$ = equal to $T_3(t)$ delayed by t_0, °F

t_0 = dead time or transportation lag, min

ρ = density of the mixing tank contents, lbm/ft^3

V = liquid volume, ft^3

$TO(t)$ = transmitter output signal on a scale from 0 to 1

$Vp(t)$ = valve position, from 0 (closed) to 1 (open)

$m(t)$ = fraction of controller output, from 0 to 1

C_{VL} = valve flow coefficient, gpm/psi$^{1/2}$

G_f = specific gravity, dimensionless

ΔP_V = pressure drop across the valve, psi

τ_T = time constant of the temperature sensor, min

τ_{V_p} = time constant of the actuator, min

A = pipe cross section, ft^2

L = pipe length, ft

The steady-state values and parameters used in this example are presented in Camacho, O. and Smith C., (2000), except L and A which has new values: 80 ft and 0.08 ft^2 respectively.

4.2 Continuous Stirred Tank Reactor (CSTR)

In the reactor, an exothermic reaction takes place ($A \rightarrow B$). The reactor is surrounded by a jacked where a cooling liquid flows to remove the heat of reaction. The process information and steady-state values used in this example were taken from Rojas, R. et al. (2014). The complete model is described by the following equations.

Component balance on reactant A:

$$\frac{dC_A(t)}{dt} = \frac{f}{V}[C_{Ai}(t) - C_A(t)] - r_A(t) \quad (27)$$

Energy balances on the reactor:

$$\frac{dT(t)}{dt} = \frac{f}{V}[T_i(t) - T(t)] - \frac{\Delta H_R}{\rho C_p} r_A(t) - \frac{UA}{V\rho C_p}[T(t) - T_j(t)] \quad (28)$$

Energy balances on the jacket:

$$\frac{dT_j(t)}{dt} = \frac{UA}{V_j \rho_j C_{pj}}[T(t) - T_j(t)] - \frac{F_j(t)}{V_j}[T_j(t) - T_{ji}(t)] \quad (29)$$

The reaction rate for this CSTR:

$$r_A(t) = k_0 e^{-E/RT(t)} C_A(t) \quad (30)$$

The valve equation:

$$F_j(t) = f_{max} \propto^{[u(t)-1]} \quad (31)$$

Where:

$C_A(t)$ = concentration of reactant in the reactor, lbmol/ft^3

$T(t)$ = temperature in the reactor, °R

T_j = temperature of boiling liquid in cooling jacket, °R

$r_A(t)$ = rate of reaction, lbmol/ft^3-min

f = process feed rate, ft^3/min

V = volume of reactor, ft^3

ΔH_R = heat of reaction, Btu/lbmol

ρ = density of reactor contents, lb_m/ft^3

C_p = heat capacity of reactants and products, Btu/lb_m°R

A = heat transfer area, ft^2

U = overall heat-transfer coefficient, Btu/min ft^2 °R

ρ_j = density of the coolant, lb_m/ft^3

C_{pj} = specific heat of the coolant, Btu/lb_m°R

F_j = coolant rate, ft^3/min

f_{max} = maximum flow through the valve, ft^3/min

$\propto$ = valve rangeability parameter, dimensionless

k_0 = Arrhenius frequency parameter, 1/min

E = activation energy of the reaction, Btu/lbmol

R = ideal gas law constant, Btu/lbmol°R

$u(t)$ = valve position, from 0 (closed) to 1 (open)

5. SIMULATION RESULTS

To test the proposed controller, set point changes were made to test tracking references and also disturbances were introduced to prove regulation tasks. The controller performance is measured using the Integral of Squared Error (ISE). The results are compared with a PI controller, tuned using the equations proposed by Dahlin (Smith, C. et al. (1997)), since this control is the most popular alternative in process control.

5.1 Characteristic and tuning parameters

Introducing a step change at the input (valve position) and then recording the transmitter output, the characteristic parameters of the FOPDT models are taken from the result reaction curves and are presented in Table 1.

Table 1. Characteristic Parameters

Process	Parameter			
	k	τ	t_0	t_0/τ
Mixing Tank	0.89	2.29	1.33	0.58
CSTR	1.03	5.55	2.45	0.44

Using these values and formulas for Dahlin synthesis, the tuning parameters for PI controller are obtained. The integral parameter used in NMCr+I is obtained using (20). The tuning parameters of both controllers are presented in Table 2.

Table 2. Controller Tuning Parameters

Process	PI		NMCr+I	
	K_p	K_i	K_x	K_i
Mixing Tank	0.978	0.436	0.8	0.069
CSTR	2.18	0.18	0.8	0.028

5.2 Performance Test

The Fig. 3 shows the $T_4(t)$ response in the mixing tank, when a set point changes from 150 °F to 160 °F is produced, and a disturbance at time 100 min occurs, the hot water flow, $W_1(t)$, changes from 250 lb/min to 260 lb/min. Figure 3 shows that the NMCr+I has a faster response, generating a little more overshoot than PI, and it returns smoothly to the set point value at same time that the PI.

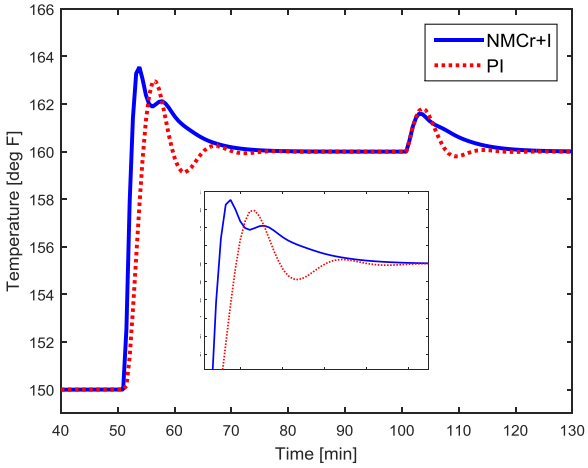


Fig.3 Response for set point and disturbance changes in mixing tank.

The Fig. 4 shows the temperature response $T(t)$ in reactor, when changes from 690 °R to 700 °R occurs and also a disturbance at time 160 min happens when the inlet feed temperature, $T_i(t)$, changes from 578 °R to 583 °R. According with this, the NMCr+I has a faster response with a small overshoot, besides it is less affected by the disturbance.

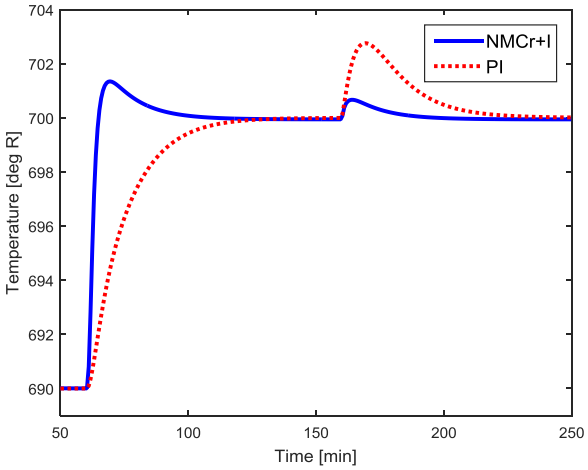


Fig.4 Response for set point and disturbance changes in CSTR.

The results obtained using the performance index ISE are shown in Table 3. In both cases, NMCr+I presented better performance than the PI controller.

Table 3. Results of Performance Index

Index	Controller	Process	
		Mixing Tank	CSTR
ISE	PI	0.0297	0.00188
	NMCr+I	0.0282	0.00036

5.3 Robustness Test

In order to test the robustness of the proposed controller, modelling errors are introduced. The errors used are the same for each one of the characteristic parameters (k, τ and t_0). Figures 5 and 6 depict the resulting ISE curves as a function of time and modelling errors.

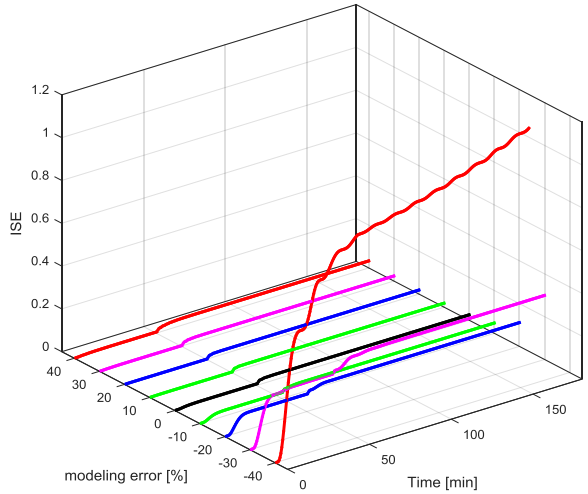


Fig.5 Model error vs ISE for NMCr+I in Mixig Tank.

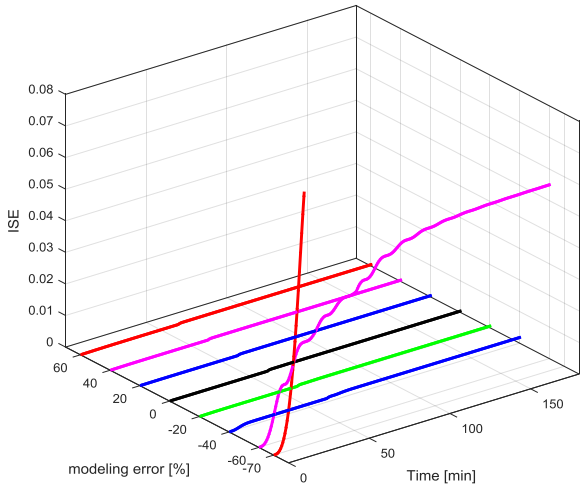


Fig.6 Model error vs ISE for NMCr+I in CSTR.

The errors were increased until the process begins to oscillate and not return to set point value.

Fig. 7 shows the temperature response, using PI and NMCr+I, in the mixing tank when modelling errors are introduced. The first change was introduced at 100 [min] with a -5% model error. The second change was introduced at 150 min with a -20% model error and the last change was introduced at 220 min with a -50% model error.

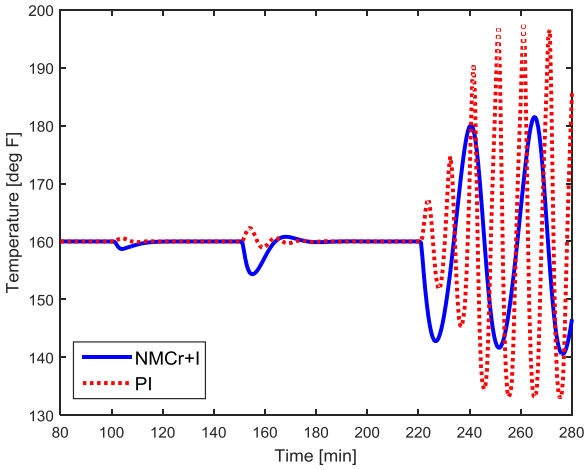


Fig.7 Response for model error changes in mixing tank.

Fig. 8 shows the response of temperature in CSTR similar to the previous case. The first change was introduced at 150 min. with a -8% model error. The second change was introduced at 200 min with a -32% model error and the last change was introduced at 270 min with an excessive -80% model error.

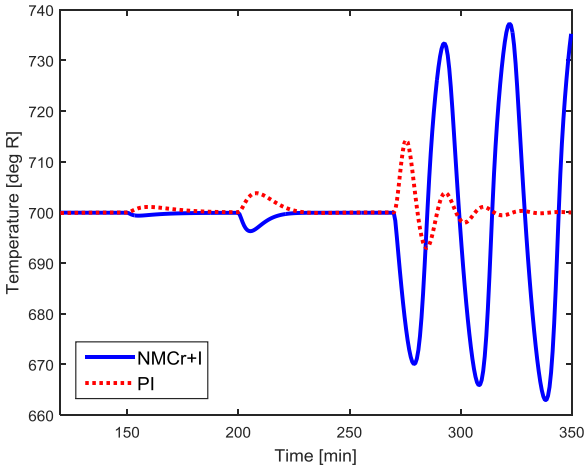


Fig.8 Response for model error changes in CSTR.

The previous results have shown that if modelling errors close to 30% are considered, the proposed control approach is still stable. Therefore, the NMCr+I is robust for reasonable errors.

6. CONCLUSIONS

This work presented the application for nonlinear processes of a Numerical Methods Controller based on a FOPDT model.

The results showed that proposed controller works well for the two different cases of study.

The performance and robustness tests proved that the proposed approach can be considered as a control alternative for nonlinear self-regulating processes.

An optimal tuning procedures, such as computational and intelligent optimization techniques, could be used to get better tunings parameters for the proposed approach.

In future works the proposed approach will be implemented in a real self-regulating open loop processes.

ACKNOWLEDGMENTS

Oscar Camacho thanks PROMETEO project of SENESCYT, Republic of Ecuador, for its sponsorship for the realization of this work.

This work was partially funded by the Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET – National Council for Scientific Research), Argentina.

REFERENCES

Camacho, O., Smith, C. and Chacón, E. (1997). Toward an Implementation of Sliding Mode Control to Chemical Processes. IEEE International Symposium on Industrial Electronics, pp. 1101–1105.

Camacho, O., Smith, C. (2000). Sliding mode control: an approach to regulate nonlinear chemical processes. ISA Transactions, vol. 39, pp. 205-218.

Chaves E. (2010). Integración numérica en el tiempo, 1st ed., España: Universidad de Castilla.

Godoy, S., Serrano, E., Scaglia, G., Ortiz, O. and Secchi, H. (2013). Control de trayectoria basado en Métodos Numéricos aplicado a un birreactor fed-batch. Proceedings of XV Reunión de Trabajo en Procesamiento de la Información y Control.

Guerrero, F. and Menéndez, O. (2013). Modelación, simulación y Control de Sistemas Aéreos no Tripulados utilizando Inteligencia Artificial. Eng. Thesis, Escuela Politécnica Nacional, Quito, Ecuador.

Guevara, L., Guevara, J., Scaglia, G., Camacho, O. and Rosales, A. (2016). A New Approach of a Numerical Methods Controller for Self-Regulating Processes. Proceedings of IEEE Argencon.

Quintero, O. L., Scaglia, G., Amicarelli, A. and Di Sciascio, F. (2008). Bio process control strategy based on numerical methods and linear algebra: second approach. Proceedings of 27th IASTED International Conference on Modelling, Identification and Control, pp. 292-297.

Rojas, R., Camacho, O. and González, L. (2004). A sliding mode control proposal for open-loop unstable processes. ISA Transactions, vol. 43, pp. 243-255.

Rosales, A., Scaglia, G. , Mut V., and Di Sciascio, F. (2010) .Formation control and trajectory tracking of mobile robotic systems – a linear algebra approach. Robotica, vol. 29, No. 3, pp. 335–349.

Scaglia G., Mut, V. and Di Sciascio, F. (2006). Trajectory Control Of Mobile Robots Based On Numerical Methods. Proceedings of XII Congreso Latinoamericano de Control Automátic.

Scaglia, G., Quintero, O. L. , Mut, V.,di Sciascio, F. (2009). Numerical Methods Based Controller Design for Mobile Robots. Robotica, vol. 27, pp. 269-279.

Scaglia G., Aballay, P., Serrano, E., Ortiz, O., Jordan, M and Vallejo, M. (2014). Linear algebra based controller design applied to a bench-scale oenological alcoholic

fermentation. Control Engineering Practice, vol. 25, pp. 66-74.
Seborg, D. E., Edgar, T. F., Mellichamp, D. A., Doyle III, F. J. (2011). Process dynamics and control, 3rd ed. John Wiley & Sons, Inc.
Smith, C. and Corripio, A. (1997). Principles and practice of automatic process control, New York: John Wiley & Sons, Inc.
Suvire, R., Scaglia, G., Serrano, E., Vega, J. and Ortiz, O. (2013). Trajectory Tracking Controller Design Based on Linear Algebra with Integral Accion: Application to CSTR Systems. VII Congreso Argentino de Ingeniería Química (CAIQ.

An Automated Indoor Low-Cost Greenhouse System for Research and Domestic Usage.

Adan Ruiz [**] , Mario Siller** [***] , Ofelia Begovich** [****]

CINVESTAV Campus Guadalajara,
Electrical Engineering and Computer Science,
Zapopan, Jalisco, Mexico,
Av. del Bosque 1145, Col. El Bajio
[**] *(e-mail: aruiz@gdl.cinvestav.mx)*
[***] *(e-mail: msiller@gdl.cinvestav.mx)*
[****] *(e-mail: obegovi@gdl.cinvestav.mx)*

Abstract: This paper presents the design and implementation of an automated control system for a hydroponic testbed for "indoor domestic environments". This testbed is an indoor research system in which automatic control techniques, low-cost computational micro-controllers, and monitoring technologies are being explored for possible applications and deployments in low research budget and limited technology access. The goal of the system is to use automation techniques to improve crop productivity in scenarios in which even water is a scarce resource. The experimental low-cost facility at this point is focusing on control variables such as irrigation time, luminosity, temperature, and relative humidity.

Keywords: aeroponic, hydroponic, automation, control, greenhouse

1. INTRODUCTION

In the recent decades, the population is growing (UN (2004)) and have expanded the necessity to increment the food production, this has led to seek new alternatives for crop development such as indoor greenhouses. An indoor greenhouse is a small structure of glass or plastic containing plants, which can be installed into small spaces such as departments, offices, among others. Indoor farming is a new tendency around the globe, this is going to benefit communities, researchers, makers, hackers, students, any people and also the environment.

For a plant growth, it is necessary to create an environment which must have the appropriate weather conditions. For this reason, this paper presents the design and implementation of an "Automated Indoor Low-Cost Greenhouse (AILCG)" for domestic environments. This experimental low-cost facility at this point is focusing mainly in the best way to control variables such as irrigation time, luminosity, temperature and relative humidity.

There are many irrigation techniques, one of them is Hydroponics which consist in using a mineral-rich solution instead of soil (Jr. (2004)). In Hydroponics the roots obtain all nutrients necessaries for their growing. Besides, there is another irrigation technique called Aeroponic, this technique uses air and mist to provide nutrients to the plant roots, (Parker (2009)). The Hydroponics and Aeroponics irrigation techniques provide water savings, as well provides nutrients to the roots of the plants in an efficient manner, also these techniques may be used in indoor greenhouses. For the AILCG, both irrigation techniques are applied, as it will be seen later.

The research focus is control automation, and the architecture and design of the Information and Communication Technology (ICT) infrastructure. This includes system modeling, computational techniques and algorithms. Previous works for monitoring and control system for greenhouses are described in Putter and Gouws (1996), Bhutada et al. (2005), Schempf et al. (2001). A related project, for indoor farming, is the Personal Food Computer (PFC) which is an agriculture technology platform which is being used around the world (Harper and Siller (2015)).

In this work, there are several control variables considered such as the lighting for plants, temperature, humidity, with this testbed further research may be possible, by example: find better strategies to provide light to the plants.

There is a wide availability of electronic devices that can handle the AILCG sensors data such as micro-controllers, Field Programmable Gate Array (FPGA) devices, digital signal processors, PLC devices, credit card sized computers, among others. For this work, we have chosen to use a microcontroller "Arduino^{TM} UNO based board" due the low-cost, which is near to $20 USD, this is cheaper than a PLC or an FPGA; the community support; the already developed gadgets and libraries; among other advantages cited in Badamasi (2014).

[⋆] The authors would like thanks to CONACYT for the PNPC program, also thanks to Adrian Lizaray, Christian Lopez, Raul Gil and Bernardo Camacho from "Universidad Politecnica de Sinaloa" for their cooperation in this project during their scholar residence program. Finally, we give thanks to Jorge Urbina and Angel Moreno by their revision of this paper.

Micro-controllers are used to recollect data from the sensors, process information (using pre-programmed functions) and provide an output to perform an action (e.g. actuators). In this work, sensors provide information about the environment in which the crops are, this allows to collect data about the different crop phases of experimentation and provide an action through the actuators.

The proposed AILCG considers a "Light-Emitting Diode (LED)" control, the sun photo-period, different irrigation techniques control, water aeration control, and climate condition control to provide the needed tools to perform research and hence obtain a healthy plant growing.

This paper is organized as follows, Section 2 is about the AILCG as the whole, in Section 3 the control system is described, Section 4 results are shown and finally, conclusions are stated.

2. AUTOMATED INDOOR LOW-COST GREENHOUSE

The AILCG is able to cultivate plants and produce food by using 3w LED lights at 660nm (red light) and 445nm (blue light) as an alternative to sunlight (Chang and Chang (2014) and Son and Oh (2013)); a 1/4 HP water pump; several sensors and micro-controllers are used to manipulate information and actuators.

The AILCG control is based on a process of reading, sending and receiving data from the sensors using an Arduino^{TM} UNO in addition with an ethernet shield for internet access. Arduino^{TM} is an open-hardware platform designed to facilitate the use of electronics in multidisciplinary projects.

2.1 System Requirements

Some variables that influence the plant development are temperature, relative humidity, light, irrigation, and water pH.

Temperature. Is beneficial for the plants, have a temperature difference between day and night temperatures (Mohr and Schopfer (2010)), also the solution temperature is important. More information about temperature for lettuce experiments is found in Thompson et al. (1998).

pH Concentration. This value is important for plants to absorb the nutrients, pH is not the same for soil culture (organic substrate) and hydroponics (cultivation in inert substrates). The vast majority of irrigation water must be optimal pH, Domingues et al. (2012).

Ventilation. The greenhouses always should have adequate ventilation, there is a relationship between plants-space and ventilation power. Plant respiration is through their leaves, if ventilation is not adequate, the pores of the leaves will obstruct and the leaves will die. Related works to ventilation can be found in Boulard and Draoui (1995), Muñoz et al. (1999), Casas-Carrillo et al. (2015), among others.

Water aeration. The presence of oxygen in water is essential for the proper development of plant roots. The lack of oxygen is called anoxia, this causes that the root development begins a premature degradation of the roots, reaching, in extreme cases, the death of plants (Mustroph and Albrecht (2003)).

2.2 Description of the AILCG

The Figure 1 shows physical design scheme of the built AILCG as it can be seen, the structure is divided into five different tray-levels, each of them has a purpose, function and/or different works. The tray-levels are Level 0, Level 1, Level 2, Level 3 and Level 4. The AILCG is composed of a rack where there are two different types of hydroponic crops and a piping circuit for recirculating the nutrient solution through the system. At the Level 0

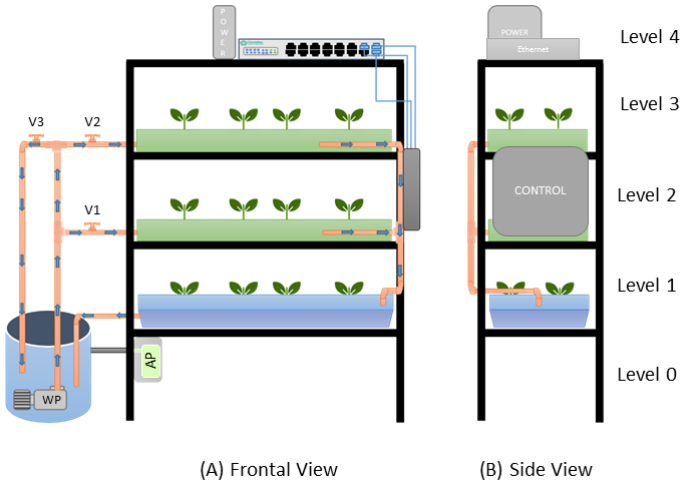


Fig. 1. AILCG Physical Design.

is located the water reservoir, which contains the nutritive solution of the plants, also the air and water pumps are found here, the last one is a submersible water pump. At Level 1 is located the first type of hydroponic technique, this is based on water culture system. At Level 2 is located another container for crop development, using an aeroponic technique, also at this tray-level is located the control box where all the sensor information is collected and is logged into a database server (MySQL database 5.6.23 running on Intel Xeon E5-2403 1.8 GHz processor, 32GB RAM) for the late information processing. At Level 3 is located the last crop, this has the same characteristics as the crop found at the second tray-level.
The power supply is located at Level 4 of the entire AILCG (control system, water and air pumps, lighting system), also there is an ethernet network switch 10/100 which is connected to the internet network, this switch allows the connection to the micro-controllers located in the control system. Those micro-controllers are able to send and receive information over the Internet using an ethernet shield.

2.3 Operation of the System

When the pump is turned on, the solution begins to flow through the pipe system to tray-levels 2 and 3, see Figure 1. The V1 and V2 valves are utilized to regulate the flow of water entering at both tray-levels, while the valve V3

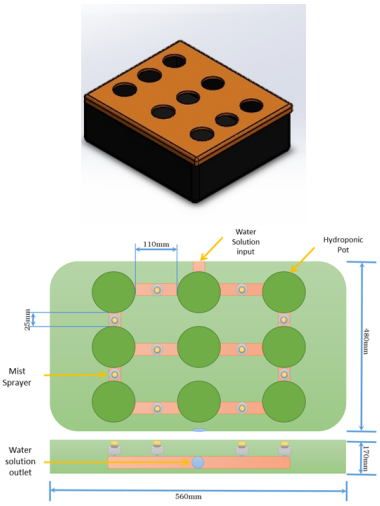


Fig. 2. Physical Design Diagram of Aeroponic Cultivation Bed.

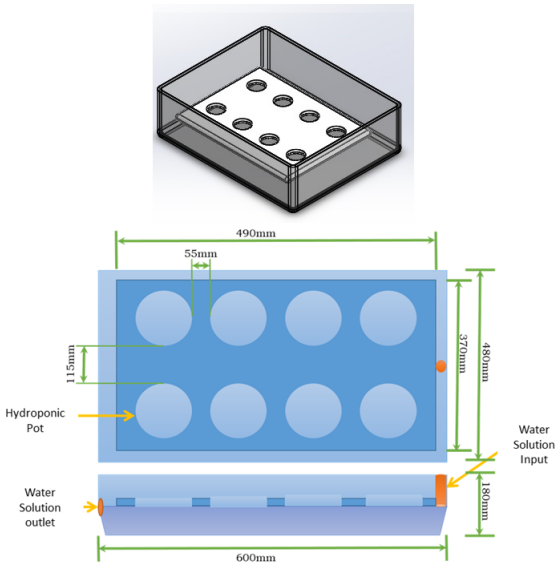


Fig. 3. Physical Design Diagram of Hydroponic Cultivation Bed.

regulates the excess flow to the container.
The nutrient solution is driven by the water pump to tray-levels 2 and 3, where nebulizers are driving out the atomized solution continuously to watering the roots of the plants. The water that is not absorbed by the crop, falls to Level 1 using drain pipes. In order to avoid the water overflow at Level 1, the amount of input water must be slightly less or equal to the amount of the output water, this allows that crops in Level 1 to receive recycled water. Also, the water overflow in Level 1 is driven to the reservoir water. The reservoir water is oxygenated using an air pump, this ensures that the entire crops will receive oxygenated water. Note that, the same cycle repeats, recirculating the water through the piping system, so the loss of solution is reduced.

Aeroponic Cultivation Bed. The cultivation beds, at tray-levels 2 and 3, have ten nebulizers each of one, which are responsible for spraying the solution to the plant roots. The solution flows through the pipes and is delivered to the crop. Also, there is an outlet to recirculate the solution. The aeroponic cultivation bed was made of styrofoam (expanded polystyrene). A sketch of aeroponic cultivation bed is shown in Figure 2.

Hydroponic Cultivation Bed. This cultivation bed, at Level 1, is designed to get the solution of the aeroponic crops (tray-levels 2 and 3) that they did not use. The nutrient solution which leaves from the cultivation bed is renovated in the reservoir where it will be oxygenated again, hence the anoxia in crops is avoided. This kind of hydroponic system is based on water culture system, so cultivation beds are floating on a base of Styrofoam allowing that plant roots are in contact with the nutrient solution every moment. Level 1 it has a transparent polypropylene container. A sketch of hydroponic cultivation bed is shown in Figure 3.

2.4 AILCG Design

The rack has four metal shelves panels and four metal beams, the metal beams forms two upright frames which are joint to the shelves using screws.

Also, recycled materials were used for the current implementation, such as an unused rack, volcanic rocks as a reusable substrate, among other materials. Each tray-levels (Levels 0-3) of the AILCG is controlled by an Arduino^{TM} microcontroller. Then there are four micro-controllers that work separately to keep the system under the required conditions. Thus, being a separate control of each of the tray-levels, if at a certain time the system fails, either by an external or internal factor, the others continue their usual functions for which they were programmed.

3. SENSORS AND CONTROL SYSTEM

The operated sensors in the control system are: "DHT11 and DHT22 (DHTXX)" which provides relative humidity and temperature in the environment, DS18B20 is a waterproof sensor to provide water solution temperature. DS18B20 and DHTXX sensors use the 1-wire protocol, which is a communication protocol designed by Dallas Semiconductor, it is based on a bus, a master and several slaves one data line (Zhang et al. (2010)).

The 1-Wire protocol specifies a master and one or more slaves which send information to a single data bus, a resistor is connected to +5V DC to "pull up" the signal.

3.1 Light Control System

LED lights are handled by a microcontroller, the input power of the LEDs is logged into a database, for this we decided to set an on/off control using a Pulse Width Modulation (PWM). The light period may be adjusted by time or by using a photo-resistance. The time-based light period might be set up from 7:00am to 7:00pm or any other time range, as shown in Figure 5.

On the other hand, given the analog value of the photo-resistance, the microcontroller decides whether to turn on or off the LEDs, this strategy allows the microcontroller perceive the sun photo-period by using a photo-resistance to receive any light beam.

It has been implemented the light control by photoresistance strategy. The photo-resistance has been placed near a window to intercept the solar light beams, due the photoresistance may perceive certain wavelength light beams, such as daylight, fluorescent light or any other lamp light.

To control the LED lights, the PWM duty cycle is influenced by an 8-bit parameter named as "LED Input Power (LIP)", it is defined by the following equation:

$$L(P) = P * 254/1023 \quad (1)$$

The equation is a cross-multiplication as described in Equation 1 where L is the function to obtain the LIP in PWM units and variable P is the measured "Analog Value of a Photoresistance(AVP)". The analog value of atmega328p microcontroller has a resolution of 4.9mV per unit, i.e. for each 5v there are 1024 units, the AVP has a value between 0 and 1023 units, and the LIP value is between 0 and 255 units. Equation 1 may be represented as an analogy: "If there is light there is a relative LED light". But in an absence of light, i.e. stormy days, we have decided to complement the Equation 1 into:

$$G(P) = \begin{cases} 255 & \text{if } L(P) > m \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Equation 2 is represented by the analogy "If there is any light there is a full power LED light", the variable m is the "photo-resistance Analog Value in the absence of Light(AVL)". The AVL is obtained by empirical observation, the experiment measured the amount of light without any kind light, and from a small sample space we determined that the expected value of m is 150, results may change due the photoresistance type, cable resistance, and other external light sources, e.g. street lamps. Finally, $G(P)$ is the LIP value.

3.2 Irrigation Control System

For an indoor plant growing there should be an automated irrigation system (Lieth and Oki (2008)). The actuators influencing directly in the irrigation are at Level 0, they are controlled by a single micro-controller. At the start of the program, the micro-controller initializes the input and output ports, then it sets up the clock and starts the process of irrigation and oxygenation (activate the water pumps and air). If is time to irrigate, the micro-controller activates the water pump using a 5v relay (compatible with 110-240v devices), until the time is up to irrigate. The same technique is used to provide the water aeration. These tasks are running continuously as shown in Figure 4.

3.3 Control System of the Levels 1,2 and 3

At the AILCG, each tray-level from 1 to 3 has a crop an independent monitoring, this means that control in the Level 1 does not interfere with the control of Level 2 and vice-versa, the same for Level 3. Temperature is an important aspect of the cultivation beds. For the temperature control, 12V DC fans are used. Each tray-level of AILCG has two low power fans that eliminate heat excess in the area of crops and lead to the required conditions for optimal plant development into AILCG.

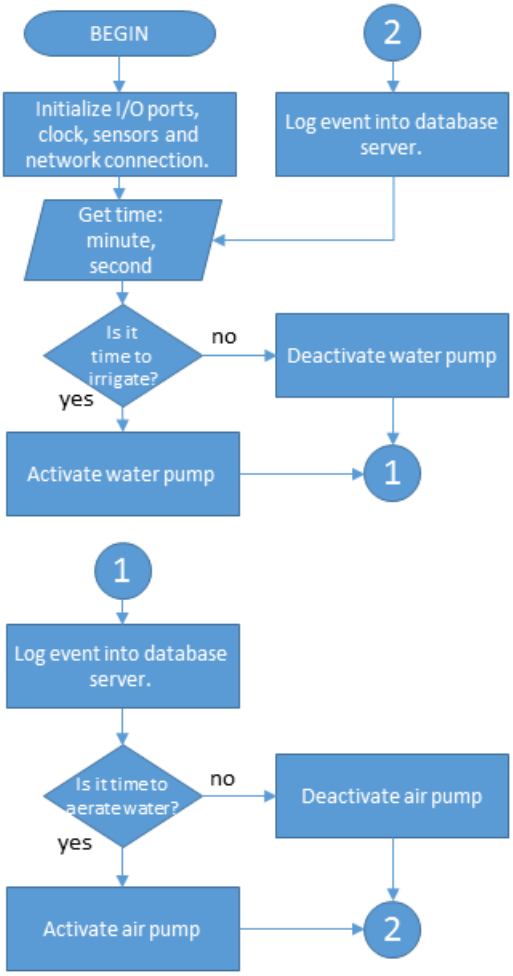


Fig. 4. Data Flow: AILCG Irrigation Control.

The first task of a micro-controller is to initialize the inputs and outputs ports, then, the clock is set up and later, it gets the information from sensors every 10 seconds to reduce the amount of data. Then based on the current time it decides to keep the LED lighting during the day (from 7hrs to 19 hrs). Also, if the air temperature is higher than 25°C or humidity is higher than 50%, multiple ventilation fans are immediately activated to regulate the temperature tray-level to a permissible one. And if the water temperature is below 20°C, a water heater is activated. At the end of this process, the collected data is sent to a database server. This process is repeated indefinitely. The Figure 5 shows a data flow chart of this process.

4. RESULTS

Figures 6 and 7 shows the physical facility of the Automated Indoor Low-Cost Greenhouse.

Figure 8 shows the data gathered from the database (MySQL Community Server 5.6.23) which is fed by the micro-controllers sensors, it is seen that the temperature starts to increase from 8:00am to 2:00pm, and then temperature decreases.

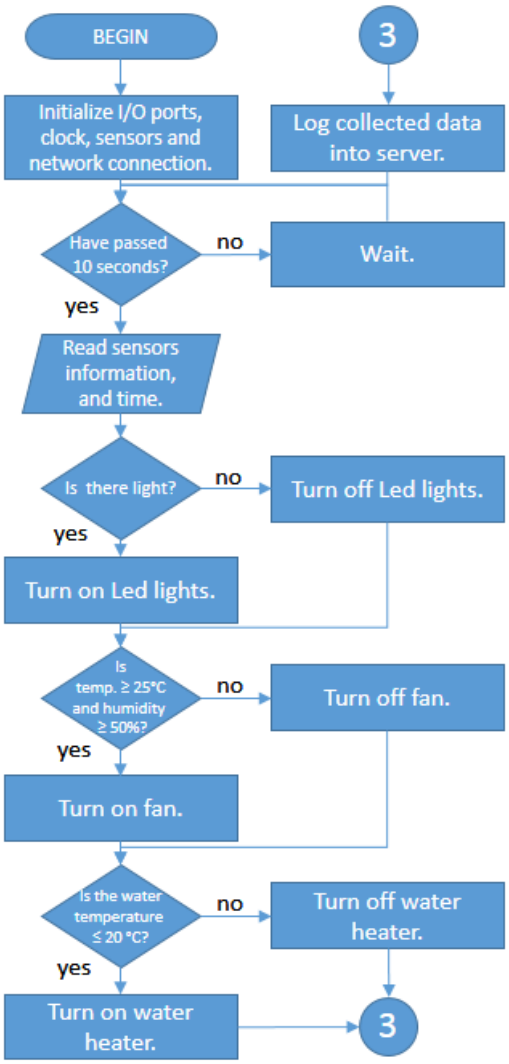


Fig. 5. Data Flow: Automated control for the crop development.

Fig. 6. Testbed setup.

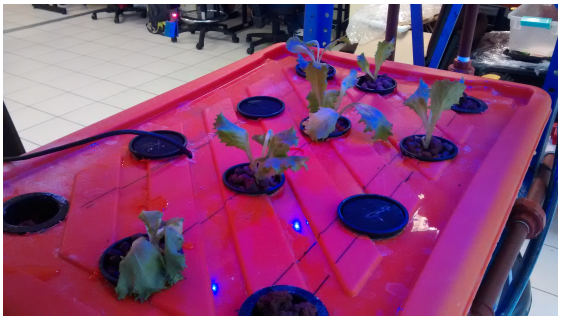


Fig. 7. LED Lights working in the testbed.

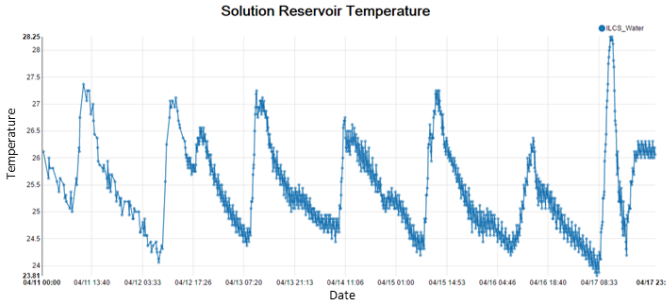


Fig. 8. Chart using data from water temperature sensor.

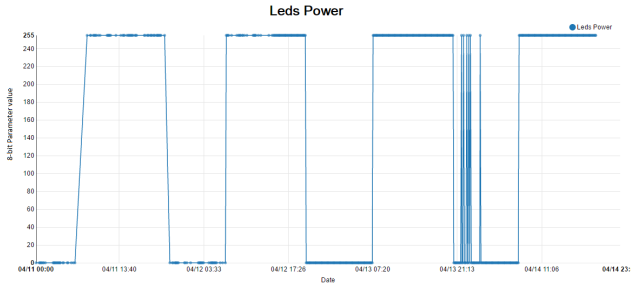


Fig. 9. Chart of LED light period.

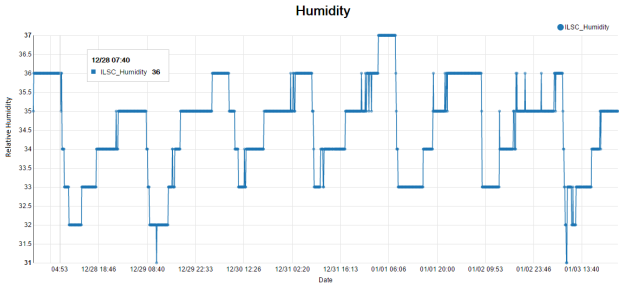


Fig. 10. Relative humidity chart.

The LED light on period is shown in Figure 9 and it is working as expected, the light period is the same period of the sunlight from 7:00hrs to 19:00hrs. Also, is shown that the date 14/04 the LIP were full at midnight due to incident light beams into the photo-resistance of external light sources, in this case, the photo-resistance was set up in front of a parking lot, by this, we deduce that the incident beams into photo-resistance were from different vehicles.

The relative humidity is increasing as the heat decreases, as shown in Figure 10, in this case, the AILCG is set up near a window in which the solar beams are heating the

equipment, reducing the humidity between 10:00am and 6:00pm.

5. CONCLUSION

A contribution of this article is the automation of a Low-Cost Indoor Greenhouse. The plants growing in the present AILCG have been more efficient by providing an adequate environment using micro-controllers, respect to a manual control. Also, we have identified and worked with four basic components that influences the plant development, which are:

- Irrigation control.
- Lighting control.
- Water condition control.
- Climate condition control.

From the experimental point of view, the data gathered from sensors and actuators allows to measure, make observations and make decisions. Finally, this AILCG provides a testbed to perform research, allowing to design new and better control strategies to indoor greenhouses.

5.1 Future Work

Today we are working with different ICT architectures, which are not part of the current article, once these architectures are validated they are going to be published on a next paper.

The next step is to implement an Internet of Plant (IoP) node consisting of a Personal Food Computer (PFC) (Harper and Siller (2015)), with the aim to contribute and share knowledge and experience to its open forum and community.

REFERENCES

Badamasi, Y.A. (2014). The working principle of an arduino. In *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*. IEEE.

Bhutada, S., Shetty, S., Malye, R., Sharma, V., Menon, S., and Ramamoorthy, R. (2005). Implementation of a fully automated greenhouse using SCADA tool like LabVIEW. In *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.*, 741–746. IEEE.

Boulard, T. and Draoui, B. (1995). Natural ventilation of a greenhouse with continuous roof vents: Measurements and data analysis. *Journal of Agricultural Engineering Research*, 61(1), 27–35.

Casas-Carrillo, R., Begovich, O., Ruiz-Leon, J., and Siller, M. (2015). Characterizing the behavior of "greenhouse climate": a LabVIEW^{TM} application. In *2015 12th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*. IEEE.

Chang, C.L. and Chang, K.P. (2014). The growth response of leaf lettuce at different stages to multiplewavelength-band light-emitting diode lighting. *Scientia Horticulturae*, 179, 78–84.

Domingues, D.S., Takahashi, H.W., Camara, C.A., and Nixdorf, S.L. (2012). Automated system developed to control pH and concentration of nutrient solution evaluated in hydroponic lettuce production. *Computers and Electronics in Agriculture*, 84, 53–61.

Harper, C. and Siller, M. (2015). OpenAG: A globally distributed network of food computing. *IEEE Pervasive Comput.*, 14(4), 24–27.

Jr., J.B.J. (2004). *Hydroponics: A Practical Guide for the Soilless Grower*. CRC Press.

Lieth, J.H. and Oki, L.R. (2008). Irrigation In Soilles Production. In *Soilless Culture*, 117–156. Elsevier BV.

Mohr, H. and Schopfer, P. (2010). *Plant Physiology*. Springer.

Muñoz, P., Montero, J., Antón, A., and Giuffrida, F. (1999). Effect of insect-proof screens and roof openings on greenhouse ventilation. *Journal of Agricultural Engineering Research*, 73(2), 171–178.

Mustroph, A. and Albrecht, G. (2003). Tolerance of crop plants to oxygen deficiency stress: fermentative activity and photosynthetic capacity of entire seedlings under hypoxia and anoxia. *Physiol Plant*, 117(4), 508–520.

Parker, R. (2009). *Plant and Soil Science: Fundamentals and Applications*. DELMAR.

Putter, E. and Gouws, J. (1996). An automatic controller for a greenhouse using a supervisory expert system. In *Proceedings of 8th Mediterranean Electrotechnical Conference on Industrial Applications in Power Systems, Computer Science and Telecommunications (MELECON 96)*. IEEE.

Schempf, H., Graham, T., Fuchs, R., and Gasior, C. (2001). Automated container-handling system for container production nurseries. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*. IEEE.

Son, K.H. and Oh, M.M. (2013). Leaf shape, growth, and antioxidant phenolic compounds of two lettuce cultivars grown under various combinations of blue and red light-emitting diodes. *HortScience*, 48.

Thompson, H., Langhans, R.W., Both, A.J., and Albright, L.D. (1998). Shoot and root temperature effects on lettuce growth in a floating hydroponic system. *Journal of the American Society for Horticultural Science*.

UN (2004). World population to 2300. Technical report, United Nations.

Zhang, C., Feng, X., and Li, L. (2010). The key technologies of a distributed temperature monitoring system based on 1-Wire bus. In *2010 8th World Congress on Intelligent Control and Automation*. IEEE.

An Integral Sliding Mode Observer for Linear Systems

Esteban Jiménez-Rodríguez * **Eliana Mejía-Estrada** **
Oscar Jaramillo * **Juan Diego Sánchez-Torres** ***

* *Department of Electrical Engineering, CINVESTAV-IPN Guadalajara, Av. del Bosque 1145 Col. El Bajío CP 45019, México (e-mail: {ejimenezr, odjaramillo}@gdl.cinvestav.mx).*
** *Department of Electrical and Control Engineering, Universidad Nacional de Colombia, Sede Medellín, Carrera 80 No 65-223, Medellín, Colombia (e-mail: elmejiaes@unal.edu.co)*
*** *Department of Mathematics and Physics, ITESO, Periférico Sur Manuel Gómez Morín 8585 C.P. 45604, Tlaquepaque, Jalisco, México (e-mail: dsanchez@iteso.mx)*

Abstract: In this paper a sliding-mode observer for linear time-invariant systems is proposed. The observer is based on integral sliding modes and the equivalent control method. In order to induce a sliding mode in the output error, a second order sliding mode algorithm is used. Convergence proofs of the proposed observer are presented. In order to expose the features of this proposal, a design example over a DC motor model is exposed, noiseless and noisy measurements cases are considered. For this case, the simulation shows the high performance of the integral observer.

Keywords: Integral Controllers, Linear Systems, Sliding-Mode Control, State Observers.

1. INTRODUCTION

A large amount of controller design methods are developed under the assumption that the state vector is available. However, the state vector can not always be completely measured, but a part of it (Luenberger, 1964). This is due to several reasons, such as there are no on-line sensors for some variables, sometimes it is impossible to install sensors due to hostile environments and some sensors are very expensive or with poor accuracy.

The state observers have taken place as a solution to this issues. The purpose of a state observer is to estimate the unmeasured state variables based on the measured inputs and outputs. Often, an observer is a replica of the original system mathematical model plus a correction signal depending on the difference between the system measured variables and the observer outputs (Luenberger, 1964; Walcott et al., 1987; Kalman, 1960; Kalman and Bucy, 1961).

Several state observers for linear systems have been proposed. A first approach is the Luenberger observer. Here, the observation problem is treated for the case when the system is completely deterministic (no statistical processes are involved) (Luenberger, 1964). When the output measurements are corrupted by zero mean, uncorrelated and white noise, the well-known Kalman Filter provides the optimal solution, once the statistical properties of noise are known (Kalman, 1960; Kalman and Bucy, 1961).

As alternative, an important class of state observers are the sliding mode observers (SMO) which have the main features of the sliding mode (SM) algorithms (Utkin, 1992). Those algorithms, are proposed with the idea to drive the dynamics of a system to an sliding manifold, that is an integral manifold with finite reaching time (Drakunov and Utkin, 1992), exhibiting very interesting features such as work with reduced observation error dynamics, the possibility of obtain a step by step design, robustness and insensitivity under parameter variations and external disturbances, and finite time stability (Utkin, 1992). In addition, some SMO have attractive properties similar to those of the Kalman filter (i.e. noise resilience) but with simpler implementation (Drakunov, 1983). Sometimes this design can be performed by applying the equivalent control method (Drakunov, 1992; Drakunov and Utkin, 1995), allowing the proposal of robust to noise observers, since the equivalent control is slightly affected by noisy measurements. On the other hand, a common and effective approach to sliding mode control is the integral SM (Matthews and DeCarlo, 1988; Utkin and Shi, 1996; Fridman et al., 2006; Galván-Guerra and Fridman, 2013). Here, it is designed an sliding manifold such that the sliding motion has the same dimension that the original system but without the influence of the matched disturbances. Those disturbances belong to the span of the control function and are rejected for the equivalent control obtained from induce the integral SM (Draženović, 1969). In order to propose that manifold, integral SM terms are designed based on the nominal system. When the system initial conditions are known, this control algorithm can be proposed with the aim to force the system trajectory starting from the sliding manifold,

eliminating the reaching phase and ensuring robustness from the initial time.

Consequently, in this paper an integral sliding mode-based observer for linear systems is proposed. The observer structure is similar to the observer presented on Drakunov (1992), but using integral SM. In addition, a step by step design of the proposed observer is provided along with a design example over a DC motor model.

The following sections are organized as follows: Section 2 presents the preliminaries for the observer. In Section 3, the integral SM observer is presented. A design example is analyzed in Section 4. In Section 5 the simulation results are shown. Finally, the conclusions of this paper are presented in Section 6.

2. MATHEMATICAL PRELIMINARIES

This section presents the previous results needed for the proposed observer.

2.1 The Super-Twisting Algorithm

Consider the first order perturbed system

$$\dot{\xi} = -u + \Delta, \tag{1}$$

where $\xi, \Delta, u \in \mathbb{R}$.

The super-twisting controller $u = \mathcal{ST}(\xi)$ (Levant, 1993), is defined as

$$\begin{aligned} \mathcal{ST}(\xi) &= \alpha_1 |\xi|^{\frac{1}{2}} \operatorname{sign}(\xi) + w \\ \dot{w} &= \alpha_2 \operatorname{sign}(\xi), \end{aligned} \tag{2}$$

with $\operatorname{sign}(x) = 1$ for $x > 0$, $\operatorname{sign}(x) = -1$ for $x < 0$ and $\operatorname{sign}(0) \in \{-1, 1\}$.

For the system (1), the controller (2) is applied, yielding the closed loop system:

$$\begin{aligned} \dot{\xi} &= -\alpha_1 |\xi|^{\frac{1}{2}} \operatorname{sign}(\xi) + q \\ \dot{q} &= -\alpha_2 \operatorname{sign}(\xi) + \dot{\Delta}, \end{aligned} \tag{3}$$

where $q = w + \Delta$.

Assuming that $\left|\dot{\Delta}\right| < \delta$, the super-twisting gains are selected as: $\alpha_1 = 1.5\delta^{\frac{1}{2}}$ and $\alpha_2 = 1.1\delta$. Therefore, a sliding mode is induced on the manifold $(\xi, q) = (0, 0)$ in a finite-time $t_q > 0$ (Moreno and Osorio, 2008). Thus, from (3), the term w in (2) becomes equal to $-\Delta$.

Now, consider the multi-variable case, with $\xi = [\xi_1 \ \dots \ \xi_p]^T, \Delta = [\Delta_1 \ \dots \ \Delta_p]^T, u = [u_1 \ \dots \ u_p]^T \in \mathbb{R}^p$. Assuming $\left|\left|\dot{\Delta}\right|\right| < \delta$, it can be shown that $\left|\dot{\Delta}_i\right| < \delta_i \ \forall i \in 1, \dots, p$. In this case, define $u = \mathcal{ST}(\xi) = [\mathcal{ST}(\xi_1) \ \dots \ \mathcal{ST}(\xi_p)]$ and note that this multi-variable case is simply the same as having p (1)-like scalar systems.

2.2 Linear Systems

Consider the following time-invariant linear system represented by the following state space equation:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx, \end{aligned} \tag{4}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector, $y \in \mathbb{R}^k$ is the output vector, $A \in \mathbb{R}^{n \times n}$ is the transition matrix, $B \in \mathbb{R}^{n \times m}$ is the input-state distribution matrix and $C \in \mathbb{R}^{k \times n}$ is the output matrix, which will be assumed to have full row rank so the measured outputs are independent. Additionally, it will be assumed that the pair (A, C) is observable.

This paper deals with the case when the measured output is a part of the state. In this case, the system (4) can be rewritten as:

$$\begin{aligned} \dot{x}_1 &= A_{11}x_1 + A_{12}x_2 + B_1 u \\ \dot{x}_2 &= A_{21}x_1 + A_{22}x_2 + B_2 u \\ y &= x_1, \end{aligned} \tag{5}$$

where $A_{11} \in \mathbb{R}^{k \times k}$, $A_{12} \in \mathbb{R}^{k \times (n-k)}$, $A_{21} \in \mathbb{R}^{(n-k) \times k}$, $A_{22} \in \mathbb{R}^{(n-k) \times (n-k)}$, $B_1 \in \mathbb{R}^{k \times m}$, $B_2 \in \mathbb{R}^{(n-k) \times m}$, are partitions of the matrices A and B, such that:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \ B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix};$$

$y = x_1 \in \mathbb{R}^k$ is the measured part of the state vector and $x_2 \in \mathbb{R}^{(n-k)}$ is the unmeasured part of the state vector.

Many linear systems can be directly expressed in the form described by (5) (i.e., the measured output is a part of the state vector). If not, under the assumption that C is full rank, there is always a linear transformation which allows to express the system (4) in the form (5), as described in (Utkin, 1992). For instance, assuming the output vector y may be represented as:

$y = K_1 x_1 + K_2 x_2, \ x^T = [x_1 \ x_2]^T, \ x_1 \in \mathbb{R}^k, \ x_2 \in \mathbb{R}^{(n-k)}$,

consider a coordinate transformation $x \mapsto Tx$ associated with the invertible matrix

$$T = \begin{bmatrix} K_1 & K_2 \\ I_k & 0 \end{bmatrix}$$

Applying the change of coordinates $x \mapsto Tx$, the triplet (A, B, C) has the form:

$$TAT^{-1} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \ TB = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \ CT^{-1} = [I_k \ 0].$$

3. INTEGRAL SLIDING MODE OBSERVER

3.1 Observer Scheme

Based on (5), the following state observer is proposed:

$$\begin{aligned} \dot{\hat{x}}_1 &= A_{11}\hat{x}_1 + A_{12}\hat{x}_2 + B_1 u + v_0 + v_1 \\ \dot{\hat{x}}_2 &= A_{21}\hat{x}_1 + A_{22}\hat{x}_2 + B_2 u + L_2 v_1 \\ v_0 &= L_1 \tilde{x}_1 \\ v_1 &= \mathcal{ST}\{\sigma\} \\ \sigma &= \tilde{x}_1 + z, \end{aligned} \tag{6}$$

where $\hat{x}_1$ and $\hat{x}_2$ are the estimates of x_1 and x_2, respectively; $\tilde{x}_1 = x_1 - \hat{x}_1$ is the estimation error variable; $v_0 \in \mathbb{R}^k$ and $v_1 \in \mathbb{R}^k$ are the observer input injections; $\sigma \in \mathbb{R}^k$ is the sliding variable and $z \in \mathbb{R}^k$ is an integral variable to be defined thereafter. Finally, $L_1 \in \mathbb{R}^{k \times k}$ and $L_2 \in \mathbb{R}^{(n-k) \times k}$ are the observer gains.

3.2 Convergence Analysis

Define the estimation error variable $\tilde{x}_2 = x_2 - \hat{x}_2$. From (5) and (6), it follows

$$\begin{aligned} \dot{\tilde{x}}_1 &= A_{11}\tilde{x}_1 + A_{12}\tilde{x}_2 - v_0 - v_1 \\ \dot{\tilde{x}}_2 &= A_{21}\tilde{x}_1 + A_{22}\tilde{x}_2 - L_2 v_1. \end{aligned} \tag{7}$$

First, note that the σ-dynamics are given by:

$$\begin{aligned} \dot{\sigma} &= \dot{\tilde{x}}_1 + \dot{z} \\ &= A_{11}\tilde{x}_1 + A_{12}\tilde{x}_2 - v_0 - v_1 + \dot{z}. \end{aligned} \tag{8}$$

Define now $\dot{z} = -A_{11}\tilde{x}_1 + v_0$, then

$$\dot{\sigma} = A_{12}\tilde{x}_2 - v_1. \tag{9}$$

The term $A_{12}\tilde{x}_2$ is assumed to be an unknown disturbance but with bounded time derivative, with $\left|\left|\frac{d}{dt}\left[A_{12}\tilde{x}_2\right]\right|\right| < \delta$ and $\delta > 0$ is a known positive constant. Then, since $v_1 = \mathcal{ST}\{\sigma\}$, it follows that $(\sigma(t), q) = (0, 0)\ \forall t > t_q$, with $q = w - A_{12}\tilde{x}_2$.

From the above analysis and (9), it follows that the equivalent control of v_1 (Utkin, 1992) is

$$\{v_1\}_{eq} = A_{12}\tilde{x}_2,$$

which implies that the motion of the system (7) constrained to the sliding manifold $(\sigma, q) = (0, 0)$ is given by:

$$\begin{aligned} \dot{\tilde{x}}_1 &= (A_{11} - L_1 I_k)\,\tilde{x}_1 \\ \dot{\tilde{x}}_2 &= A_{21}\tilde{x}_1 + (A_{22} - L_2 A_{12})\,\tilde{x}_2. \end{aligned} \tag{10}$$

where $I_k \in \mathbb{R}^{k\times k}$ is the k-order identity matrix. Hence, the system (10) associated eigenvalues are given by

$$\det\begin{bmatrix} \lambda I_k - (A_{11} - L_1 I_k) & 0 \\ -A_{21} & \lambda I_{n-k} - (A_{22} - L_2 A_{12}) \end{bmatrix} = \det\left[\lambda I_k - (A_{11} - L_1 I_k)\right] \det\left[\lambda I_{n-k} - (A_{22} - L_2 A_{12})\right].$$

Since the pair (A_{11}, I_k) is always observable, it is possible to choose the gain L_1 so the matrix $A_{11} - L_1 I_k$ be Hurwitz. On the other hand, since the pair (A, C) was assumed to be observable, it can be shown that the pair (A_{22}, A_{12}) is also observable (Drakunov and Utkin, 1995; Shtessel et al., 2013). Then, the gain L_2 can be chosen so the matrix $A_{22} - L_2 A_{12}$ be Hurwitz. Hence, $\tilde{x}_1, \tilde{x}_2 \to 0$ as $t \to \infty$, and the convergence analysis is completed.

Remark 3.1. It is important to note that, with the proposed observer scheme (6), the dynamic behavior of the estimation blocks $\tilde{x}_1$ and $\tilde{x}_2$ can be tuned independently (see (10)).

4. DESIGN EXAMPLE

To verify the proposed observer performance, it will be applied to the following DC motor model (Utkin and Shi, 1996):

$$\begin{aligned} \dot{i} &= \frac{-R}{L} i - \frac{\lambda}{L}\omega + \frac{1}{L} V \\ \dot{\omega} &= \frac{K}{J} i - \frac{b}{J}\omega \\ y &= i \end{aligned} \tag{11}$$

where i is armature current, V is terminal voltage, ω is shaft speed, R is armature resistance, L is armature inductance, J is moment of inertia of the rotor, b is motor viscous friction constant and λ is back-EMF constant. Finally the measurable output of system is the armature current i.

Note that the DC motor model (11) has the form (5), with $i = x_1 = y$ and $\omega = x_2$; $A_{11} = -\frac{R}{L}$, $A_{12} = -\frac{\lambda}{L}$, $A_{21} = \frac{K}{J}$ and $A_{22} = -\frac{b}{J}$; $B_1 = \frac{1}{L}$ and $B_2 = 0$. Then, the integral SMO is given by (6), with $\dot{z} = -A_{11}\tilde{x}_1 + v_0$.

The simulation results for this design example are shown in the next section.

5. SIMULATION RESULTS

All simulations presented here were conducted using the Euler integration method with a fundamental step size of 1×10^{-3} [s]. The DC Motor parameters are shown in Table 1 (Utkin et al., 1999).

Table 1. Nominal Parameters of the DC motor model (11).

Parameter	Values	Unit
L	0.001	V
R	0.5	Ω
λ	0.001	$\mathrm{V \cdot s \cdot rad^{-1}}$
b	0.001	$\mathrm{N \cdot m \cdot s \cdot rad^{-1}}$
k	0.008	$\mathrm{N \cdot m \cdot A^{-1}}$
J	0.001	$\mathrm{kg \cdot m^2}$

The initial conditions for the system (11) were selected as: $i(0) = 31.5\,\mathrm{A}$ and $\omega(0) = 250\,\mathrm{rad/s}$; furthermore, for the designed observer in the form (6), the initial conditions were chosen as: $\hat{i}(0) = 25.2\,\mathrm{A}$, $\hat{\omega}(0) = 200\,\mathrm{rad/s}$, $z(0) = 0$ and $w(0) = 0$. In addition, applying super twisting algorithm (2), the parameter values for the observer were adjusted as: $L_1 = 2 \times 10^{-4}$, $L_2 = -0.01$, $\alpha_1 = 4.7434$ and $\alpha_2 = 11$.

This section is divided into two parts. In the first part, there is assumed that the current measurements are noiseless; in the second part instead, there is included a normally distributed random signal as measurement noise in the current. The applied voltage V is a DC source, with a magnitude of 16 V, which is suddenly reduced to 15 V at $t = 25$ [s].

5.1 Noiseless Measurements

In this subsection, there is assumed no noise in the current measurements. The following results were obtained:

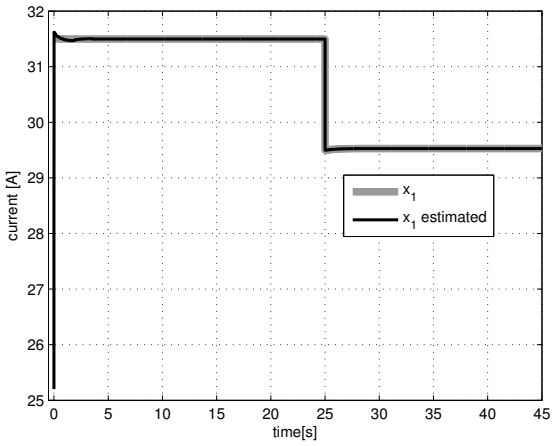


Figure 1. Armature current (i) (actual and estimated).

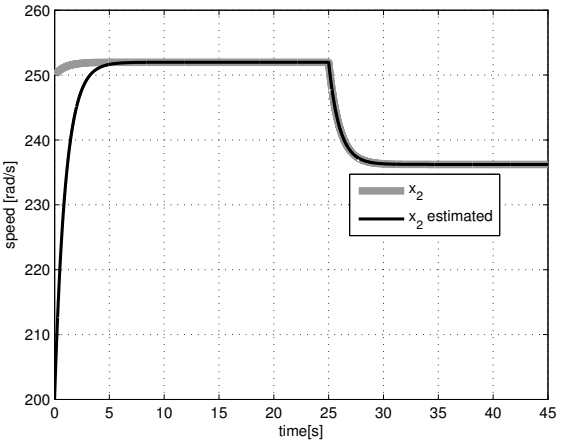


Figure 2. Shaft speed (ω) (actual and estimated).

Fig. 1 and Fig. 2 show the comparison between the actual and estimated variables corresponding to the armature current i and shaft speed ω respectively, for noiseless measurements.

5.2 Noisy Measurements

In this subsection, it is assumed that the current measurements were corrupted by a normally distributed random signal with zero mean and a variance of 10. This assumed variance corresponds to a current sensor with an accuracy of ± 9.5 A. This large variance was assumed to see significant variations in the simulation due to the noise and verify the filtering capabilities of the proposed observer. The following results were obtained:

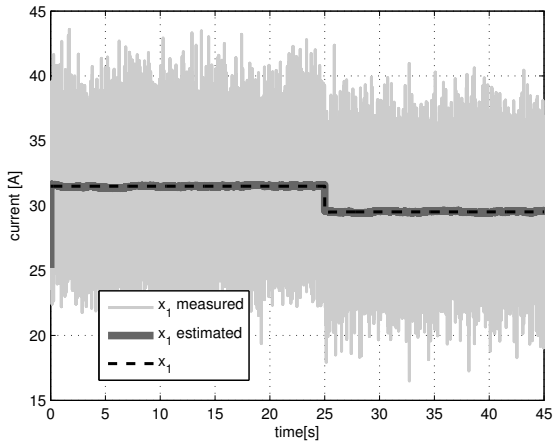


Figure 3. Armature current (i) (measured, estimated and actual).

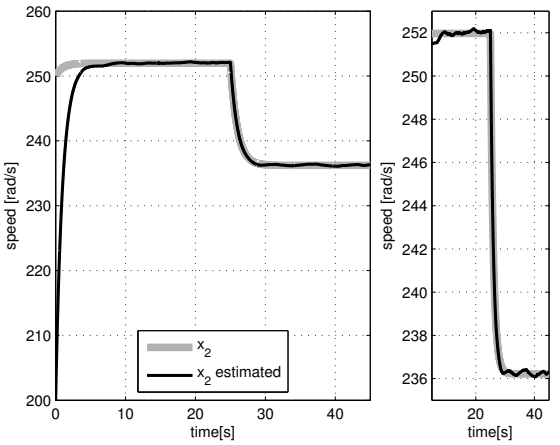


Figure 4. Shaft speed (ω) (actual and estimated).

Fig. 3 and Fig. 4 show the comparison between the actual and estimated variables corresponding to the armature current i and shaft speed ω respectively, for noisy measurements.

Based on the presented figures, it can be observed a good performance of the proposed observer. Under noisy conditions the armature current estimation $\hat{i}$ is much closer to its actual value than its measurement (Fig 3). In addition, a correct estimation of ω using the integral sliding mode observer is achieved (Figs. 2, 4) making the proposed observer suitable for observer-based control applications.

6. CONCLUSION

In this paper an integral sliding mode observer for linear time-invariant systems is proposed. The convergence of the estimation errors to zero for the proposed observer was proved. A step by step design of the proposed observer was provided along with a design example over a DC motor model. The simulation results of the example shown the filtering capabilities of the proposed observer.

ACKNOWLEDGEMENTS

The authors would like to thank the PhD Héctor A. Botero, who is currently an associate professor at the Department of Electrical and Control Engineering, Universidad Nacional de Colombia, Sede Medellín, for all his support through the realization of this paper.

Esteban Jiménez acknowledges to CONACyT, México for the MSc scholarship number 426598.

REFERENCES

Drakunov, S.V. (1983). An adaptive quasioptimal filter with discontinuous parameters. *Automation and Remote Control*, 44(9), 1167–1175.

Drakunov, S.V. (1992). Sliding-mode observers based on equivalent control method. *[1992] Proceedings of the 31st IEEE Conference on Decision and Control*. doi: 10.1109/CDC.1992.371368.

Drakunov, S.V. and Utkin, V.I. (1992). Sliding mode control in dynamic systems. *International Journal of Control*, 55, 1029–1037.

Drakunov, S.V. and Utkin, V.I. (1995). Sliding mode observers. Tutorial. *Proceedings of 1995 34th IEEE Conference on Decision and Control*, 4. doi: 10.1109/CDC.1995.479009.

Draženović, B. (1969). The invariance conditions in variable structure systems. *Automatica*, 5(3), 287 – 295.

Fridman, L., Levant, A., and Davila, J. (2006). High-order sliding-mode observer for linear systems with unknown inputs. In *2006 14th Mediterranean Conference on Control and Automation*, 1–6. doi: 10.1109/MED.2006.328736.

Galván-Guerra, R. and Fridman, L. (2013). Output integral sliding mode observer for linear time variant systems. In *Automatic Control, México 2013. CNCA 2013. National Congress of*.

Kalman, R.E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(Series D), 35–45. doi: 10.1115/1.3662552.

Kalman, R.E. and Bucy, R.S. (1961). New Results in Linear Filtering and Prediction Theory. doi: 10.1115/1.3658902.

Levant, A. (1993). Sliding order and sliding accuracy in sliding mode control. *International Journal of Control*, 58(6), 1247–1263.

Luenberger, D.G. (1964). Observing the State of a Linear System. *IEEE Transactions on Military Electronics*, 8(2). doi:10.1109/TME.1964.4323124.

Matthews, G.P. and DeCarlo, R.A. (1988). Decentralized tracking for a class of interconnected nonlinear systems using variable structure control. *Automatica*, 24(2), 187–193. doi:http://dx.doi.org/10.1016/0005-1098(88)90027-1.

Moreno, J. and Osorio, M. (2008). A lyapunov approach to second-order sliding mode controllers and observers. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 2856–2861. doi: 10.1109/CDC.2008.4739356.

Shtessel, Y., Edwards, C., Fridman, L., and Levant, A. (2013). *Sliding Mode Control and Observation*. Springer Science & Business Media.

Utkin, V.I. (1992). *Sliding Modes in Control and Optimization*. doi:10.1007/978-3-642-84379-2.

Utkin, V.I., Guldner, J., and Shi, J. (1999). *Sliding Mode Control in Electro-mechanical Systems*. Taylor and Francis.

Utkin, V.I. and Shi, J. (1996). Integral sliding mode in systems operating under uncertainty conditions. *Proceedings of 35th IEEE Conference on Decision and Control*, 4. doi:10.1109/CDC.1996.577594.

Walcott, B.L., Corless, M.J., and Zak, S.H. (1987). Comparative study of nonlinear state observation techniques. *Int. J. Control*, 45, 2109–2132.

Automatic Face Recognition in Thermal Images Using Deep Convolutional Neural Networks ⋆

Rubén D. Fonnegra * **Andrés Felipe Cardona-Escobar** *
Andrés Felipe Pérez-Zapata * **Gloria Mercedes Díaz** *

* *Grupo de Investigación Automática, Electrónica y Ciencias Computacionales*
Instituto Tecnológico Metropolitano, Medellín, Colombia
(e-mail: rubenfonnegra@itm.edu.co,
andrescardona134713@correo.itm.edu.co,
andresperez75267@correo.itm.edu.co, gloriadiaz@itm.edu.co).

Abstract: Recently, the use of infrared images has shown be a feasible technique for addressing problems as illumination dependency and facial expressions in face recognition applications. Due the recent approaches that use deep learning methodologies for image analysis, which have had remarkable performances, a deep learning strategy for facial recognition in thermographic images is proposed. A convolutional neural network is designed and evaluated for recognizing different people in an experimental home-made database. The use of convolutional networks avoids to implement preprocessing and feature extraction algorithms, which is one of the important parts in image classification. Results show significant improvements compared to others works reported in the literature, which demonstrates robustness and effectiveness of the proposed approach.

Keywords: Face recognition, Thermal images, Deep learning, Convolutional neural networks

1. INTRODUCTION

Face recognition remains an active and challenging area in the computer vision field. Although, several approaches have been proposed, problems such as illumination requirements for image acquisition are unsolved yet. Infrared imagery (IR) have shown promising results in this area, due its invariance to changes in the illumination condition, which allows to acquire approachable images even in the absence of visible light (Ghiass et al., 2014).

Face recognition based on infrared images are commonly classified in four categories (Ghiass et al., 2014; Arya et al., 2015): holistic appearance based, feature based, multispectral based, and multi-modal fusion based approaches. The first one were the earliest attempts to use IR for recognition purposes. These methods use all infrared face information in the recognition task; eigenfaces approaches fall into this category. Feature based approaches represent images as a feature vector, which can be obtained applying general feature extraction techniques, such as wavelets coefficients, local descriptors, among others; or context-dependent features such as vascular network patterns or blood perfusion measurements. On the other hand, multispectral and multi-modal approaches take advantages of the possibility of use information from different spectra or other image modalities for improving recognition capabilities.

⋆ This work was supported by the Instituto Tecnológico Metropolitano from Medellín, Colombia trough Research Group in Automática, Electrónica y Ciencias Computacionales.

The majority of the proposed approaches are characterized by a complex processing pipeline that involves, preprocessing, feature extraction and selection, and recognition learning techniques, which results in a large number of parameters that must be tuned. Recently, Deep learning has emerged as the best strategy for solving several traditional machine learning problems (LeCun et al., 2015). So, because a biometric system must be able to provide enough roughness, we considered using convolutional artificial neural networks (CNN) approaches as face recognition algorithm. The use of CNN provides the alternative of discard every preprocessing or feature extraction algorithm due the net capacity of recognizing characteristics of images by the use of convolutional transformations in their layers. Although the results show that the extracted features of previous works found in the state-of-the-art are highly relevant; our methodology based on CNN significantly improves performance without the need of a preprocessing or feature extraction stages. This advantage allows to cover all recognition structures in one same procedure.

This paper is organized as follows: section 2 introduces the theoretical concepts for the implementation of the proposed CNN; section 3 describes details of database acquisition process (section 3.1) and presents the architecture of the proposed neural network (section 3.2). Section 4 presents the experimental evaluation and results obtained when different image scales are used as input to the CNN; and finally, conclusions and future work are discussed in section 5.

2. BACKGROUND

Deep learning is the concept that involves a set of machine learning approaches that use complex architectures of non-linear transformations to representing relevant information in non-structured data with high levels of abstraction (Schmidhuber, 2015). These approaches appear a few years ago due to the development of new processors with the capacity of executing different processes using parallel programming strategies. From the proposed Deep learning approaches, convolutional neural networks have recently been widely adopted due that it is easier to train and generalized much better than other architectures. In this paper, a convolutional neural network is proposed to solve the problem of identification of people in thermal images (Hassairi et al., 2015).

2.1 Convolutional Neural Networks (CNN)

CNN is a feed-forward neural network with a particular architecture that allows to process an entire image without preprocessing requirements. It consists of a set of convolutional and subsampling layers followed by fully connected layers. Each convolutional layer accepts a feature map (in this case an image) and transforms it to another feature map (another image) through a differentiable function (i.e., kernels). Neurons in the same feature map share the same kernel, which allows to go deeper between every convolution for the transformation of the images Gong et al. (2015). Because CNNs can learn from raw data automating the process of feature extraction (Ji et al., 2013), these are known as *end-to-end* methods i.e., they can compute feature maps using spatial information of pictures. The CNN algorithms have demonstrated high performance in pattern recognition tasks over the past few years, especially in computer vision where CNNs outperform conventional classifiers, for instance at the ImageNet Classification challenge, CNNs have been used by the better proposals in the last years (Krizhevsky et al., 2012), (Simonyan and Zisserman, 2014) (Szegedy et al., 2015).

2.2 Rectified linear Units (ReLU)

Rectified Linear Units (ReLU) is an activation function, which is the most commonly deployed activation function for the outputs of the CNN neurons. This function was developed as an alternative to classical way of to determine the output of a neuron x in a net, it is given by Equation 1.

$$F(x) = tanh(x) = (1 - e^{-x})^{-1} \tag{1}$$

According to Glorot et al. (2011), the main advantage of ReLU activation is that non-linear units are not saturated compared to other non-linear activation functions (such as Logistic or Tanh units). This characteristic implies expensive processing, doing that networks with saturated neurons be slower than those using ReLU function. Although ReLU is not symmetric, this property can be achieved by combining two units Glorot et al. (2011), and sometimes this function is replaced with a smooth version named *softplus*.

In a general form, the output of a neuron with ReLU activation function can be computed by Equation 2.

$$ReLU(x_i) = max(0, x_i) \tag{2}$$

2.3 Softmax

Softmax function is an activation function, which is a generalization of a logistic function that maps a K-dimensional vector $\mathbf{Z}$ of real values to a K-dimensional vector $\sigma(\mathbf{Z})$ of real values in the range 0, 1. This function allows to handle multiclass in the output layer; it is computed one unique output when there are several units in the output layer. Due that Softmax function mapping is now considered as scores with unnormalized log probabilities for each class, the cost function correction named Cross entropy loss can be defined by Equation 3.

$$L_i = -Log(\frac{e^{f_{y_i}}}{\sum_J^{j=1} e^{f_j}}) \tag{3}$$

with f_j being the $j-th$ element of class vector scores f.

Likewise, the Softmax activation function is defined by equation 4.

$$softmax(\mathbf{z}) = \frac{e^{zj}}{\sum_K^{k=1} e^{zk}} \quad forj = 1, ..., K. \tag{4}$$

with z being an arbitrary vector with real values to be scaled into a zero-to-one values.

2.4 Dropout Regularization

Dropout regularization is used for preventing co-adaption among units. This novel technique detailed in (Srivastava et al., 2014) avoids overfitting by dropping out units randomly in hidden layers during training. for doing so, outputs of units with a probability of 0.5 in hidden layers are set to zero during the forward pass, as it is explained in Krizhevsky et al. (2012). The "Dropped out" neurons not participate in backpropagation. So, every algorithm execution changes the net architecture, but neurons also share its weights. This characteristic reduces co-adaptations of the net since every neuron does not depend on other neurons in the net. Therefore, the main goal of dropout is to add stochastically noise in the activation states of certain hidden units (Srivastava et al., 2014).

2.5 Adaptive Moment Estimation (Adam Optimizer)

Adam is a stochastic optimization algorithm introduced in Kingma and Ba (2014), which employs first order gradients as updating mechanism, it uses exponential moving averages denoted by m_t and squared gradients v_t. Parameters updating is performed defining an objective function $f(\theta)$, which is tuned by evaluating random subsets known as mini batches.

3. MATERIALS AND METHODS

3.1 The thermal Image Database

An image database was acquired using a FLIR A655SC thermographic camera, configured to a 6.3 Hz sampling frequency. A group of 21 people with ages between 22 and

45 years was involved. For each persona thermographic video with approximately 80 images of 640×480 pixels was recorded. During recording, each person was instructed to rotate his/her head in four directions (up, down, right and left). Resulting videos were converted to gray-scale, with pixel values distribution remaining those of infrared radiation for each frame. From each video, a set of images were manually selected, for doing so it was considered that heads were completely forward or slightly tilted. In this sense; the database was composed of 588 images, i.e., 21 people and 28 images per person.

Discarding the implementation of any preprocessing algorithm (this means that every input for the network corresponds to every pixel of the input image to be classified for the network), the use of original sized images represented a high computational cost to training stages. So, images were conveniently scaled using a typical bicubic interpolation. Preserving spectral relationship, We evaluate three different scales, i.e. 320×240, 160×120 and 80×60 pixels. Although this down-scaling could means losing information; the loss is not considerable enough to prevent the net to recognize all people. In Fig. 1 could be observed some examples of the subsets of images involved in this study.

Fig. 1. Example images from the home-made database. Two subjects with different pose are showed.

3.2 Neural Network Architecture

With the purpose of developing a biometric system capable of identifying every person according to the experimental database described in subsection 3.1; a convolutional neural network was implemented. The main goal of using CNN was to reach that without to apply any preprocessing technique. The Architecture of the proposed net is illustrated in Figure 2. It consists of 6 convolutional layers followed by 2 densely-connected layers that were randomly weighted. Convolving layers implemented kernels using the RelU function activation for image transformations. Every convolutional layer was connected to the immediately next convolutional layer so that there were no connections between nonconsecutive layers. The last convolving layer was connected to two densely-connected layers, which were composed of 200 and 100 output neurons, respectively. Last of them uses Softmax and dropout functions for handling multiple classes in the output layer and avoiding overfitting.

The number of input layers was varied according to the number of pixels in the input images. So, input layers with 76800, 19200 and 4800 neurons were evaluated (for 320×240, 160×120 and 80×60 pixels, respectively). This means that each neuron in the first layer possesses information of each pixel in the image. Size image variations also modify the size of the image in the last convolutional layer but did not change operations in others layer sizes because these do not depend on the size of the input vector.

4. EXPERIMENTAL RESULTS

4.1 People recognition using CNN

Implementation of the convolutional neural network was performed using TensorFlow framework Abadi et al. (2015); proposed approach was evaluated using a 10-fold cross validation strategy . For picking the training samples, it was considered to keep a stratified probability function by selecting the 10% of data per class for validation test. Along with the 10-fold cross validation, the data were randomly permuted before of evaluating the model. This also guaranteed that each experiment had different input data for training and validation stages; and demonstrates independence of them for learning task.

Figure 3 shows the average accuracy in the validation stage for different image size evaluated (320×240, 160×120 and 80×60), when number of iterations was varied between 10 and 100. In the three cases, perfect recognition i.e., a 100% of accuracy is obtained before of 30 iterations, although it is faster for larger images (around of 10 iterations), which was expected because this kind of image preserves more information than the small one.

4.2 Processing optimization with GPU

Due that the high computational cost of training a CNN, We considered the use of parallelization process in the network, using graphical processing units (GPU). For doing it, training and validation stages were implemented using NVIDIA series Quadro K4000 GPU along with CUDA Toolkit 7.5 libraries and computational cost was compared with them obtained when it was performed on a Workstation with CPU Intel® Xeon® E5 - 2687 (32 cores) and 32GB RAM inside. Tables 1 and 2 show the time consuming in the GPU and CPU configurations. Interation time refers to processing time (in seconds) for each execution cycle of the algorithm; training time is the time (in seconds) taken for training the overall net, when accuracy of 100% is accomplished; Validation time is the cost of classifying a subject after training process and memory consumption corresponds to RAM memory required to execute the training of the network.

Table 1. Processing times for GPU Implementation

Size	Iteration Time (s)	Training Time (s)	Validation Time (s)	Memory consumption (MB)
320x240	8.6211	283.927	0.210511	2440
160x120	2.29826	47.9025	0.0594253	2360
80x60	0.615124	24.1448	0.017618	2298

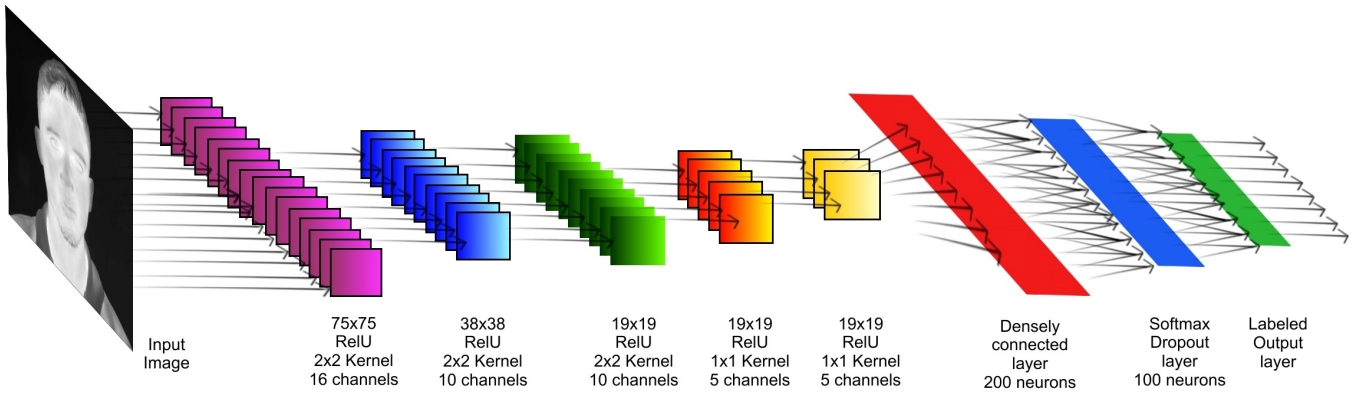


Fig. 2. Architecture of the implemented convolutional neural network.

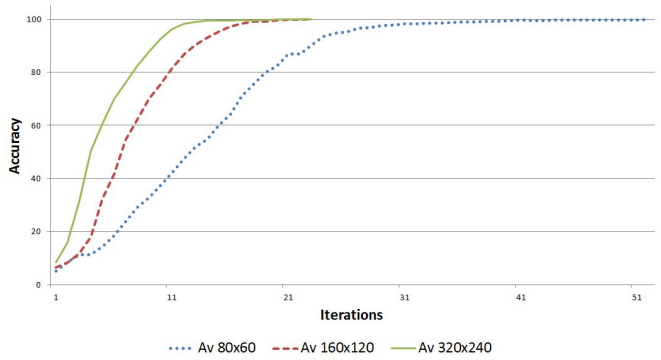


Fig. 3. Accuracy in the validation stage for input images with 320 × 240 (Green, continued line), 160 × 120 (red,discontinued line) and 80 × 60 (blue, dot line) pixels, when number of iterations in the training stage is varied

Table 2. Processing times for CPU Implementation

Size	Iteration Time (s)	Training Time (s)	Validation Time (s)	Memory consumption (MB)
320x240	31.7805	810.902	0.460278	1762
160x120	7.95153	130.649	0.139201	1103
80x60	1.9524	107.607	0.0426403	501

GPU parallelization allows to reduce the training and validation processes in 2 to 4 times respect to CPU implementation but It requires more RAM memory capacities.

5. CONCLUSIONS

In this work a biometric system for facial recognition using thermographic images was presented. Involving deep learning approaches introduced through a convolutional neural network frame, the proposed approach was able to identify each person in a home-made image database without previous face segmentation, image preprocessing or feature extraction and selection stages. The proposed framework demonstrated high performances in the training and validation tests even when the input image were scaled to a quarter of the original one. Those results are comparable to other works reported in the state of the art; obtaining notable accuracy during training and validation experiments.

Computational times are considerably reduced when the proposed neural network is implemented on GPU, although RAM memory requirements are also incremented. This results make the proposed approach a promissory alternative for real biometric applications, which could to take advantage of illumination invariance of thermographic images and the recognition performance of deep neural networks.

In the future, we plan to perform an evaluation with a higher number of subjects and to combine information from another spectrum such as near infrared or visual spectra.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. URL `http://tensorflow.org/`. Software available from tensorflow.org.

Arya, S., Pratap, N., and Bhatia, K. (2015). Future of face recognition: A review. *Procedia Computer Science*, 58, 578–585.

Ghiass, R.S., Arandjelović, O., Bendada, A., and Maldague, X. (2014). Infrared face recognition: A comprehensive review of methodologies and databases. *Pattern Recognition*, 47(9), 2807–2824.

Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier neural networks. In *International Conference on Artificial Intelligence and Statistics*, 315–323.

Gong, T., Fan, T., Guo, J., and Cai, Z. (2015). Gpu-based parallel optimization and embedded system application of immune convolutional neural network. In *2015 International Workshop on Artificial Immune Systems (AIS)*, 1–8. doi:10.1109/AISW.2015.7469248.

Hassairi, S., Ejbali, R., and Zaied, M. (2015). Supervised image classification using deep convolutional wavelets network. In *Tools with Artificial Intelligence (ICTAI),*

2015 IEEE 27th International Conference on, 265–271. doi:10.1109/ICTAI.2015.49.

Ji, S., Xu, W., Yang, M., and Yu, K. (2013). 3d convolutional neural networks for human action recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1), 221–231.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85–117.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1), 1929–1958.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9.

Bioprocesses Control Based on Linear Algebra

M. C. Fernández*, M. N. Pantano*, D. Patiño, O. Ortiz*, G. Scaglia*.**

** Instituto de Ingeniería Química, Universidad Nacional de San Juan (UNSJ), CONICET, Av. Lib. San Martín Oeste 1109, San Juan J5400ARL, Argentina. (e-mail: mcfernandez@unsj.edu.ar; npantano@unsj.edu.ar; rortiz@unsj.edu.ar; gscaglia@unsj.edu.ar)*

*** Instituto de Automática, Universidad Nacional de San Juan (UNSJ), CONICET, Av. Lib. San Martín Oeste 1109, San Juan, Argentina. (e-mail: dpatino@inaut.unsj.edu.ar)*

Abstract: Fed-batch fermenters have gained particular attention due to its wide range of high valued products production possibilities. Nowadays, the optimization and control of these systems is an important task given its significant economic impact and particular characteristics. The paper proposes a linear algebra based controller for nonlinear and multivariable bioprocesses. It involves finding the control actions to make the system follow predefined optimal concentration profiles. The controller parameters are selected with a Monte Carlo experiment. Finally, the good performance of the controller is proved under normal conditions and adding perturbations in the control action.

Keywords: nonlinear dynamics control, fed-batch fermentation, penicillin production, profiles tracking control.

1. INTRODUCTION

On the one side, the increasingly demand of high added value products, with specific characteristics for the market, has made essential the dispose of reliable tools in the production area. Due to this fact, in the last few years the optimization and control techniques development has become a fundamental topic for scientific investigations. On the other side, the bioprocesses industry has demonstrated to be an excellent option for the obtaining of a long variety of products in a natural and ecological way. Although, this kind of processes have many complications, and that is what make them to be the biggest challenge from control engineer's viewpoint (Ashoori et al., 2009).

The biological processes have been characterized for using microscopic organisms to obtain valuable substances. For example: recombinant proteins, vaccines and antibiotics in the pharmaceutical industry, or beer, wine, yeast in the manufacturing of agro-food goods, biogas and compost in the treatment of urban and industrial solid organic wastes and wastewater (Mangesh and Jana, 2008), or biopolymers in the chemical industry (Chung et al., 2015), between others.

A bioreactor can be operated in one of the following forms: batch, fed-batch or continuous. The continuous operation mood is characterized to work in a stationary way, while in batch and fed-batch procedures, the states vary in the time. This particularity of batch and fed-batch processes added to the many difficulties that any bioprocess present, make the control of them to be an arduous task to achieve. Some of the mentioned complications are: the nonlinear and unstable dynamic behavior of microorganisms which means strong modeling approximations; the presence of numerous external disturbances; troubles for most of the representative variables on-line measurement; both the initial states of the process and the parameters of the model may vary randomly from batch to batch (Liang and Chen, 2003). All this avoid the possibility of using classic industrial controllers, like PI and PID controllers, being necessary to implement control algorithms specifically developed for bioprocesses (De Battista et al., 2012).

The objective of this work is to develop a control technique for a fed-batch penicillin production process. It consists in tracking predefined state variables profiles (preferably optimized ones) with minimal error. The methodology was originally designed for robotics systems by Scaglia et al (Scaglia et al., 2009, Scaglia et al., 2010) and lately extended to other systems like unmanned vehicles (Cheein and Scaglia, 2014), chemical processes (Serrano et al., 2014a), underactuated surface vessels (Serrano et al., 2013, Serrano et al., 2014b), between others, all of them with excellent results in the path tracking.

The main advantage of this procedure is the application of linear algebra for the controller design. Here the control action (substrate feed rate) is obtained solving a linear equations system, although the controller structure rises from a nonlinear mathematical model. This means that the controller performance is independent from the operation point, obtaining good result even when the initial conditions and the parameters change.

The paper is organized as follows: The description of the system and the process is presented in Section 2. The controller design and the selection of its parameters are exposed in Section 3. While in Section 4, the simulation results are shown, this include: the performance under normal conditions and with perturbation in the control action. Finally, in Section 5, the conclusions are exposed.

2. SYSTEM AND PROCESS DESCRIPTION

The system under study is a fed-batch bioreactor for penicillin production. The substrate and the microorganism used are glucose and Penicillum crysogenum, respectively. The equations that model the process were originally proposed in (Cuthrell and Biegler, 1989), from which many optimization and control papers have been written (Lim et al., 1986, Luus, 1993, Riascos and Pinto, 2004, Ronen et al., 2002). It is a single input multi output system (SIMO), where the input is the substrate feed rate, and the outputs are the cells, product (penicillin) and substrate concentrations inside the reactor. The mathematical model of the process is:

$$\begin{cases} \dot{X}(t) = \mu(X,S)X - \left(\frac{X}{S_F V}\right)U \\ \dot{P}(t) = \rho(S)X - K_{deg}P - \left(\frac{P}{S_F V}\right)U \\ \dot{S}(t) = -\mu(X,S)\left(\frac{X}{Y_{X/S}}\right) - \rho(S)\left(\frac{P}{Y_{P/S}}\right) - \left(\frac{m_s S}{K_m + S}\right)X + \left(1 - \frac{S}{S_F}\right)\frac{U}{V} \end{cases} \quad (1)$$

$$\mu(X,S) = \mu_{max}\left(\frac{S}{K_{XG}X + S}\right)$$
$$\rho(S) = \rho_{max}\left(\frac{S}{K_{PP} + S(1 + S/K_{in})}\right) \quad (2)$$
$$\dot{V}(t) = \frac{U}{S_F}$$

In these equations, the state variables are: biomass (X), product (P) and glucose (S) concentrations. The control action is the substrate feed rate (U). S_F is the substrate feed concentration. The auxiliary equation defined are: the specific biomass growth rate *($\mu(X, S)$)*, the specific penicillin production rate *($\rho(S)$)* and the culture volume *(V)*.

In Table 1 are shown the initial variable values, whereas in Table 2 the parameter definitions and its values.

Table 1. Initial variable values for penicillin biosynthesis

Variable	Initial value
X (g/L)	1.5
P (g/L)	0.0
S (g/L)	0.0
V (L)	7.0

Table 2. Parameters of penicillin biosynthesis model

Parameter	Definition	Value
μ_{max}	Maximum specific biomass growth rate (h^{-1})	0.11
ρ_{max}	Maximum specific production rate (gP/gX h)	0.0055
K_{XG}	Saturation parameter for biomass growth (gS/gX)	0.006
K_{PP}	Saturation parameter for production (gS/L)	0.0001
K_{in}	Inhibition parameter for production (gS/L)	0.1
K_{deg}	Product degradation rate (h^{-1})	0.001
K_m	Saturation parameter for maintenance consumption (gS/L)	0.0001
m_s	Maintenance consumption rate (gS/gX h)	0.029
$Y_{X/S}$	Yield factor for substrate to biomass (gX/g S)	0.47
$Y_{P/S}$	Yield factor for substrate to product (gP/g S)	1.2
S_F	Feed concentration (gS/L)	500

3. CONTROLLER DESIGN

3.1 Controller Structure

For the application of this technique, it is necessary to know the mathematical model that represents the process, and the reference profiles that are expected to be followed by the system.

The most important variables within de bioreactor are the cells and product concentration, so their reference profiles are expected to be tracked by the controller.

The optimal substrate feed rate presented by Riascos & Pinto (Riascos and Pinto, 2004) is taken as the reference profile for substrate feed rate. The reference concentrations of cells and penicillin are determined in an open-loop simulation of the system (see Fig. 1). For this, the feed rate mentioned, the mathematical model, and the information from Table 1 and 2, are used.

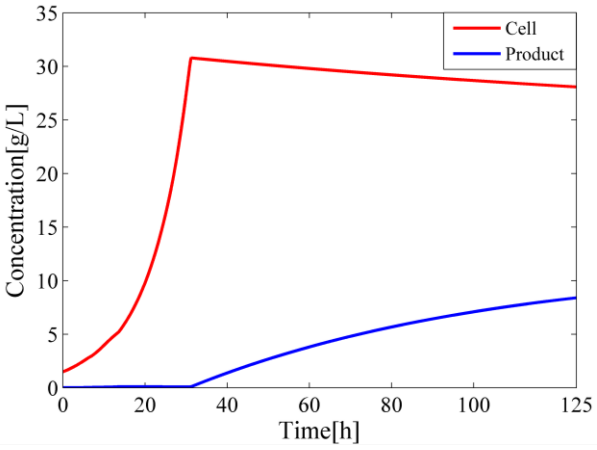


Fig. 1. Reference profiles of cells and penicillin concentration, obtained by an open-loop simulation.

In order to achieve the objective presented above, numerical methods are used to determine the evolution of the system. The method developed by Euler is used in order to integrate numerically an ordinary differential equation given an initial value.

$$\left(\frac{d\sigma}{dt}\right) = \frac{\sigma_{n+1} - \sigma_n}{T_0} \quad (3)$$

Where σ represents each state variable, σ_n is the present value of σ measured from the reactor (on-line), while σ_{n+1} is the value of σ in the next measurement instant. T_0 is the sampling time; for this study is adopted a value of 0.1 h. The process lasts 125 h.

The error in a given sampling moment is defined as the difference between the actual and reference values. As the error in *n+1* is proportional to the error in *n*, σ_{n+1} can be expressed as follows:

$$\sigma_{n+1} = \sigma_{ref\,n+1} - k_\sigma \underbrace{(\sigma_{ref\,n} - \sigma_n)}_{error} \tag{4}$$

where k_σ is a proportionality factor and represents the parameter of the controller for the generic variable σ. There are three different k_σ for this model of penicillin production, k_X, k_P and k_S.

Replacing Eq. (4) in (3) is obtained the following expression. It allows the approximation of the derivatives.

$$\left(\frac{d\sigma}{dt}\right) = \frac{\overbrace{\left[\sigma_{ref\,n+1} - k_\sigma(\sigma_{ref\,n} - \sigma_n)\right]}^{\sigma_{n+1}} - \sigma_n}{T_0} \tag{5}$$

Substituting Eq. (5) in the mathematical model:

$$\begin{cases} \dot{X}(t) = \dfrac{\left(X_{ref\,n+1} - k_X\left(X_{ref\,n} - X_n\right) - X_n\right)}{T_0} = \mu(X_n, S_n)X_n - \left(\dfrac{X_n}{S_F V_n}\right)U_n \\ \dot{P}(t) = \dfrac{\left(P_{ref\,n+1} - k_P\left(P_{ref\,n} - P_n\right) - P_n\right)}{T_0} = \rho(S_n)X_n - K_{deg}P_n - \left(\dfrac{P_n}{S_F V_n}\right)U_n \\ \dot{S}(t) = \dfrac{\left(S_{ref\,n+1} - k_S\left(S_{ref\,n} - S_n\right) - S_n\right)}{T_0} = \\ \qquad -\mu(X_n, S_n)\left(\dfrac{X_n}{Y_{X/S}}\right) - \rho(S_n)\left(\dfrac{P_n}{Y_{P/S}}\right) - \left(\dfrac{m_s S_n}{K_m + S_n}\right)X_n + \left(1 - \dfrac{S_n}{S_F}\right)\dfrac{U_n}{V_n} \end{cases} \tag{6}$$

With this new appreciation of the Eq. (1) as a system of linear equations, a simple possibility for calculating the control action is available. Besides, it could be expressed in a matrix form through placing the state variables as a function of *U*. In this way, is more straightforward to clear the control action:

$$\overbrace{\begin{pmatrix} -X_n \\ -P_n \\ S_f - S_n \end{pmatrix}}^{A} \overbrace{\frac{U_n}{S_f V_n}}^{u} = \underbrace{\begin{pmatrix} \dfrac{\left(X_{ref\,n+1} - k_X\left(X_{ref\,n} - X_n\right) - X_n\right)}{T_0} - \mu(X_n, S_n)X_n \\ \dfrac{\left(P_{ref\,n+1} - k_P\left(P_{ref\,n} - P_n\right) - P_n\right)}{T_0} - \rho(S_n)X_n + K_{deg}P_n \\ \dfrac{\left(S_{ref\,n+1} - k_S\left(S_{ref\,n} - S_n\right) - S_n\right)}{T_0} + \mu(X_n, S_n)\left(\dfrac{X_n}{Y_{X/S}}\right) + \rho(S_n)\left(\dfrac{X_n}{Y_{P/S}}\right) + \left(\dfrac{m_s S_n}{K_m + S_n}\right)X_n \end{pmatrix}}_{b} \tag{7}$$

To simplify the mathematical expression of the problem, Eq. (7) is expressed generically as follows:

$$A\,u = b$$

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \frac{U}{S_f V} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{8}$$

So as to find *U* is necessary to ask the system to have exact and unique solution. To accomplish this, **b** have to be a linear combination of **A** columns (Strang, 2006), that is to say, **A** and **b** must be parallel. This condition can be satisfied in different ways; one of them is:

$$\begin{cases} \dfrac{a_3}{a_1} = \dfrac{b_3}{b_1} \rightarrow a_3 b_1 = b_3 a_1 \\ \dfrac{a_3}{a_2} = \dfrac{b_3}{b_2} \rightarrow a_3 b_2 = b_3 a_2 \end{cases} \tag{9}$$

It is important to highlight that there are many ways to express a parallelism condition, and the results will be the same with any of them. Replacing in Eq. (9) with each corresponding matrices component values:

$$\begin{cases} \left(S_f - S_n\right)\left(\left(\dfrac{X_{ref\,n+1} - k_X\left(X_{ref\,n} - X_n\right) - X_n}{T_0}\right) - \mu(X_n, S_n)X_n\right) = \\ \left(\left(\dfrac{S_{ref\,n+1} - k_S\left(S_{ref\,n} - S_n\right) - S_n}{T_0}\right) + \mu(X_n, S_n)\left(\dfrac{X_n}{Y_{X/S}}\right) + \rho(S_n)\left(\dfrac{X_n}{Y_{P/S}}\right) + \left(\dfrac{m_s S_n}{K_m + S_n}\right)X_n\right)(-X_n) \\ \left(S_f - S_n\right)\left(\left(\dfrac{P_{ref\,n+1} - k_P\left(P_{ref\,n} - P_n\right) - P_n}{T_0}\right) - \rho(S_n)X_n + K_{deg}P_n\right) = \\ \left(\left(\dfrac{S_{ref\,n+1} - k_S\left(S_{ref\,n} - S_n\right) - S_n}{T_0}\right) + \mu(X_n, S_n)\left(\dfrac{X_n}{Y_{X/S}}\right) + \rho(S_n)\left(\dfrac{X_n}{Y_{P/S}}\right) + \left(\dfrac{m_s S_n}{K_m + S_n}\right)X_n\right)(-P_n) \end{cases} \tag{10}$$

For (10) to have solution, it is defined the "sacrificed variable", which is denoted by the subscript "ez". To select the it is necessary to analyze and interpret the role of each variable in the process. In a bioprocess, the substrate concentration directly affects cells and product concentrations, and can be regulated by varying the feed flow rate. Considering this, we chose *S* as sacrificed variable. Replacing S_{ref} by S_{ez} in Eq. (10):

$$\begin{cases} \left(S_f - S_n\right)\left(\left(\dfrac{X_{ref\,n+1} - k_X\left(X_{ref\,n} - X_n\right) - X_n}{T_0}\right) - \mu(X_n, S_{ez\,n})X_n\right) = \\ \left(\left(\dfrac{S_{ez\,n+1} - k_S\left(S_{ez\,n} - S_n\right) - S_n}{T_0}\right) + \mu(X_n, S_n)\left(\dfrac{X_n}{Y_{X/S}}\right) + \rho(S_n)\left(\dfrac{X_n}{Y_{P/S}}\right) + \left(\dfrac{m_s S_n}{K_m + S_n}\right)X_n\right)(-X_n) \\ \left(S_f - S_n\right)\left(\left(\dfrac{P_{ref\,n+1} - k_P\left(P_{ref\,n} - P_n\right) - P_n}{T_0}\right) - \rho(S_{ez\,n})X_n + K_{deg}P_n\right) = \\ \left(\left(\dfrac{S_{ez\,n+1} - k_S\left(S_{ez\,n} - S_n\right) - S_n}{T_0}\right) + \mu(X_n, S_n)\left(\dfrac{X_n}{Y_{X/S}}\right) + \rho(S_n)\left(\dfrac{X_n}{Y_{P/S}}\right) + \left(\dfrac{m_s S_n}{K_m + S_n}\right)X_n\right)(-P_n) \end{cases} \tag{11}$$

This system is solved using the fminsearch function in MATLAB. Note that the only unknown is $S_{ez,n+1}$, as $S_{ez,n}$ is obtained in the previous sampling time.

Once $S_{ez,n+1}$ value is calculated, it is replaced in the original matrix system (7). Then, the control action ($U_n/S_f V_n$) can be determined at any sampling time using least squares (Strang, 2006).

$$\frac{U}{S_f V} = \left(A^T A\right)^{-1} A^T b = \frac{a_1 b_1 + a_2 b_2 + a_3 b_3}{a_1^2 + a_2^2 + a_3^2} \tag{12}$$

3.2 Controller Parameters Selection

As it was introduced in subsection *3.1*, the performance of the bioreactor is directly affected by the controller parameters (k_σ). Those parameters take values among zero and one ($0 < k_\sigma < 1$), which makes the tracking error tends to zero when *n* tends to infinity (for space reasons, the demonstration is not shown, however, a similar example could be appreciated in (Scaglia et al., 2010)). The "*tracking error*" is defined as follows:

$$\|e_n\| = \sqrt{(X_{refn} - X_n)^2 + (P_{refn} - P_n)^2 + (S_{ezn} - S_n)^2} \tag{13}$$

The following procedure aims to find the best values for k_σ, such that the tracking error is minimized. In this test, the Monte Carlo algorithm is applied. The experiment consist in simulate the process a number of times using random values of k_σ. Then, the total error is calculated for each iteration. The k_σ that make a minimum total error are selected.

To determine the number of simulations (N) is used Eq. (14) (Tempo and Ishii, 2007). Note that in order to limit the chance of a wrong answer, an appropriate confidence (δ) and accuracy (ε) must be indicated.

$$N \geq \left\lceil \frac{\log \frac{1}{\delta}}{\log \frac{1}{1-\varepsilon}} \right\rceil \tag{14}$$

Depending on the precision to be obtained, δ and ε values are selected. For this study, $\delta=0.01$ and $\varepsilon=0.005$ Consequently, $N=1000$.

The tracking error $||e_n||$ was defined in Eq. (13) and the total error is found with the following expression:

$$E=\sum_{n=1}^{1250} \|e_n\| \tag{15}$$

The simulation results show that de best values for the parameters are:

Table 3. Controller Parameters

Parameter	Value
k_X	0.9798
k_P	0.976
k_S	0.8979

4. SIMULATION RESULTS

4.1 Controller operation under normal conditions

To simulate under normal conditions, the initial state variables values shown in Table I, the parameters of Table II, and the controller parameters determined in *3.2* subsection are used. Here, it is considered that there are no disturbances in the external environment that could affect the process.

Figure 2 present how the real cells and product concentration follow perfectly the references. Finally, Fig. 3 shows the tracking error. As it can be seen, this error tends to zero as the process moves forward.

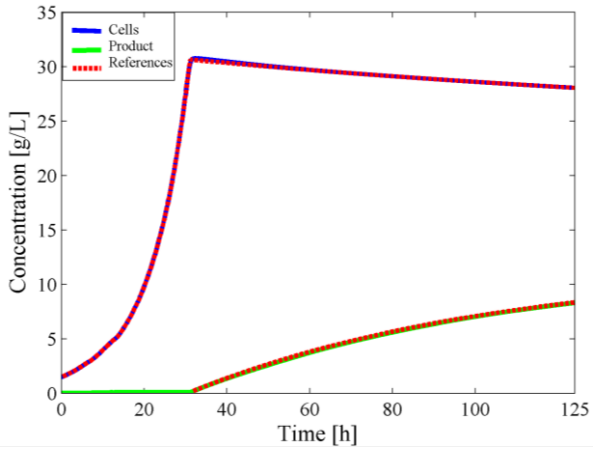


Fig. 2. Reference and real profiles of cells and penicillin concentration, obtained under normal operation conditions.

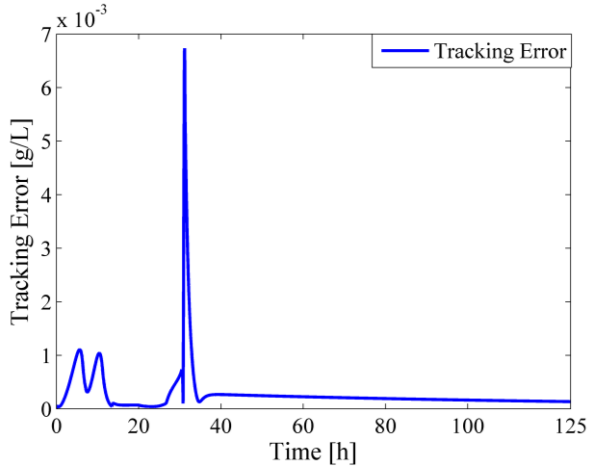


Fig. 3. Tracking error for the simulation under normal operation conditions.

4.2 Test with perturbations in the control action

In order to prove the performance of the controller, it is added a disturbance in the control action. To achieve this, the control action is affected in a 20% of its original value with a random perturbation. This can be explained as a random noise that results in non-zero-mean Gaussian disturbances (George, 2014).

Figure 4 shows the perturbed control actions. Figures 5 and 6 show the tracking of cells and product profiles in the perturbed system. Finally, Fig. 7 shows the tracking error. The disturbances in the control action cause an increasing of the tracking error, however, it remains at acceptable levels.

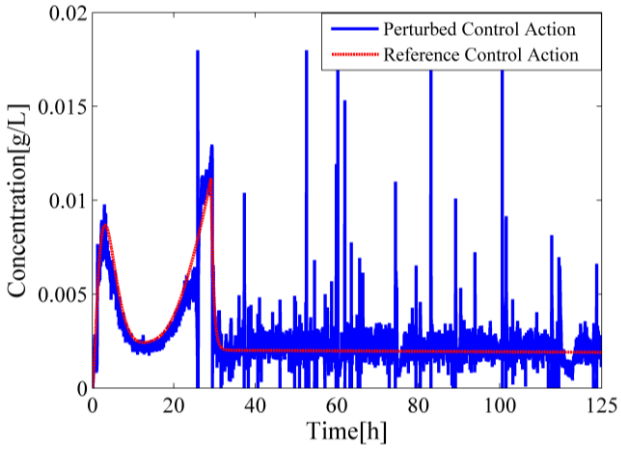


Fig. 4. Perturbed Control Action.

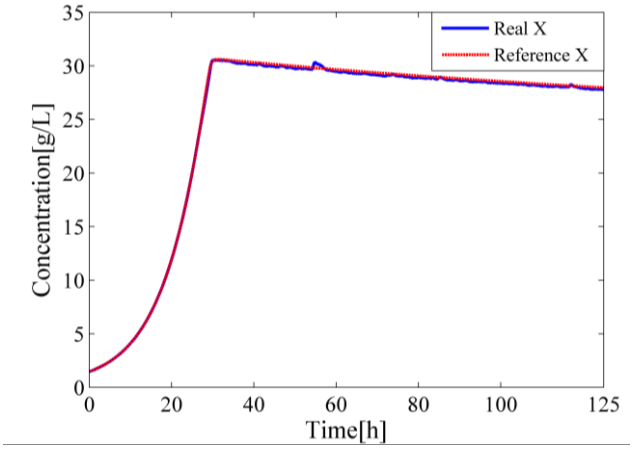


Fig. 5. Reference and real cells concentration profiles applying perturbations in the control action.

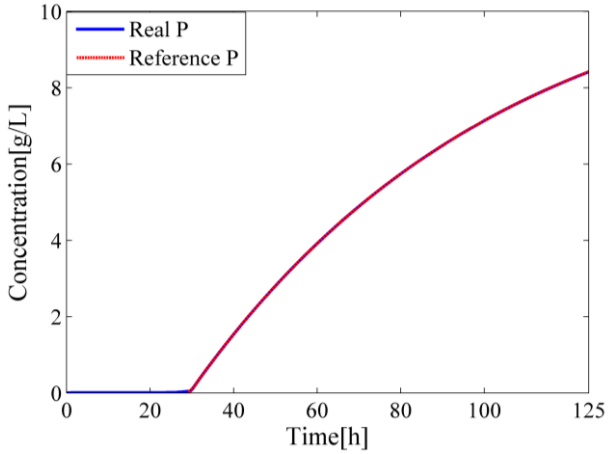


Fig. 6. Reference and real penicillin concentration profiles applying perturbations in the control action.

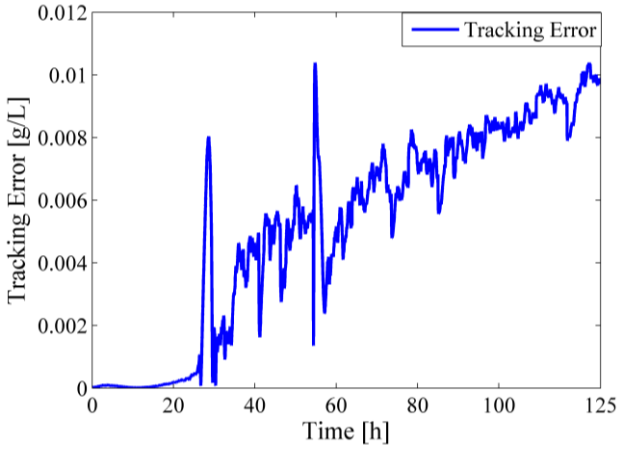


Fig. 7. Tracking error for the simulation with perturbations in the control action.

6. CONCLUSIONS

This controller has several advantages over others: it can be easily applied, there is no need to have advanced knowledge about automatic control to do it; the methodology has less mathematical complexity because it allows obtaining the control action as a solution of linear equations system, although the controller structure arises from a nonlinear mathematical model. As a result, this technique can be used in a long variety of systems. Furthermore, this controller is versatile against different disturbances; this was supported by the tests made, which results show that the controller manages to achieve the reference profile successfully at any time, and prove its good performance.

ACKNOWLEDGMENT

Grateful to the National Council of Scientific and Technological Research (CONICET) and the Chemical Engineering Institute (IIQ) from the National University of San Juan for making possible this research.

REFERENCES

ASHOORI, A., MOSHIRI, B., KHAKI-SEDIGH, A. & BAKHTIARI, M. R. 2009. Optimal control of a nonlinear fed-batch fermentation process using model predictive approach. *Journal of Process Control*, 19**,** 1162-1173.

CUTHRELL, J. E. & BIEGLER, L. T. 1989. Simultaneous optimization and solution methods for batch reactor control profiles. *Computers & Chemical Engineering*, 13**,** 49-62.

CHEEIN, F. A. & SCAGLIA, G. 2014. Trajectory tracking controller design for unmanned vehicles: A new methodology. *Journal of Field Robotics,* 31**,** 861-887.

CHUNG, H., YANG, J. E., HA, J. Y., CHAE, T. U., SHIN, J. H., GUSTAVSSON, M. & LEE, S. Y. 2015. Bio-based production of monomers and polymers by metabolically engineered microorganisms. *Current opinion in biotechnology,* 36**,** 73-84.

DE BATTISTA, H., PICÓ, J. & PICÓ-MARCO, E. 2012. Nonlinear PI control of fed-batch processes for growth rate regulation. *Journal of Process Control,* 22**,** 789-797.

GEORGE, J. 2014. On adaptive loop transfer recovery using Kalman filter-based disturbance accommodating control. *Control Theory & Applications, IET,* 8**,** 267-276.

LIANG, J. & CHEN, Y. 2003. Optimization of a fed-batch fermentation process control competition problem using the NEOS server. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering,* 217**,** 427-432.

LIM, H., TAYEB, Y., MODAK, J. & BONTE, P. 1986. Computational algorithms for optimal feed rates for a class of fed-batch fermentation: Numerical results for penicillin and cell mass production. *Biotechnology and Bioengineering,* 28**,** 1408-1420.

LUUS, R. 1993. Optimization of fed-batch fermentors by iterative dynamic programming. *Biotechnology and Bioengineering,* 41**,** 599-602.

MANGESH, M. G. & JANA, A. K. 2008. A comparison of three sets of DSP algorithms for monitoring the production of ethanol in a fed-batch baker's yeast fermenter. *Measurement,* 41**,** 970-985.

RIASCOS, C. A. & PINTO, J. M. 2004. Optimal control of bioreactors: a simultaneous approach for complex systems. *Chemical Engineering Journal,* 99**,** 23-34.

RONEN, M., SHABTAI, Y. & GUTERMAN, H. 2002. Optimization of feeding profile for a fed-batch bioreactor by an evolutionary algorithm. *Journal of biotechnology,* 97**,** 253-263.

SCAGLIA, G., MONTOYA, L. Q., MUT, V. & DI SCIASCIO, F. 2009. Numerical methods based controller design for mobile robots. *Robotica,* 27**,** 269-279.

SCAGLIA, G., ROSALES, A., QUINTERO, L., MUT, V. & AGARWAL, R. 2010. A linear-interpolation-based controller design for trajectory tracking of mobile robots. *Control Engineering Practice,* 18**,** 318-329.

SERRANO, M., SCAGLIA, G., ABALLAY, P., ORTIZ, O. & MUT, V. 2014a. LINEAR ALGEBRA AND OPTIMIZATION BASED CONTROLLER DESIGN FOR TRAJECTORY TRACKING OF TYPICAL CHEMICAL PROCESS. *Latin American Applied Research,* 44**,** 313-318.

SERRANO, M. E., SCAGLIA, G., MUT, V., ORTIZ, O. & JORDAN, M. 2013. Trajectory Tracking Controller Based on Linear Algebra: a case study in underactuated surface vessels. *Journal of Control Engineering and Applied Informatics,* 15**,** 15-25.

SERRANO, M. E., SCAGLIA, G. J., GODOY, S. A., MUT, V. & ORTIZ, O. A. 2014b. Trajectory tracking of underactuated surface vessels: A linear algebra approach. *Control Systems Technology, IEEE Transactions on,* 22**,** 1103-1111.

STRANG, G. 2006. Linear algebra and its applications, Thomson, Brooks/Cole, Belmont, CA. ISBN 0-030-10567-6.

TEMPO, R. & ISHII, H. 2007. Monte Carlo and Las Vegas Randomized Algorithms for Systems and Control*: An Introduction. *European journal of control,* 13**,** 189-203.

Classification of emotions by Artificial Neural Networks: a comparative study

Fernando J. Muñoz*, Emanuel B. Tello*, Daniel Patiño, Flavio Roberti**, Elisa Pérez*, Natalia M. López***

**Universidad Nacional de San Juan. Facultad de Ingeniería. Gabinete de Tecnología Médica - San Juan, Argentina.*

***Universidad Nacional de San Juan. Facultad de Ingeniería. Instituto de Automática - San Juan, Argentina.*

Abstract: Every emotion evidences a biological sign while predisposes the body to a different kind of response. In Human-Computer interaction area, the voice recognition techniques are widely used in text engines. In this context, the automatic emotion recognition of the speech aims at identifying the emotional or physical condition of a human being from his voice. Both emotional and physical states of a speaker are included in so-called paralinguistic aspects. In this work we used a speech database containing 7 different emotions in which 4 emotions were selected and 12 features were extracted. Emotions were classified by different types of neural networks in order to compare the efficiency of them to discriminate different moods.

Keywords: Classifiers, Speech recognition, Neural-Network Models, Emotions, Audio Features.

1. INTRODUCTION

Emotions govern almost all modes of human communication: facial expressions, gesture, posture, voice tone, words, breathing, body temperature, etc. Human perception of emotions is about 55 % from facial expressions, and 38 % from 7% speech from text (Busso et al., 2004). Although the emotional state does not alter the linguistic content, this is an important human communication factor as it provides feedback information for the development of applications in a wide spectrum, such as assistive technologies, psychiatric diagnosis and lie detection (Nasr and Ouf, 2012).

Emotion recognition methods require the extraction of certain characteristics of speech (Espinosa and Reyes García, 2010). Some are based on acoustic features such as Mel Frequency Cepstrum Coefficients (MFCC) and the Fundamental Frequency to detect emotional signs. Others use prosodic features in speech to achieve higher classification accuracy.

Many researches focus on the classification using artificial neural networks (ANN) (Pérez-Gaspar et al., 2015). In this work different topologies of ANN based on naive Bayes (NB), Support Vector Machines (SVM), Multi-layer Perceptron (MLP) and Radial Basis Function (RBF) classifiers were designed in order to compare and classify a set of audio signals from Berlin database, which show opposite emotional states. A total of 12 features were taken from the audio signals. The performance of the classifiers was evaluated through 2 stages. In the first stage, three emotions (happiness, sadness, and neutral) were selected. In the second stage, a fourth emotion (fear) was added. The ANNs were adjusted to achieve optimal classification. Finally the comparison between the different hit rates obtained by the classifiers was established in order to evidence the performance of classifiers and the capacity of description of the features set used.

2. MATERIALS AND METHODS

2.1 Database

The Berlin database was used, containing about 500 speech audio files performed by 10 different actors, expressing different emotions: happiness, anger, sadness, fear, boredom and disgust, as well as a neutral expression (serene). There are a total of ten different phrases. Four emotions of the database were selected: happiness, sadness, fear and neutral. The choice was based on the regionalization proposed in Russell circumplex model (Figure 1) in which each selected emotion characterizes one quadrant model (Russell, 1980).

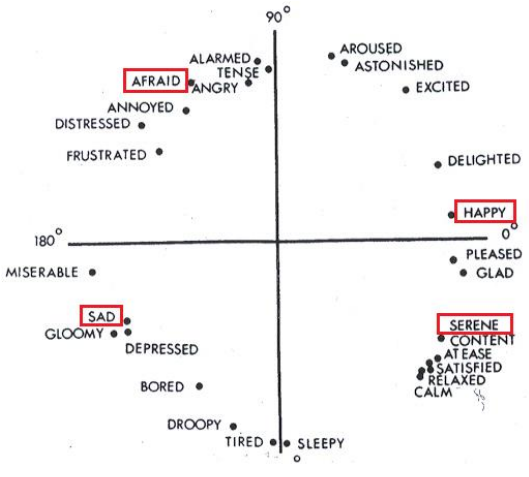


Fig 1. Russell Circumplex model (Russell, 1980)

2.2 Feature Extraction

A total of 12 Features of the audio signals were extracted and divided into 3 groups: prosodic, temporary and frequency features. The number of selected parameters is based on recent literature of emotions and audio signal processing (Bustamante et al., 2015).

The statistical and temporal parameters were mean (ME), variance (VAR), standard deviation (SD), zero crossings (ZC) and kurtosis (K). Prosody has a very important paralinguistic role that complements the linguistic message and reflects the emotional state of the speaker (Espinosa, 2010). Prosodic parameters were based on the energy (E) related to the intensity of the audio signal, the duration of pauses and phonemes (D) and the fundamental frequency (pitch). In frequency domain the extraction of formant, frequency cepstral coefficients in Mel scale (MFCC) and power spectral density (PSD) was performed.

2.3 ANN structures

This work was made in two stages in order to analyze and track changes in the performance of classifiers when a new emotion is introduced. As pre-processing of the audio signal, an initial re-sampling to 16 KHz was performed. The signal was normalized and limited in a frequency band by a recursive Chebyshev type 1 band pass filter, with a lower cut-off frequency at 20 Hz and upper cut-off frequency at 6800 Hz.

A voice detector was applied to the filtered signal in order to extract the voice segments and eliminate silences. Finally it was carried out a windowing of the signal, setting the limit at 20 ms. with 50% overlap.

The structures for the four models of ANN selected were established. 105 samples for the first stage and 140 for the second stage were taken from database as training set (35 samples of each emotion: happiness, sadness, neutral and fear). The input data, corresponding to the 12 selected features, were previously normalized. Modelling of different ANNs was performed in MATLAB®.

2.3.1 Multi-Layer Perceptron

In the first stage, a MLP network was configured with 12 input nodes, 2 hidden layers and three output nodes. Activation functions employed were established empirically, defining hyperbolic functions for the hidden layers and logistic functions for the output layer. The Levenberg-Marquardt was implemented as learning rule. For each emotion only one output is in high state (Y=1), while the others remain in the low state (Y=0). In the second stage, was added a new output node in order to identify the emotion "fear", then was applied the same process mentioned before. The Figure 2 shows the topologies used for these architectures.

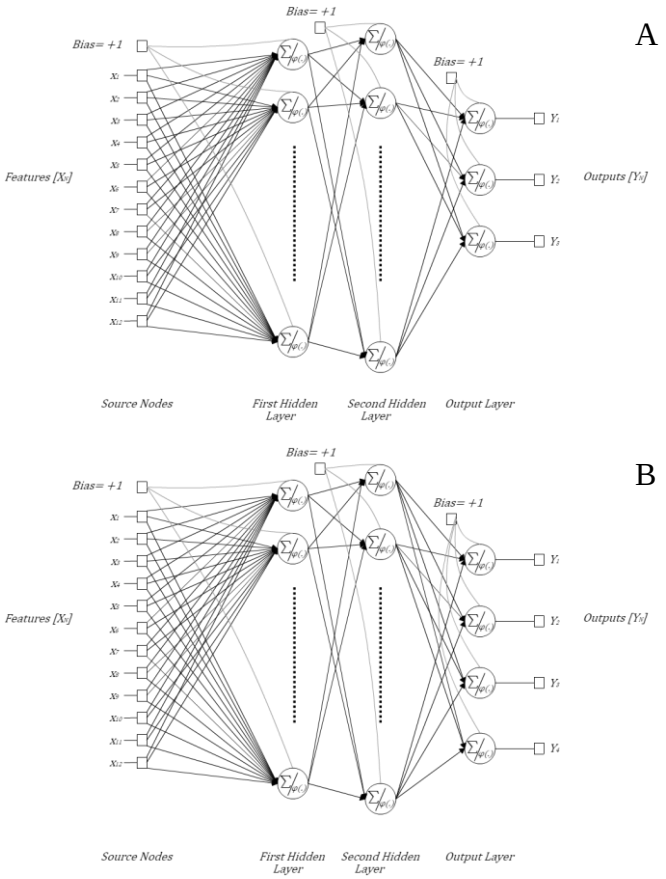


Fig 2. MLP structures for three (A) and four (B) emotions.

2.3.2 Radial Basis Function

The RBF network was designed with 12 input nodes and one output neuron (Figure 3). It can be considered as a special MLP network of three layers. Neurons in the hidden layer containing Gaussian transfer functions whose outputs are inversely proportional to the distance from the centre of the neuron. The activation function for the output node presents a linear behaviour.

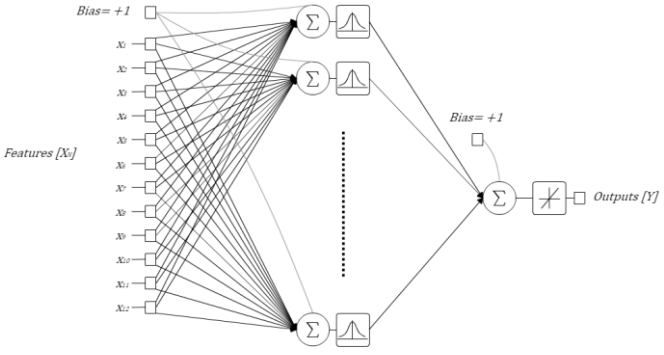


Fig 3. RBF structure.

2.3.3 Support Vector Machines

SVM is a type of supervised learning machine that belongs to the category of linear classifiers, since they induce linear separators or hyperplanes, either in the original space of input

examples, if they are separable or quasi-separable, or in a transformed space (feature space), if the examples are not linearly separable in the original space. In these cases, the search for the hyperplane will be done implicitly using kernel functions (Chavan and Gohokar, 2012).

In this work, a total of three SVM classifiers (Figure 4) were designed based on combinations of the emotions selected: Happiness/Sadness; Happiness/Neutral; Sadness/Neutral. After training, different kernels of inner product were used (Linear, Quadratic, Polynomial, Radial Basis Function and Multilayer Perceptron) to analyse which of them provided a better performance in data classification. Then, three new classifiers were designed to include the emotion "Fear": Happiness/Fear; Neutral/Fear; Sadness/Fear.

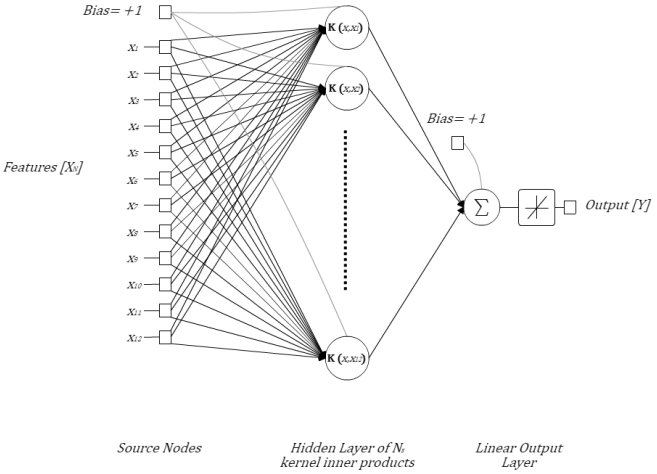


Fig 4. SVM structure.

2.3.4 Naive Bayes

The NB network was configured as a typical Naive Bayes classifier (Figure 5) with three possible classes, corresponding to the emotions selected for the first stage. Uniform probabilities were supposed and the algorithm assumes that predictors (inputs) are conditionally independent given the class (Castillo Reyes et al., 2014; Vinay et al., 2013).

For the second stage, a new class was added. The NB network was then configured with four possible classes: Happiness, Neutral, Sadness and Fear.

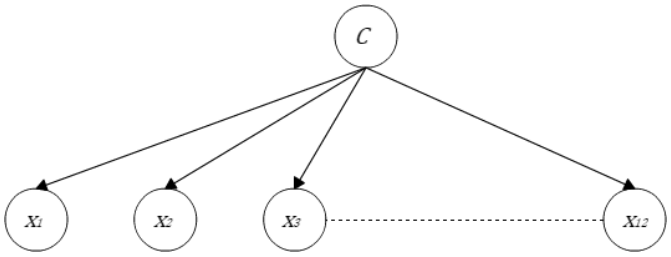


Fig 5. NB classifier.

2.4 Simulation

In the first simulation stage of ANNs a total of 45 known samples of audio were used as test set, picked from Berlin database, 15 for each emotion (happiness, sadness and neutral). The 12 features of the samples were extracted, normalized and introduced in the different topologies. The estimated outputs were compared with the expected values. The percentages of successes and overall performance for emotions classified in different network models are shown in Table 1.

In the second stage, 35 samples were added to the training set, corresponding with the "fear" emotion and 15 samples were used for simulation. The estimated outputs were compared with the expected values. The percentage of hits and overall performance are shown in Table 2.

Table 1. Hit rate and overall performance, with three emotions.

ANN	Happiness	Sadness	Neutral	Overall Performance
MLP	73.33%	66.67%	66.67%	68.89%
RBF	46.67%	80%	33.33%	53.33%
SVM-L	73.33%	86.66%	93.33%	84.44%
SVM-Q	46.67%	66.67%	53.33%	55.56%
SVM-P	73.33%	60%	40%	57.77%
SVM-RBF	80%	93.33%	46.67%	73.33%
SVM-MLP	73.33%	66.67%	60%	66.67%
NB	86.67%	93.33%	80%	86.67%

Table 2. Hit rate and overall performance, with four emotions

ANN	Happiness	Sadness	Neutral	Fear	Overall Performance
MLP	53.33%	60%	80%	46.67%	40%
RBF	33.33%	73.33%	53.33%	33.33%	48.33%
SVM-L	73.33%	86.67%	66.67%	46.67%	68.33%
SVM-Q	20%	66.67%	46.67%	40%	43.33%
SVM-P	53.33%	60%	20%	46.67%	44.99%
SVM-RBF	66.67%	93.33%	33.33%	46.67%	59.99%
SVM-MLP	60%	66.67%	46.67%	33.33%	51.66%
NB	86.67%	86.67%	66.67%	40%	70%

3. RESULTS

The values obtained in Tables 1 and 2 are the result of the optimization of the training process for the designed configurations, based on the analysis and the number of features used as inputs to the ANNs.

In the recognition with three emotions, the SVM-L, SVM-RBF and NB structures showed the best overall performances (over 70 %), with variability in their hits rates according to the emotion considered.

Figure 6 shows the detection rate of classifiers for each emotion. It is observed that the mean values for "happiness" and "sadness" are close to 75%, and the dispersion of classifiers in the recognition of emotion "sadness" is the smallest of the three cases. In the case of the "neutral" emotion, it is evident that the mean value is less than 60% and the dispersion of the hit rate is greater than 50%.

This evidences the relation between the descriptors used as inputs with the performance of the ANNs designed. Temporal and prosodic features, as the zero crossings and the duration of speech segments, are useful in distinguishing the emotion

"sadness" from the others, because usually a speaker in high state of excitation / valence tends to speak quickly with fewer and shorter breaks, while a depressed speaker talks slowly, introducing longer pauses.

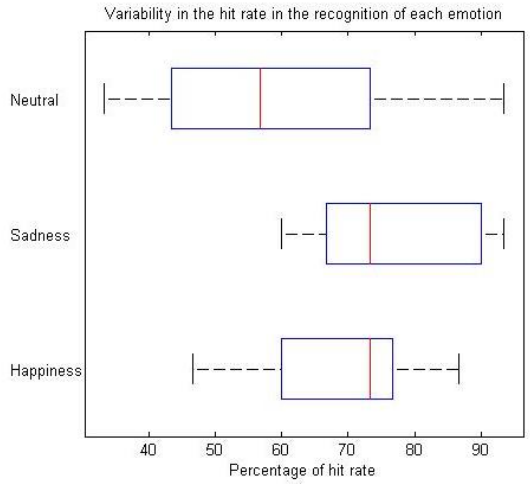


Fig 6. Hit rate variability with three emotions

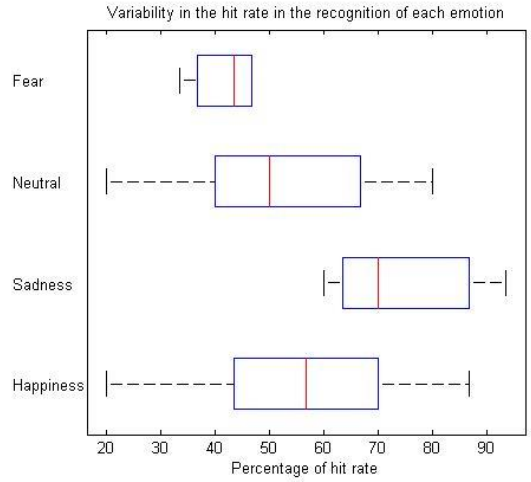


Fig 7. Hit rate variability with four emotions

When a fourth emotion is added (Figure 7), the emotion "sadness" keeps its hit percentage in relation to "happiness", "fear" and "neutral". Table 2 shows that the overall performance for these emotions is about 50%, obtaining in the best cases a percentage of 75%.

Some pairs of emotions are usually confused. This is the case of "happiness" and "fear". The same trend appears between "happiness" and "neutral". Speech associated with "fear" and "happiness" has a longer utterance duration, shorter breaks, higher pitch and energy values with wider ranges. Therefore, these emotions are difficult to be classified.

Although the overall performance decrease when "fear" is incorporated, the classifiers SVM-L, SVM-RBF and NB (that showed the best performances in the first stage) are still higher compared to others, with percentages around 60%. The emotion "sadness" remains as the most recognizable, with a success rate above 70%.

4. CONCLUSIONS

In this paper we have evaluated the performance of several classifiers in distinguishing four emotional states under two stages.

In the analysis with three emotions, for the classifiers NB and SVM-L the overall performances are of 86.67% and 84.44% respectively. With four emotions, the best classifiers remain NB (70%) and SVM-L (68.33%) whose hit rates for "happiness" and "sadness" are above 60 %. Considering the capacity of the classifiers to discriminate between different emotions, it is evident that "sadness" proves to be the most distinguishable, regard to the others. This may indicate the need to incorporate new features of the processed audio signals, or even combine the best classifiers in order to achieve better performances.

This work shows that the performance of classifiers is given by the number and capacity of description of the input features. Although the selected emotional states are located in visually separable regions, according to the Russell model, certain features of the audio signals have similar values, making it difficult to recognize them by ANNs. An increase in the number of descriptors and a deeper understanding analysis of the relations between them and the proposed emotions might be necessary in future works.

REFERENCES

Busso, C., Deng, Z., Yildirim, S., Bulut, M., Lee, C., Kazemzadeh, A., et al. (2004). Analysis of Emotion Recognition Using Facial Expressions, Speech and Multimodal Information. International Commission on Mathematical Instruction, pp. 205-211.

Bustamante P., López N., Perez E., Quinteros L. (2015). Recognition and regionalization of emotions in the arousal-valence plane. Engineering in Medicine and Biology Society (EMBC), 37th Annual International Conference of the IEEE. Milan -Italia.

Castillo Reyes G. et al. (2014). Técnica de clasificación bayesiana para identificar posible plagio en información textual. Cuba cienc informat [online]. 2014, vol.8, n.4, pp. 130-144. ISSN 2227-1899.

Chavan, V.M., Gohokar, V.V. (2012) Speech emotion recognition by using SVM-classifier. International Journal of Engineering and Advanced Technology (IJEAT) 1(5), 11-15

Espinosa H., Reyes García C. (2010). Reconocimiento de Emociones a Partir de Voz Basado en un Modelo Emocional Continuo. Reporte Técnico No. CCC-10-005

Vinay , Shilpi Gupta, Anu Mehra. (2013). Vocal Emotion Recognition using Naïve Bayes Classifier. Proc. of Int. Conf. on Advances in Computer Science, AETACS, 2013

Nasr, M., Ouf, S. (2012). A Proposed Smart E-Learning System Using Cloud Computing Services. PAAS, IAAS and Web 3.0, Helwan University, Helwan, Egypt; iJET, Volume 7, Issue 3, http://dx.doi.org/10.3991/ijet.v7i3.2066

Pérez-Gaspar L., Caballero-Morales S., Trujillo-Romero F. (2015). Integración de optimización evolutiva para el reconocimiento de emociones en voz. Research in Computer Science, Vol.93, pp. 9-21

Russell J. (1980) A Circumplex Model of Affect. Journal of Personality and Social Psychology – Vol.39, Nº6, 1161-1178

Song, M., You, M., Li, N., Chen, C. (2008) A robust multimodal approach for emotion recognition. Neurocomputing 71, 1913-1920.

Control of underactuated unmanned surface vessels with linear flatness-based filters

Bardalez L.*, Sotomayor-Moriano J., Enciso L.*****
**Pontificia Universidad Católica del Perú, Av.Universitaria 1801*
Lima, Perú (e-mail: lbardalezg@pucp.pe).
***Pontificia Universidad Católica del Perú, Av.Universitaria 1801*
Lima, Perú (e-mail: jsotom@pucp.pe).
****Pontificia Universidad Católica del Perú, Av.Universitaria 1801*
Lima, Perú (e-mail: lenciso@pucp.pe).

Abstract: Proper functioning of unmanned surface vessels requires effective trajectory tracking. To achieve this, upon selecting the method for control system design, the knowledge required regarding vessel dynamics described by a model must be taken into account. In this case, it would be advantageous to design the controller requiring the least task effort in modeling for fulfilling the tracking requirements. This paper proposes an active disturbance rejection control system based on linear filters applied to tracking arbitrary trajectories of an underactuated unmanned surface vessel, as an alternative that uses a simple model. Finally, a performance comparison between the proposed control and another control technique (whose design requires a more complex model) is presented.

Keywords: Disturbance rejection, Nonlinear control, Marine systems, Flatness based control.

1. INTRODUCCIÓN

Unmanned surface vessel (USV) control is focused on path-following and trajectory-tracking problems, reducing the USV's tracking error.

Currently, there are many control techniques for fully actuated USV's (Fossen, 2000; Fossen, 2011), where the number of actuators is the same as the number of degrees of freedom; however, underactuated USV (UUSV) control is a field that still requires further investigation as the number of actuators is lesser than the degrees of freedom. The techniques used for the latter can be classified according to the required knowledge level of the mathematical model chosen for the control system design.

In non-linear control, when employing controllers such as those based on the stability method of Lyapunov and additional modifications (Do, 2010; Ding, et. al., 2011; Bao-Li, 2013), backstepping (Ding, et. al., 2011; Yang, et. al., 2014) and successive linearization (Chwa, 2011), complete knowledge of parameters of the UUSV mathematical model is required. In addition, some of these methods (Ding, et. al., 2011; Bao-Li, 2013) do not exhibit robustness to parametric variations or to non-modeled disturbances.

In learning-based control, developing a model based on neural networks requires data acquisition and performing controller optimization, which facilitates its design since it does not depend closely on UUSV modeling. An advantage of these techniques is the adaptability, which provides control under various operating conditions (Dai, et al., 2012; Pan, et al., 2013). The main disadvantage is the computational cost of the adaptation mechanism as well as the number of calculations at each time interval.

In active disturbance rejection control, ultra-local models or dynamic simplified models are used (Li, et. al., 2012; Li, et. al., 2013), avoiding the use of complex mathematical models. The main disadvantage found in these controllers is that, due to their design, the inherent UUSV kinematics is ignored and only a priori known geometrics can be tracked.

This paper presents the design of active disturbance rejection control based on linear flatness-based filters applied on an underactuated unmanned surface vessel (UUSV) to track arbitrary trajectories. This design employs the knowledge of system kinematics and only requires knowledge of dynamic parameters regarding the actuators, which in practice are easily recognizable.

Section 2 describes the system dynamic model. Section 3 includes a review of active disturbance rejection control, emphasizing the robust generalized proportional-integral controller as a linear flatness-based filter. Section 4 presents the controller design for a UUSV. Simulation testing results are analyzed and contrasted with a controller based state-feedback linearization in section 5. Conclusions are described in Section 6.

2. SYSTEM DYNAMIC MODEL

Considering a UUSV with coordinated axes and actuator arrangement as defined in Figure 1, where positions x and y, yaw angle ψ (measured from the X-axis), surge and

displacement speeds *u* and *y*, yaw speed *r*, driving force *F* and steering torque *T*.

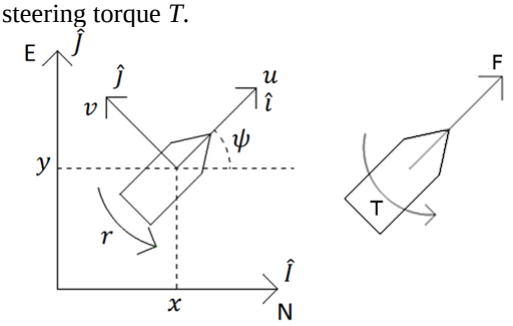


Figure1. Bottom view of coordinated system (left) and actuator arrangement (right).

The dynamic model of this UUSV (Fossen, 2011) is presented in (1)

$$\begin{aligned} \dot{\eta} &= R(\psi)\nu \\ M\dot{\nu} + C(\nu)\nu + D(\nu)\nu &= B_\tau \tau \end{aligned} \tag{1}$$

$$\eta = \begin{bmatrix} x \\ y \\ \psi \end{bmatrix}, \nu = \begin{bmatrix} u \\ v \\ r \end{bmatrix}, \tau = \begin{bmatrix} F \\ T \end{bmatrix}$$

$$R(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$M = \begin{bmatrix} M_{11} & 0 & 0 \\ 0 & M_{22} & M_{23} \\ 0 & M_{23} & M_{33} \end{bmatrix}$$

$$C(\nu) = \begin{bmatrix} 0 & 0 & -M_{22}v - M_{23}r \\ 0 & 0 & M_{11}u \\ M_{22}v + M_{23}r & -M_{11}u & 0 \end{bmatrix}$$

$$D(\nu) = \begin{bmatrix} D_{11}(u) & 0 & 0 \\ 0 & D_{22}(v,r) & D_{23}(v,r) \\ 0 & D_{32}(v,r) & D_{33}(v,r) \end{bmatrix}; \quad B_\tau = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Where $R(\psi)$ is rotation matrix from the UUSV system fixed $\hat{\imath} - \hat{\jmath}$ to the inertial system $\hat{I} - \hat{J}$, M is the full mass matrix, $C(v)$ is the Coriolis matrix, $D(v)$ is the damping matrix composed of non-linear functions and B_τ is the geometric matrix of system actuators.

This mathematical model was chosen since it adequately approximates the actual UUSV dynamics and it will make possible to illustrate the effectiveness of the proposed controller exposed to a highly non-linear dynamic behavior.

In the case of fully actuated USV's, the model would also include a force *H* applied in the transversal axis (Wondergem et. al. 2011); however, in small USV's, it is common to avoid this configuration because it would reduce available space, increase USV's weight and consume more energy.

3. ACTIVE DISTURBANCE REJECTION CONTROL

3.1. Active disturbance rejection control

The dynamics of a system of order *n* can be expressed according to (2), where μ is the control action, *b* is a system constant; and *f(...)* is a function that depends on unknown and non-linear system dynamics and on its endogenous and exogenous disturbances *w*:

$$x^{(n)} = f\left(x, \dot{x}, \ldots, x^{(n-1)}, u, w\right) + b\mu \tag{2}$$

There are different alternatives to control (2) that are based on active disturbance rejection control (ADRC), where the objective is to eliminate the term *f(...)* from the dynamics in order to simplify controller design. The best known solution entails the use of extended state observers (ESO), which estimates the term *f(...)* and the derivatives of *x* (Gao, 2006; Han, 2009). Another approach is to use a model-free control through intelligent PID controllers (Fliess, 2013), in which the term *f(...)* is estimated and eliminated through integrals of variables *x* and *u*. It is also possible to perform control without the estimation of the term *f(...)* through the use of GPI controllers (Sira et. al., 2010). These controllers are robust to polynomial disturbances in time (Morales and Sira, 2010); however, they are not robust against complex disturbances. For the latter case, in (Sira, et. al., 2008), a robust GPI controller that includes an integral action in the resulting transfer function is proposed and, in (Sira, et. al, 2010), a further order reduction is proposed.

3.2. Robust GPI Controller

The general structure of a robust GPI controller can be generalized as in (3) (Luviano-Juárez, et. al., 2008), where *n* is the order of the system, *m* is the order considered in the disturbance *f(...)* and e_x is the tracking error with respect to reference x^*; thus, $e_x = x - x^*$.

$$\mu = -\frac{1}{b}\frac{\lambda_{n+m-1}s^{m+n-1} + \lambda_{n+m-2}s^{m+n-2} + \cdots + \lambda_1 s + \lambda_0}{s^m\left(s^{n-1} + \lambda_{2n+m-2}s^{n-2} + \cdots + \lambda_{n+m+1}s + \lambda_{n+m}\right)}e_x \tag{3}$$

For the particular case of a system with first order dynamics and first order (constant) disturbance, the controller in (4) is used, where closed-loop dynamics can be chosen by selecting ζ and ω_n; and the corresponding controller parameters can be obtained from (5).

$$\mu = -\frac{1}{b}\frac{\lambda_1 s + \lambda_0}{s}e_x \tag{4}$$

$$s^2 + \lambda_1 s + \lambda_0 \equiv s^2 + 2\zeta\omega_n + \omega_n{}^2 \tag{5}$$

Similarly, for a system with second order dynamics and first order (constant) disturbance, the controller is shown in (6) and the characteristic equation in (7).

$$\mu = -\frac{1}{b}\frac{\lambda_2 s^2 + \lambda_1 s + \lambda_0}{s(s + \lambda_3)}e_x \tag{6}$$

$$s^4 + \lambda_3 s^3 + \lambda_2 s^2 + \lambda_1 s + \lambda_0 \equiv \left(s^2 + 2\zeta\omega_n + \omega_n{}^2\right)^2 \tag{7}$$

3.3. Configuration of a linear flatness-based filter

A flat system is a system whose inputs and outputs can be expressed in function of a variable and its derivatives (Sira, et. al. 2004). Without loss of generality, the controller shown in (6) is used. Thus, the robust GPI controller can be seen as a flat system (Sira, et. al, 2010) since, through definition of the filtered error e_f in (8), tracking error e_x and the control action u can be expressed as a function of e_f as in (9) and (10). As a result, the controller can be expressed as an integrator chain in (11).

$$e_f = \frac{1}{s(s+\lambda_3)} e_x \tag{8}$$

$$e_x = \ddot{e}_f + \lambda_3 \dot{e}_f \tag{9}$$

$$\mu = -\frac{1}{b}\left(\lambda_2 \ddot{e}_f + \lambda_1 \dot{e}_f + \lambda_0 e_f\right) \tag{10}$$

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= -\lambda_3 z_2 + e_x \\ \mu &= \frac{1}{b}\left((\lambda_2\lambda_3 - \lambda_1)z_2 - \lambda_0 z_1 - \lambda_2 e_x\right) \end{aligned} \tag{11}$$

4. CONTROL SYSTEM DESIGN

4.1. Problem formulation

The goal is that a UUSV can follow arbitrary trajectories defined in time without restrictions in their form and with speed of approximately 1 *m/s.*, x_d and y_d will be used to denote a point of the desired trajectory at a given time instant. The proposed control system is presented in Figure 2.

4.2. Trajectory Controller Design

In the dynamic model presented in (1) the states derivatives can be isolated (12).

$$\begin{aligned} \dot{x} &= u\cos\psi - v\sin\psi \\ \dot{y} &= u\sin\psi + v\cos\psi \\ \dot{\psi} &= r \\ \dot{u} &= U_u(u)u + U_{vr}vr + U_{rr}r^2 + U_F F \\ \dot{v} &= V_v(v,r)v + V_r(v,r)r + V_{uv}uv + V_{ur}ur + V_T T \\ \dot{r} &= R_v(v,r)v + R_r(v,r)r + R_{uv}uv + R_{ur}ur + R_T T \end{aligned} \tag{12}$$

U_i, V_j and R_k can be obtained from (1); however, in accordance with ARDC, (12) can be taken to the form (13), where its values are not considered.

$$\begin{aligned} \dot{x} &= u\cos\psi - v\sin\psi \\ \dot{y} &= u\sin\psi + v\cos\psi \\ \dot{\psi} &= r \\ \dot{u} &= \cdots + U_F F \\ \dot{v} &= \cdots \\ \dot{r} &= \cdots + R_T T \end{aligned} \tag{13}$$

Defining the reference trajectory x_r and y_r, generated from the desired trajectory and errors regarding the trajectory in reference $e_x = x - x_r$ and $e_y = y - y_r$, the dynamics of these errors can be expressed by (14). It can be observed that the kinematics of the UUSV naturally appears in these dynamics, which obliges the use of such structure in the controller design.

$$\begin{bmatrix} \ddot{e}_x \\ \ddot{e}_y \end{bmatrix} = \begin{bmatrix} U_F\cos\psi & -u\sin\psi \\ U_F\sin\psi & u\cos\psi \end{bmatrix} \begin{bmatrix} F \\ r \end{bmatrix} + \begin{bmatrix} \cdots - \dot{v}\sin\psi - vr\cos\psi - \ddot{x}_r \\ \cdots + \dot{v}\cos\psi - vr\sin\psi - \ddot{y}_r \end{bmatrix} \tag{14}$$

It is considered that r varies faster than the remaining variables and can be used as an intermediate control; also, the control of r is carried out using torque T.

Using (14) the control can be decoupled through (15). In addition, the remainder terms in (14) can be encapsulated in disturbance dynamics, resulting in decoupled dynamics (16).

$$\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix} = \begin{bmatrix} U_F\cos\psi & -u\sin\psi \\ U_F\sin\psi & u\cos\psi \end{bmatrix} \begin{bmatrix} F \\ r \end{bmatrix} \tag{15}$$

$$\begin{bmatrix} \ddot{e}_x \\ \ddot{e}_y \end{bmatrix} = \begin{bmatrix} \mu_x + f_x(\eta, v, \tau) \\ \mu_y + f_y(\eta, v, \tau) \end{bmatrix} \tag{16}$$

In this manner, the system can be controlled by two independent control laws (17) similar to (6)

$$\begin{aligned} \mu_x &= -\frac{\lambda_2 s^2 + \lambda_1 s + \lambda_0}{s(s+\lambda_3)} e_x, \\ \mu_y &= -\frac{\lambda_2 s^2 + \lambda_1 s + \lambda_0}{s(s+\lambda_3)} e_y, \end{aligned} \tag{17}$$

where parameters are selected using (7) and must be the same for both controllers since similar responses are desired in both directions.

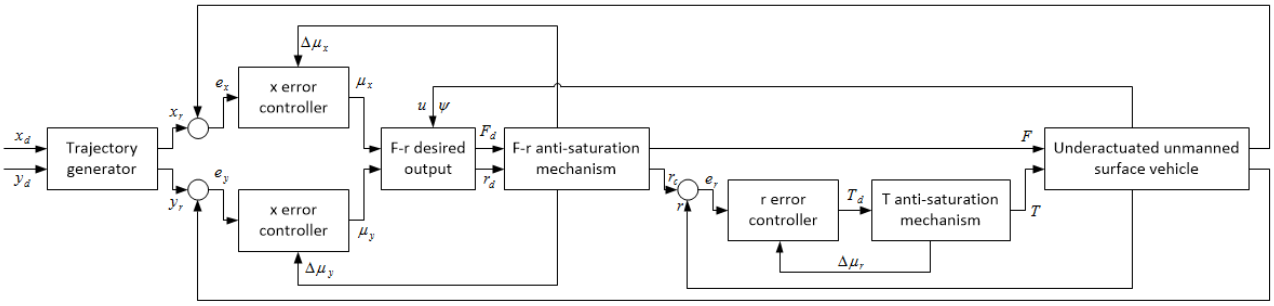


Figure 2. Proposed control system.

The desired control action is isolated from (15), obtaining (18); due to the presence of u in the denominators, its value in the controller should be set in a range between u_{min} and u_{max} using the definition of the function (19). The values $u_{min} > 0$ and u_{max} are chosen such that the desired performance of r is not greatly reduced, thus the desired control can be expressed by (20).

$$\begin{bmatrix} F_d \\ r_d \end{bmatrix} = \begin{bmatrix} \frac{\cos\psi}{U_F} & \frac{\sin\psi}{U_F} \\ -\frac{\sin\psi}{u} & \frac{\cos\psi}{u} \end{bmatrix} \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix} \quad (18)$$

$$u_p = sat(u, u_{\min}, u_{\max}) = \begin{cases} u_{\min}, u < u_{\min} \\ u, u_{\min} \le u \le u_{\max} \\ u_{\max}, u_{\max} < u \end{cases} \quad (19)$$

$$\begin{bmatrix} F_d \\ r_d \end{bmatrix} = \begin{bmatrix} \frac{\cos\psi}{U_F} & \frac{\sin\psi}{U_F} \\ -\frac{\sin\psi}{u_p} & \frac{\cos\psi}{u_p} \end{bmatrix} \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix} \quad (20)$$

In contrast, fully actuated USV's model would present F and H in (13), resulting (21). Errors could be controlled using both actuators, so r would not be considered as an actuator and ψ could be freely controlled by T. Each controller can be design using (6) and (7).

$$\begin{bmatrix} \ddot{e}_x \\ \ddot{e}_y \end{bmatrix} = \begin{bmatrix} U_F\cos\psi & -V_H\sin\psi \\ U_F\sin\psi & V_H\cos\psi \end{bmatrix} \begin{bmatrix} F \\ H \end{bmatrix} + \begin{bmatrix} \cdots - \dot{v}\sin\psi - vr\cos\psi - \ddot{x}_r \\ \cdots + \dot{v}\cos\psi - vr\sin\psi - \ddot{y}_r \end{bmatrix} \quad (21)$$
$$\ddot{\psi} = \cdots + R_T T$$

4.3. Design of angular velocity controller

Since the angular velocity r was considered to behave as a system actuator, this should be controlled on its respective subsystem. Using the torque T as an actuator in

$$\dot{e}_r = \cdots + R_T T, \quad (22)$$

where e_r is the error between the angular velocity measured in the system and the desired angular velocity (20). In a similar way to (16), the subsystem is presented based on a change of variable and a term of disturbance dynamics

$$\dot{e}_r = \mu_r + f_r(\eta, \nu, \tau). \quad (23)$$

Since the subsystem is a first order system, the control law (4) will be used as well as the selection of parameters from (5)

$$\mu_r = -\frac{\gamma_1 s + \gamma_0}{s} e_r, \quad (24)$$

$$s^2 + \gamma_1 s + \gamma_0 \equiv s^2 + 2\zeta_r \omega_r + {\omega_r}^2, \quad (25)$$

The desired torque is calculated by

$$T_d = (1/R_T)\mu_r. \quad (26)$$

4.4. Anti-saturation mechanism

Controllers using integral terms often present saturation, resulting in performance reduction. An anti-saturation mechanism is used to avoid this problem, using the projection of saturated actuator errors (27) with respect to the desired actuator values (20) on the decoupled control presented in (17). Consequently, the corrections are calculated by (28), where β_F and β_r are parameters that determine the aggressiveness of the anti-saturation mechanism and $\Delta\mu_x$ and $\Delta\mu_y$ are the required corrections to be applied in (17).

$$F = sat(F_d, F_{\min}, F_{\max}), \qquad r = sat(r_d, r_{\min}, r_{\max}) \quad (27)$$

$$\begin{bmatrix} \Delta\mu_x \\ \Delta\mu_y \end{bmatrix} = \begin{bmatrix} U_F\cos\psi & -u\sin\psi \\ U_F\sin\psi & u\cos\psi \end{bmatrix} \begin{bmatrix} \beta_F(F - F_d) \\ \beta_r(r - r_d) \end{bmatrix} \quad (28)$$

In this manner, the controllers (17) expressed in form of linear flatness-based filters (11) including the anti-saturation mechanism, will present the following form

$$\begin{aligned} \dot{p}_1 &= p_2 - \Delta\mu_x \\ \dot{p}_2 &= -\lambda_3 p_2 + e_x \\ \mu_x &= (\lambda_2\lambda_3 - \lambda_1)p_2 - \lambda_0 p_1 - \lambda_2 e_x, \\ \dot{q}_1 &= q_2 - \Delta\mu_y \\ \dot{q}_2 &= -\lambda_3 q_2 + e_y \\ \mu_y &= (\lambda_2\lambda_3 - \lambda_1)z_2 - \lambda_0 z_1 - \lambda_2 e_y. \end{aligned} \quad (29)$$

In addition, because torque T may also present saturation

$$T = sat(T_d, T_{\min}, T_{\max}), \quad (30)$$

an anti-saturation mechanism is provided based on the difference between the effective torque and the desired torque

$$\Delta\mu_r = R_T\beta_T(T - T_d), \quad (31)$$

so the resulting controller in form of a linear flatness-based filter is

$$\begin{aligned} \dot{h}_1 &= e_r - \Delta\mu_r \\ \mu_r &= -\gamma_1 e_r + \gamma_1\Delta\mu_r - \gamma_0 m_1 \end{aligned} \quad (32)$$

4.5. Trajectory generator

Since the initial values of the desired trajectory x_d, y_d may be distant from the UUSV's origin point, a trajectory reference generator x_r, y_r is provided, based on a simplified mathematical model of the UUSV in the form

$$\begin{aligned} \dot{x}_r &= u_r\cos\psi_r \\ \dot{y}_r &= u_r\sin\psi_r \\ \dot{\psi}_r &= \omega \\ \dot{u}_r &= a \end{aligned} \quad (33)$$

where the control objective is that x_r, y_r follow x_d, y_d. The controller for the simplified UUSV (33) can be developed using a procedure similar to the one used for the original UUSV.

5. SIMULATION RESULTS AND ANALYSIS

The parameters of the vehicle in (Wondergem et. al. 2011) were used to carry out simulations:

$$M = \begin{bmatrix} 25.8 & 0 & 0 \\ 0 & 33.8 & 1.0115 \\ 0 & 1.0115 & 2.76 \end{bmatrix}$$

$$C(\nu) = \begin{bmatrix} 0 & 0 & -33.8v - 1.0115r \\ 0 & 0 & 25.8u \\ 33.8v + 1.0115r & -25.8u & 0 \end{bmatrix}$$

$$D(\nu) = \begin{bmatrix} 0.72 + 1.33|u| & 0 & 0 \\ 0 & 0.86 + 36.28|v| & -0.11 \\ 0 & -0.11 - 5.04|v| & 0.5 \end{bmatrix}$$

Table 1 shows the parameters used for the UUSV controller and the trajectory generator. All parameters are expressed in IS units.

Table 1. Controller design parameters

USV		Trajectory generator	
Z	1.2	ζ	1
ω_n	1	ω_n	1
F_{min}	-10	a_{min}	-0.5
F_{max}	10	a_{max}	0.5
r_{min}	-2	ω_{min}	-1
r_{max}	2	ω_{max}	1
u_{min}	1	u_{min}	0.01
u_{max}	1	u_{max}	2
β_F	0.5	β_F	0.5
β_r	0.5	β_r	0.5
ζ_r	0.7		
ω_r	5		
T_{min}	-10		
T_{max}	10		
β_T	0.05		

The trajectory used in the simulation, using a simulation time of 100 seconds, was

$$\begin{aligned} x_d &= 20\cos(t/30)\cos(t/20), \\ y_d &= 20\cos(t/30)\sin(t/20). \end{aligned} \qquad (34)$$

In addition, a variable external force with inclination of 10° in respect to axis X was applied.

The proposed controller was compared with a controller based on state-feedback linearization (Fahimi, 2008), the latter requiring knowledge of all model (1) parameters for its design.

In the following figures, FPL is the controller based on linear flatness-based filters, LRE is the controller based on state-feedback linearization, and DT is the desired trajectory.

Figure 3 shows trajectory tracking performed by both controllers. It shows that the LRE converges faster than FPL; however, after the approach, both properly follow the trajectory. Figure 4 shows the tracking of position variables separately.

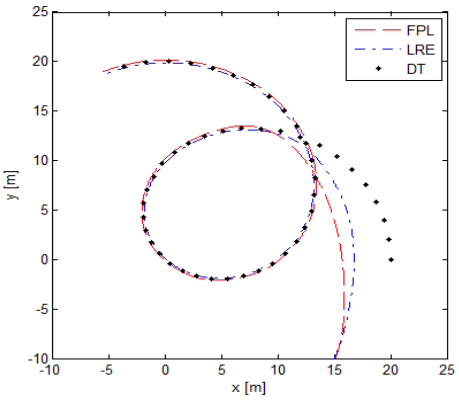


Figure 3. Trajectory tracking

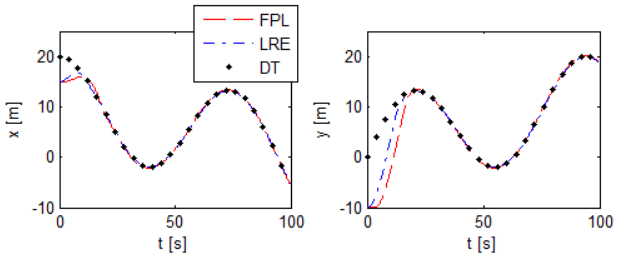


Figure 4. Position evolution

Figure 5 shows that the FPL presents greater error at the beginning; this is due to the use of the trajectory generator, which limits the distance to the reference regarding the original position of the UUSV. The zoom in Figure 5 reveals that both controllers have similar absolute errors after approaching to the trajectory. However, oscillations are found in the FPL error since the term *f(...)* is not constant (contrary to what was considered in (4) and (6)).

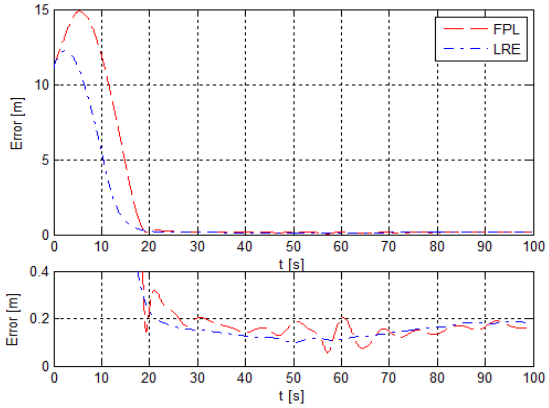


Figure 5. Absolute tracking error

Figure 6 presents the applied external disturbance and the resulting control action for times lesser than 30 seconds. A delay time in the FPL is evidenced; however, it is observed that the latter does not exhibit large oscillations in the torque applied, in comparison to the LRE. For times greater than 30 seconds, the control action in both cases follows the same tendency.

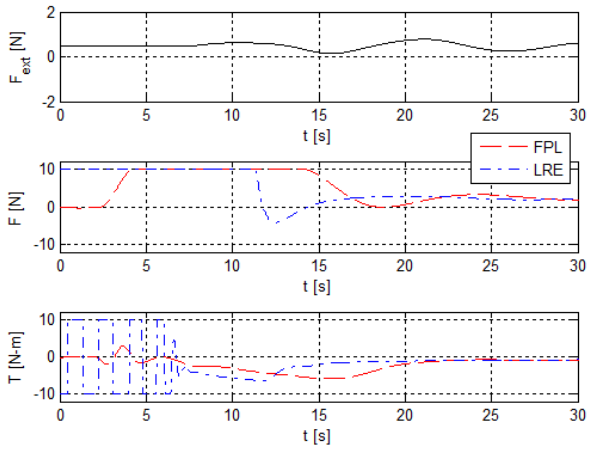


Figure 6. Control action ($t < 30$)

6. CONCLUSIONS

A control system for a UUSV was proposed, with a trajectory controller and an angular velocity controller for tracking arbitrary form trajectories. The proposed design of the system controllers was carried out using active disturbance rejection control based on linear flatness-based filters, requiring a deeper analysis compared to the fully actuated case. Comparisons were made between the trajectory tracking of the FPL control proposed and the LRE control, obtaining similar results; however, the torque applied in the FPL control shows less oscillation. The main advantage of the proposed control with respect to the LRE and other control techniques is that its design only requires knowledge of the dynamic parameters related to the actuators (in practice easily recognizable). A future paper will propose a strategy of trajectory generation and develop methodology for calculation of parameters. Given that the structure kinematics as presented is akin to a group of marine, land and air systems, it is recommended to assess the implementation of the proposed controller in these systems.

REFERENCES

Bao-Li, M.; Wen-Jing, X. (2013) Global asymptotic trajectory tracking and point stabilization of asymmetric underactuated ships with non-diagonal inertia/damping matrices. *International Journal of Advanced Robotic Systems*, vol. 10.

Chwa, D. (2011). Global tracking control of underactuated ships with input and velocity constraints using dynamic surface control method. *IEEE Transactions on Control Systems Technology*, 19(6), pp 1357-1370.

Dai, S. L.; Wang, C.; Luo, F. (2012). Identification and learning control of ocean surface ship using neural networks. *IEEE Transactions on Industrial Informatics*, 8(4), pp 801-810.

Ding, F.; Wang, Y.; Wang, Y. (2011). Trajectory-tracking controller design of underactuated surface vessels. *IEEE OCEANS 2011*, pp 1-5.

Do, K. D. (2010). Practical control of underactuated ships. *Ocean Engineering*, 37(13), pp 1111-1119.

Fahimi, F. (2008) *Autonomous robots: modeling, path planning, and control*. Springer Science & Business Media.

Fliess, M., Join, C. (2013). *Model-free control*. International Journal of Control, 86(12), pp 2228-2252.

Fossen, T. (2000). A survey on nonlinear ship control: from theory to practice. *Proceedings of the 5th IFAC Conference on Manoeuvring and Control of Marine Craft*, pp 23-25

Fossen, T. (2011). *Handbook of marine craft hydrodynamics and motion control, first edition*, John Wiley & Sons, UK.

Gao, Z. (2006). Active disturbance rejection control: a paradigm shift in feedback control system design. *American Control Conference, 2006*: pp 7.

Han, J. (2009). From PID to active disturbance rejection control. *IEEE transactions on Industrial Electronics, 56(3)*, pp 900-906.

Li, R.; Li, T.; Zheng, Q.; Li, Q. (2012). Ship tracking control based on linear active disturbance rejection control. *2012 Third International Conference on Intelligent Control and Information Processing (ICICIP)*, pp 201-205.

Li, R.; Li, T.; Bu, R. (2013). Disturbance decoupling control based trajectory tracking for underactuated ships. *2013 32nd Chinese Control Conference (CCC)*, pp 8108-8113.

Luviano-Juárez, A., Cortés-Romero, J., & Sira-Ramírez, H. (2008). *Chaotic synchronization between oscillators using robust GPI control*. 5th International Conference on Electrical Engineering, Computing Science and Automatic Control. CCE 2008. pp. 114-119. IEEE.

Morales, R., & Sira-Ramirez, H. (2010). Trajectory tracking for the magnetic ball levitation system via exact feedforward linearisation and GPI control. *International Journal of Control*, 83(6), pp 1155-1166.

Pan, C. Z.; Lai, X. Z.; Yang, S. X.; Wu, M. (2013). An efficient neural network approach to tracking control of an autonomous surface vehicle with unknown dynamics. *Expert Systems with Applications*, 40(5), pp 1629-1635.

Sira-Ramirez, H., & Agrawal, S. K. (2004). *Differentially flat systems*. CRC Press.

Sira-Ramirez, H., Beltrán-Carbajal, F., & Blanco-Ortega, A. (2008). A generalized proportional integral output feedback controller for the robust perturbation rejection in a mechanical system. *STA*, 5(4), pp 24-32.

Sira-Ramirez, H; Luviano-Juarez, A.; Cortés-Romero, J. (2010). A generalized proportional integral approach to sliding mode controller design in switched systems. In: *2010 49th IEEE Conference on Decision and Control (CDC)*. IEEE, p 5092-5097.

Yang, Y.; Du, J., Liu, H.; Guo, C.; Abraham, A. (2014). A trajectory tracking robust controller of surface vessels with disturbance uncertainties. *IEEE Transactions on Control Systems Technology*, 22(4), pp 1511-1518.

Wondergem, M., Lefeber, E., Pettersen, K. Y., & Nijmeijer, H. (2011). Output feedback tracking of ships. IEEE Transactions on Control Systems Technology, 19(2), pp 442-448.

Convergence Time Estimation for Jumping Mechanical Systems using Twisting Algorithm ⋆

Raúl Santiesteban Cos, Manuel Beltran Cardenas *
Guillermo Rubio Astorga **

* *Department of Metal-Mecánica. Tecnológico Nacional de Mexico, Campus Culiacán.*
e-mail:raulsncos@gmail.com, manuel.beltranc@hotmail.com
** *Department of Electronics. TNM, Culiacán.*
e-mail:guillermo.rubio@itculiacan.edu.com.mx

Abstract. In this paper, a double integrator affected by bounded external perturbations and jumps in the velocity are considered. A second order sliding mode, such as Twisting control synthesis is presented in the presence of uniformly bounded persistent disturbances and rigid body inelastic impacts. First, a nonsmooth state transformation is employed to transform the original system into a jump-free system. Second, a nonsmooth strict Lyapunov function is identified to establish global finite time stability of the transformed system. Third, a Twisting plus PD control law is analyzed and global finite time stability of the origin of the system with velocity jumps is established without having to analyze the Lyapunov function at the jump instants. An upper bound of the convergence time is estimated for both closed loop systems, in spite of bounded external perturbations and jumps in the velocity.

Keywords: Sliding mode control; Stability analysis; Lyapunov methods.

1. INTRODUCTION

Second Order Sliding Mode Algorithms (SOSM) have become very important for variable structure control theory and engineering applications, because of their properties such as finite time convergence in spite of the presence of external uncertainties (see for example Fridman and Levant [2002], Shtessel et al. [2007], Emelyanov et al. [1986]). Moreover, in the last years, nonsmooth strict Lyapunov functions have been studied in order to analyse their properties from a different point of view.

A linear double integrator, *i.e.* a mechanical system, is considered with jumps in its velocity when it hits a constraint surface. The velocity undergoes an instantaneous jump when the inelastic collision occurs. In this paper, the restitution in velocity, representing loss of energy which occurs at the time of impact, is considered fully known. It is clear from the literature (for example see Cortes [2008], B. Brogliato and Orhant [1997] and Brogliato [1999]) that a jump in the Lyapunov function occurs whenever the velocity undergoes a jump. Therefore the Lyapunov stability needs to be specifically defined for all possible jumps in the Lyapunov function. The rigorous theoretical developments in the theory of nonsmooth mechanics have been accompanied by applications such as biped robots (see for example Bourgeot and Brogliato [2005], J. Grizzle and Plestan [2001] and Y. Hurmuzlu and Brogliato [2004]).

The existing approaches (Orlov [2005], Polyakov and Poznyak [2009], Levant [1993] and Santiesteban et al. [2010]) do not apply to the case of jumps in the velocity dynamics. In H. B. Oza and Spurgeon [2014], the effect of the jump in velocity is considered as a destabilizing one for the dynamic system. Then the stability analysis of the closed loop system is important for this kind of applications (see Stewart [2000]).

This paper is based on the ideas of H. B. Oza and Spurgeon [2014], a mechanical system with resets in velocity, affected by bounded external perturbations with two main objectives. First, a nonsmooth state transformation B. Brogliato and Orhant [1997] is utilized to render a jump-free system with its solutions clearly defined in the sense of Filippov Filippov [1988]. Second, as in Santiesteban et al. [2010], the stability analysis of the closed loop system is made within nonsmooth strict Lyapunov methodology for discontinuous systems. There is well know theory that study this concepts, for example Shevitz and Paden [1994], Bacciotti and Ceragioli [1999], Bacciotti and Rosier [2001], Moreno and Osorio [2008]. Indeed, in H. B. Oza and Spurgeon [2014] homogeneity properties are used to analyze the stability of the system and estimate an upper bound for convergence time of the closed loop system. Another work that use homogeneity properties of a discontinuous system is Orlov [2005].

Moreover, based on previews work, such as Santiesteban et al. [2010], a nonsmooth strict Lyapunov function is identified to show global finite time stability of the closed loop system using Twisting algorithm. Furthermore, Twisting algorithm plus PD control law is consider, identifying another nonsmooth strict Lyapunov function, giving proof

⋆ CONACYT grant 53869

of global finite time stability of the transformed and the original impact system without having to analyze the jumps of the Lyapunov function. Finally, both sliding mode synthesis are shown to stabilize the double integrator with impacts in finite time, in spite of bounded external perturbations. Moreover, an estimation of the convergence time of the trajectories of the closed loop systems is obtained.

In section 2 the problem statement is described : the stabilization of an uncertain system, affected with velocity jumps. In order to present the contribution clearly, in section 3, a mathematical background is shown. In section 4, the stability of the perturbed closed loop system, using Twisting algorithm, is under study. In section 5 the closed loop system, using Twisting plus PD control law, is analyzed. In both cases, finite time stability for the closed loop system is concluded. Moreover, in both cases, an upper bound of the convergence time is estimated. In section 6, to support theoretical results, the control laws are implemented in a numerical simulation and in section 7, presents the conclusions of this work. In Appendix I and II, the mathematical proof of the main theorems of this work are presented.

2. PROBLEM STATEMENT

The general model of second-order mechanical systems, written in the state space form is given by

$$\begin{aligned} \dot{x} &= y \\ \dot{y} &= f(x,y) + \tau + \delta(t,x,y) \\ x &\geq 0 \\ y(t_k^+) &= -\overline{e}y(t_k^-) \;\; if \;\; y(t_k^-) < 0, x(t_k) = 0 \end{aligned} \tag{1}$$

where x and y are the position and the velocity respectively, τ is the control input, and the nominal known part of the system dynamics is represented by the function $f(x,y)$, while the uncertainties are concentrated in $\delta(t,x,y)$. t_k is the k^{th} jump time instant where the velocity undergoes a reset or jump, $\overline{e}$ denotes the loss of energy and $y(t_k^+)$ and $y(t_k^-)$ represent the right and left limits respectively of y at the jump time t_k. The third equation represents the dynamics with unilateral constraints on position x (see for example H. B. Oza and Spurgeon [2014]). It is assumed that the jump event occurs instantaneously within an infinitesimally small time and hence mathematically can be represented by *Newton's restitution rule* given by the fourth equality of (1).

For system (1) the following controller design is proposed

$$\tau = (U - f(x,y)) \tag{2}$$

where U is a new input control.

$$U_1 = -\alpha \text{sgn}(x) - \beta \text{sgn}(y) - px - dy \tag{3}$$

with α, β, p and d are positive constants.

Indeed, according to (Orlov [2005], Theorem 4.2), the disturbed system (2-3) renders the finite time stability, regardless of whichever disturbance δ with a uniform upper bound

$$|\delta(x,y)| \leq M, \tag{4}$$

where M is a positive constant, for $t \geq 0$, affects the system provided that

$$0 < M < \beta < \alpha - M \tag{5}$$

The method of nonsmooth transformation Brogliato [1999] is employed to transform the impact system (1-2-3) into a jump free system. Let the nonsmooth transformation be defined as follows:

$$x = |s|, y = R\nu \text{sgn}(s), R = 1 - k\text{sgn}(s\nu), k = \frac{1-\overline{e}}{1+\overline{e}} \tag{6}$$

The variable structure-wise transformed system

$$\begin{aligned} \dot{s} &= R\nu \\ \dot{\nu} &= R^{-1}\text{sgn}(s)\left(U(|s|, R\nu\text{sgn}(s)) + \delta(t)\right) \end{aligned} \tag{7}$$

is then derived by employing (6) (see section 1.4.3 Brogliato [1999]). By combining (6), the controller (3) can be represented in the transformed coordinates as follows:

$$U(|s|, R\nu\text{sgn}(s)) = -\alpha - \beta\text{sgn}(s\nu) - p|s| - dR\nu\text{sgn}(s) \tag{8}$$

Substituting (8) into (7), the closed-loop system in the new coordinate frame can be obtained as follows:

$$\begin{aligned} \dot{s} &= R\nu \\ \dot{\nu} &= -R^{-1}\Big(\alpha\text{sgn}(s) + \beta\text{sgn}(\nu) + ps + Rdv - \text{sgn}(s)\delta(t)\Big) \end{aligned} \tag{9}$$

Notice that the origin $s = \nu = 0$ of the system (9) corresponds to the origin $x_1 = x_2 = 0$ of the system (1-3). Note that the transformation is not invertible, one starts from the closed-loop system (9) and that the original dynamics can be recovered via (6). The solutions of system (9) are well defined in the sense of Filippov (see Filippov [1988]) . Furthermore, such formulation admits both friction and jump phenomena, while guaranteeing existence of solutions.

Previous results for the considered systems defined above are presented in the next section.

3. BACKGROUND

Some notions, fundamental for the rest of this work, are now recalled. The notation of some theorems was modified for readability. Let consider the closed-loop nonlinear system

$$\begin{aligned} \dot{x} &= y \\ \dot{y} &= -\alpha\text{sgn}(x) - \beta\text{sgn}(y) + \delta(x,y) \end{aligned} \tag{10}$$

where $\delta(x,y)$ is denoted as equation (4) and $M \in \mathbb{R}$ is a positive constant.

In Santiesteban et al. [2010] the following theorem was reported:

Theorem 1. Santiesteban et al. [2010] Let

$$\gamma_2 < \frac{4\sqrt{2}}{3}\gamma_1(\beta - M)\sqrt{\alpha} \tag{11}$$

and

$$\alpha - M > \beta > M \tag{12}$$

be satisfied then the function

$$\begin{aligned} V(x,y) = {} & \alpha^2\gamma_1 x^2 + \gamma_2|x|^{\frac{3}{2}}\text{sgn}(x)y \\ & + \alpha\gamma_1|x|y^2 + \frac{1}{4}\gamma_1 y^4; \;\; \gamma_1, \gamma_2 \in \mathbb{R}^+ \end{aligned} \tag{13}$$

is a strict Lyapunov function for system system (10). Then all trajectories of the perturbed system (10) converge to zero in finite time transient with a maximal duration of

$$t_{reach} < \frac{12\lambda_{max}^{\frac{3}{4}}(\Theta)}{\min(\kappa)} V^{\frac{1}{4}}(x(0), y(0)) \tag{14}$$

and $\kappa_\delta = \{\gamma_2(\alpha - \beta - M), \alpha\gamma_1(\beta - M), \frac{1}{2}\gamma_2, \frac{1}{2}\gamma_1(\beta - M)\}$.

Let consider the closed-loop nonlinear system

$$\begin{aligned} \dot{x} &= y \\ \dot{y} &= -\alpha \text{sgn}(x) - \beta \text{sgn}(y) - px - dy + \delta(x, y) \end{aligned} \tag{15}$$

Theorem 2. Santiesteban et al. [2010] Let

$$\begin{aligned} W(x, y) = V(x, y) &+ \frac{1}{2}p\gamma_1 x^2 y^2 + \frac{1}{4}p^2\gamma_1|x|^4 \\ &+ p\alpha\gamma_1|x|^3 + \frac{2}{5}d\gamma_2|x|^{\frac{5}{2}} \\ \gamma_1, \gamma_2, p, d &\in \mathbb{R}^+ \end{aligned} \tag{16}$$

a strict Lyapunov function for the perturbed system (15), where

$$\gamma_2 < \frac{4\sqrt{2}}{3}\gamma_1(\beta - M)\sqrt{\alpha} \tag{17}$$

holds for all $\gamma_i > 0$, $i = 1, 2$, and let the parameters α and β be such that the condition

$$\alpha - M > \beta > M \tag{18}$$

is satisfied. Then the system (15) is global finite time stable around the origin with

$$t_{reach} \leq \frac{2}{aW^{\frac{1}{8}}(x_0, y_0)} + \frac{2\sqrt{b}}{a^{\frac{3}{2}}} tan^{-1}\left(\frac{\sqrt{b}}{\sqrt{a}} W^{\frac{1}{8}}(x(0), y(0))\right)$$

and $a = \frac{\min(\kappa)}{3\alpha_{max}^{\frac{3}{4}}(\Theta)}$, $b = \frac{1}{2}\frac{\min(h,p)}{\gamma_1}$, $\kappa_\delta = \{\gamma_2(\alpha - \beta - M), \alpha\gamma_1(\beta - M), \frac{1}{2}\gamma_2, \frac{1}{2}\gamma_1(\beta - M)\}$ as an upper bound for time convergence for the trajectories to the equilibrium point.

In H. B. Oza and Spurgeon [2014] the following result was reported:

Theorem 3. H. B. Oza and Spurgeon [2014] The closed-loop impact system (1-3) and its transformed version (7 - 8) are globally finite time stable, regardless of whichever disturbance ω, satisfying condition (4) with $M < \beta < \alpha - M$, affects the system.

Now, in the following sections the main result of this paper will be developed.

4. TWISTING ALGORITHM

In this section a stability analysis of the system

$$\begin{aligned} \dot{s} &= R\nu \\ \dot{\nu} &= -R^{-1}\Big(\alpha \text{sgn}(s) + \beta \text{sgn}(\nu) - \text{sgn}(s)\delta(t)\Big) \end{aligned} \tag{19}$$

is under study. Indeed, a strict Lyapunov function is proposed in order to show finite time stability of the system (19).

Theorem 4. if

$$\begin{aligned} &\alpha - M > \beta > M \\ &\alpha > \left(\frac{\gamma_2}{2\gamma_1}R\right)^{\frac{2}{3}}; \ \alpha\Big(\beta - M\Big)^2 > \frac{9}{32}\left(\frac{\gamma_2}{\gamma_1}R\right)^2 \end{aligned} \tag{20}$$

holds, then system (19) has finite time convergence to the point $(x, y) = (0, 0)$ with

$$t_{reach} \leq \frac{4}{\zeta_{min}}\eta^{\frac{3}{4}} V^{\frac{1}{4}}(x(0), y(0)) \tag{21}$$

as an estimation of the convergence time, with

$$\begin{aligned} \zeta_{min} = \min\Bigg\{ &\frac{3}{2}\gamma_2 R, \gamma_2 R^{-1}\Big(\alpha - \beta - M\Big), \\ &2\gamma_1\alpha(\beta - M)R^{-1}, \gamma_1(\beta - M)R \Bigg\} \end{aligned} \tag{22}$$

and

$$\eta = \max\left\{\gamma_1\alpha^2 R^{-2}, \gamma_2, \gamma_1\alpha, \frac{1}{4}\gamma_1 R^2\right\} \tag{23}$$

.

In the next section, the stability analysis of the system (19) plus a linear PD controller will be treated.

5. TWISTING PLUS PD ALGORITHM

In this section a strict nonsmooth Lyapunov function is proposed to show finite time convergence of the trajectories of the system

$$\begin{aligned} \dot{s} &= R\nu \\ \dot{\nu} &= -R^{-1}\Big(\alpha \text{sgn}(s) + \beta \text{sgn}(\nu) + ps + Rdv - \text{sgn}(s)\delta(t)\Big) \end{aligned} \tag{24}$$

to the point $(x, y) = (0, 0)$ in spite of bounded external perturbations, *i.e.* $\delta(t) \neq 0$.

Theorem 5. if

$$\begin{aligned} &\alpha - M > \beta > M \\ &\alpha > \left(\frac{\gamma_2}{2\gamma_1}R\right)^{\frac{2}{3}}; \ \alpha(\beta - M)^2 > \frac{9}{32}\left(\frac{\gamma_2}{\gamma_1}R\right)^2 \end{aligned} \tag{25}$$

holds, then system (24) has finite time convergence to the point $(x, y) = (0, 0)$ with

$$t_{reach} \leq \frac{8}{5\zeta_{min-w}}\eta_w^{\frac{3}{8}} W^{\frac{5}{8}}(x(0), y(0)) \tag{26}$$

as an estimation of the convergence time, with

$$\begin{aligned} \zeta_{min-w} = \min\Bigg\{ &\frac{3}{2}\gamma_2 R, \gamma_2 pR^{-1}, \gamma_2 R^{-1}\Big(\alpha - \beta - M\Big), \\ &2\gamma_1\alpha(\beta - M)R^{-1}, \gamma_1(\beta - M)R, \gamma_1 dR^2, 2\gamma_1 d\alpha, \\ &\gamma_1 p(\beta - M)R^{-1}, \gamma_1 pd \Bigg\} \end{aligned} \tag{27}$$

and

$$\begin{aligned} \eta_w = \max\Bigg\{ &\gamma_1\alpha^2 R^{-2}, \gamma_2, \gamma_1\alpha, \frac{1}{4}\gamma_1 R^2; \ \frac{1}{2}p\gamma_1; \\ &\frac{1}{4}p^2\gamma_1 R^{-2}; \ p\alpha\gamma_1 R^{-2}; \ \frac{2}{5}d\gamma_2 R^{-1} \Bigg\} \end{aligned} \tag{28}$$

Example 1. It is possible to fix $\gamma_1 = R$ and $\gamma_2 = \epsilon$ as a small parameter, resulting in:

$$\alpha - M > \beta > M \qquad (29)$$
$$\alpha > \epsilon > 0;\ \ \alpha(\beta - M)^2 > \epsilon > 0$$

and

$$\zeta_{min-w} = \min\Big\{\frac{3}{2}\epsilon R;\ \ \epsilon R^{-1}\Big(\alpha - \beta - M\Big);$$
$$2\alpha(\beta - M);\ \ (\beta - M)R^2\Big\} \qquad (30)$$

and

$$\eta_w = \max\Big\{\frac{1}{2}\lambda_{\max}(P_p);\ \ R;\ \ \frac{1}{2}pR;\ \ \frac{1}{4}p^2R^{-1};$$
$$p\alpha R^{-1};\ \ \frac{2}{5}d\epsilon R^{-1}\Big\} \qquad (31)$$

This example has no other objective but to give an idea how this positive constants work.

In the next section, in order to support theoretical results a numerical experiment is under study using a one link pendulum as test bed. Moreover, a numerical exercise is used to show, for another example, how to fix the positive constants γ_1 and γ_2.

6. NUMERICAL EXPERIMENTS

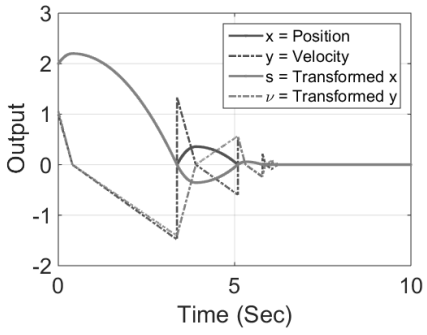
In this section a numerical simulation for the closed loop system (1-3) and its jump free system (24) it is shown. But first a numerical simulation for the system (1) with a twisting algorithm is performed. Using appropriate initial conditions $x(t_0) = 2$, $y(t_0) = 1$ and $s(t_0) = 2$, $v(t_0) = [1 - k]^{-1}$ and the parameter $\bar{e} = 0.9$. Figure 1 (a) shows the response of (1) with a twisting algorithm control law with $\alpha = 2$, $\beta = 1$ and the bounded disturbance M = 0.5, also it's shown the jump free system (19). To compute the reaching time for system (19), $\gamma_1 > 0$ and $\gamma_2 > 0$ needs to hold the conditions from (20), for this to be done, all possible values for γ_i, $i = 1, 2$, that holds for (20) are used to compute (21). Figure 1 (b), (c) shows the control input for systems (1) with twisting and (19) respectively.

$$t_{reach} \leq 58.05sec, \gamma_1 = 3.61, \gamma_2 = 3.24$$

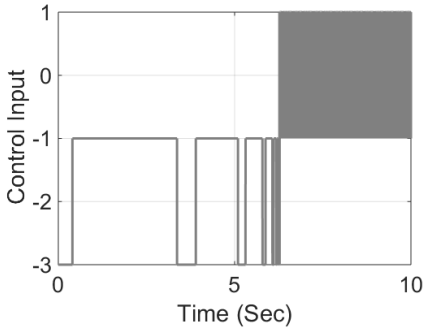
Now a numerical simulation for the system (1) using a twisting plus PD algorithm (3) and its jump free system (24) is displayed on figure 2 (a). The previous conditions are again been used in addition of the PD gains $p = 1$, $d = 0.5$. To compute the reaching time for system (24), $\gamma_1 > 0$ and $\gamma_2 > 0$ needs to hold the conditions from (25), for this to be done, all possible values for γ_i, $i = 1, 2$, that holds for (25) are used to compute (26). Figure 2 (b), (c) shows the control input for systems (1-3)and (24) respectively.

$$t_{reach} \leq 72.56sec, \gamma_1 = 1.67, \gamma_2 = 1.42$$

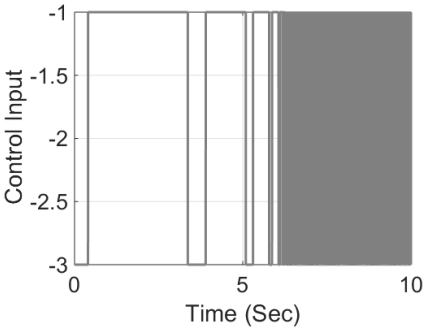
Furthermore, Figure 3 shows a performance comparison between the two controllers proposed in this paper. As we can see, the response given by (24) is faster and smoother due to the addition of a linear part given by the proportional and derivative controller.



(a)



(b) Control input for (1)



(c) Control input for (19)

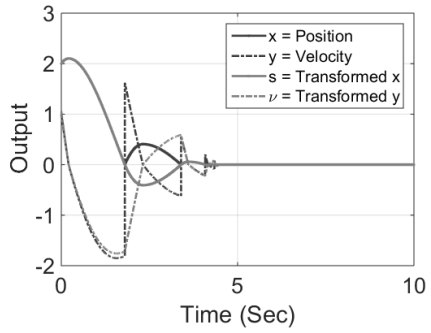
Figure 1. Systems (1) with a twisting control law and its jump-free system (19)

7. CONCLUSIONS

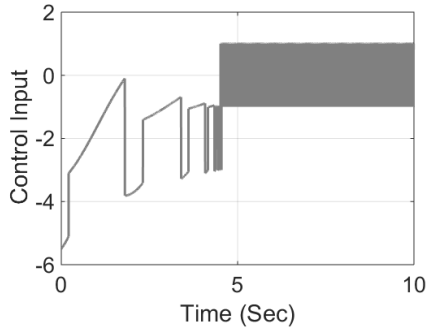
Finite time Stability and an estimation of convergence time of the double integrator with jumps in velocity and bounded external perturbations is shown, using Twisting and PD control laws. With this aim two non-smooth strict Lyapunov functions are proposed allowing an estimation of the upper bound of the convergence time. The performance of the algorithms were shown by a numerical simulation, in spite of bounded external perturbations. The closed loop system showed to be robust and provide nice performance in spite of unknown but bounded uncertainties. For future work, this result can be easily generalized for multidimensional case. Moreover, it can be extended when a state variable is not available for measurement, then a finite time observer can be applied.

REFERENCES

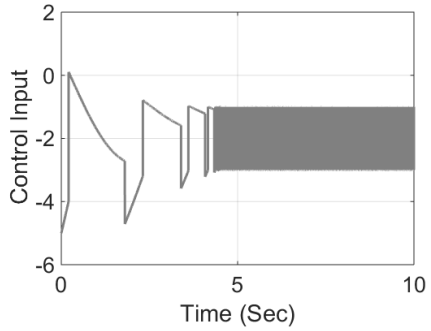
B. Brogliato, S.I.N. and Orhant, P. (1997). On the control of finitedimensional mechanical systems with unilateral



(a)



(b) Control input for (1-3)



(c) Control input for (24)

Figure 2. Systems (1-3) and (24)

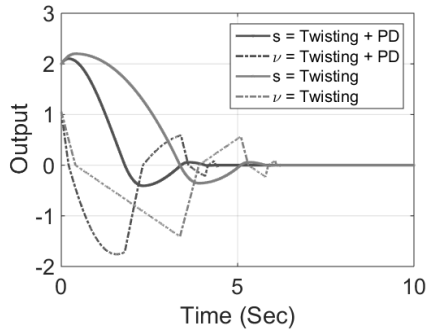


Figure 3. Comparison of the jump-free systems (19) and (24)

constraints. *IEEE Transactions on Automatic Control*, 42(2), 200–215.

Bacciotti, A. and Ceragioli (1999). Stability and stabilization of discontinuous systems and nonsmooth lyapunov functions. *ESAIM: Control, Optimisation and Calculus of Variations*, 4, 361–376.

Bacciotti, A. and Rosier, L. (2001). *Lyapunov functions and stability in control theory*. Springer, New York.

Bourgeot, J.M. and Brogliato, B. (2005). Tracking control of complementarity lagrangian systems. *International Journal of Bifurcation and Chaos*, 15(6), 18391866.

Brogliato, B. (1999). *Nonsmooth Mechanics*. Springer London, London.

Cortes, J. (2008). Discontinuous dynamical systems. *Control Systems Magazine, IEEE*, 28(03), 36 73.

Emelyanov, S., Korovin, S., and Levantovsky, L. (1986). Second order sliding modes in controlling uncertain systems. *Soviet Journal of Compute and System Science*, 24(4), 63–68.

Filippov, A.F. (1988). *Differential equations with discontinuous right-hand sides*. Kluwer Academic Publisher, Dordrecht.

Fridman, L. and Levant, A. (2002). *Higher order sliding modes. In Sliding mode control in engineering*. Control Engineering Series. London: CRC Press.

H. B. Oza, Y.V.O. and Spurgeon, S.K. (2014). Finite time stabilization of a perturbed double integrator with unilateral constraints. *Journal Mathematics and Computers in Simulation*, 95, 200–212.

J. Grizzle, G.A. and Plestan, F. (2001). Asymptotically stable walking for biped robots: analysis via systems with impulse effects. *IEEE Transactions on Automatic Control*, 46(1), 51 64.

Levant, A. (1993). Sliding order and sliding accuracy in sliding mode control. *International Journal of Contro*, 58, 1247–1263.

Moreno, J.A. and Osorio, M. (2008). A lyapunov approach to second-order sliding mode controllers and observers. *47th IEEE Conference on Decision and control 2008*, 2856–2861.

Orlov, Y. (2005). Finite-time stability and robust control synthesis of uncertain switched systems. *SIAM Journal on Control and Optimization*, 43, 1253–1271.

Polyakov, A. and Poznyak, A. (2009). Lyapunov function design for finite-time convergence analysis: Twisting controller for second-order sliding mode realization. *Automatica*, 45(3), 444–448.

Santiesteban, R., Fridman, L., and Moreno, J.A. (2010). Finite time convergence analysis for twisting controller via a strict lyapunov function. In *Proceedings of 11th Workshop on Variable Structure Systems*. Mexico City.

Shevitz, D. and Paden, B. (1994). Lyapunov stability theory of nonsmooth systems. *IEEE Transactions on Automatic Control*, 39, 1470–1476.

Shtessel, Y.B., Shkolnikov, I.A., and Levant, A. (2007). Smooth second-order sliding modes: Missile guidance application. *Automatica*, 43, 1470–1476.

Stewart, D.E. (2000). Rigid-body dynamics with friction and impact. *SIAM*, 42(1), 3–39.

Y. Hurmuzlu, F.G. and Brogliato, B. (2004). Modeling, stability and control of biped robotsa general framework. *Automatica*, 40, 1647 1664.

Delayed Observer Control for a Leader-Follower Formation with Time-gap Separation.

R. D. Cruz-Morales. A. Rodríguez-Ángeles. M. Velasco-Villa.

CINVESTAV-IPN
Departamento de Ingeniería Eléctrica , Sección de Mecatrónica
Av. I.P.N. No. 2508, Col. San Pedro Zacatenco 07360, México, D.F.
(e-mail: rdcruz,aangeles,velasco@cinvestav.mx).

Abstract: The main objective of this paper is to maintain a leader follower formation with a time gap separation between the mobile robots, with this control, the longitudinal distance variation between robots only depends on the leader velocity. The formation problem is solved using an observer that serves as a virtual reference to the follower robot that, correspond to the leader trajectory delayed τ units of time. The strategy is theoretically formally proven, and numerical simulations are presented.

Keywords: Time-gap separation, mobile robots, leader-follower formation.

1. INTRODUCTION

Mobile robots have been studied for many years, by using formation of mobile robots several problems have been solved and allowing dangerous, repetitive or security tasks, as is the case of patrolling different areas with obstacles Matveev et al. [2012]. For example, the cooperation among robots to complete tasks in which is required a formation that recreates a virtual structure to carry a heavy object, or just to maintain a geometric configuration between them Mehrjerdi et al. [2011] along a path. Some formations with mobile robots try to imitate the animal behavior with the objective of resembling a swarm or herd as in Sun and Wang [2007] where controlling and switching different formations between robots using synchronization to create the swarm movement. Another techniques use robots that move combining their sensors to improve the security by maximizing the possibilities of detecting dangerous situations, obstacles or covering areas on military search and rescue tasks or even security patrols Balch and Arkin [1998]. Sometimes, the formation can switch the leader in order to maintain the formation while the real leader avoids an obstacle and then the formation retakes the original leader or completes a specific goal with all the agents as a team Swaminathan et al. [2015]. In other cases mobile robots are used in rescue and recovery tasks in different environments as in Murphy et al. [2009], where several robots are use in mining accidents to manipulate fan doors, push aside obstacles, recolect gas, temperature reading and video.

The leader-follower formation has been studied for many years, is one of the most used formation, in some cases the formation is only studied for one follower as in Consolini et al. [2008], where only one robot is considered to prove the stability of the formation with the proposed control, In Tanner et al. [2004], Desai et al. [1998], the stability conditions are stablished for a set of followers using graph theory to prove the stability of a chained leader-follower formation, proving as well the conditions for safety and robustness. Leader-follower formation in most of the cases, is based on a longitudinal distance separation between robots Kawabe [2000], thats why the follower robot does not always track the same path that the leader robot describes, especially when the leader goes in a curved path. When the robots have to go on a predetermiated road, tracking the described path of the leader robot is very important. For that reason this paper focuses mainly in a time-gap separation between the robots that maintain the leader-follower formation, in contrast with most of the papers that the objective is to maintain a predetermined distance between the leader and the follower robot. This strategy is used in Adaptive Cruise Control (ACC) as in Bareket et al. [2003], where it has been shown that it is better to maintain a time-gap separation on the road that having a predeterminated distance. Taking into consideration the analysis made in Seppelt and Lee [2007] which shows the limitations of the ACC, it is possible to say that maintaining a security distance by taking a secure time-gap is better than only taking a secure longitudinal distance.

A time-gap separation strategy to perform the leader-follower formation is presented, the use of an observer that estimates the time delayed trajectory of the leader mobile robot serves as desired reference trajectory for the follower robot.

The document is presented as follows: first, the kinematic model of the mobile robot and the problem formulation is presented. The development of the observer and its convergence properties are presented in the next section. The work continues presenting the observer based strategy control together with its related convergence proof. Then, numerical simulations with two robots are done to show

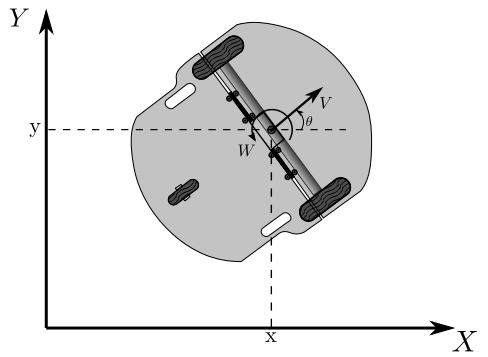


Fig. 1. Mobile robot type (2,0).

the performance of the proposed strategy. Finally, the paper closes with some conclusions.

2. KINEMATIC MODEL AND PROBLEM FORMULATION

It is considered a pair of differentially driven mobile robots (tipe(2,0)) as shown in Fig. 1. Their kinematic model are given by Canudas et al. [1996],

$$\begin{bmatrix} \dot{x}_i(t) \\ \dot{y}_i(t) \\ \dot{\theta}_i(t) \end{bmatrix} = \begin{bmatrix} \cos\theta_i & 0 \\ \sin\theta_i & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_i(t) \\ w_i(t) \end{bmatrix}, \quad i = L, F. \tag{1}$$

where (x_i, y_i) are the coordinates of the middle point of the robot axis in the plane $X - Y$, θ_i is the orientation of the vehicle with respect to X and v_i, w_i are the linear and angular input velocities respectively. It is assumed that the mobile robots are made up of rigid bodies, the wheels are non-deformable and they are moving on an horizontal plane, the contact between the wheels and the ground is in a single point of the plane and satisfy the pure rolling and non-slipping conditions along the motion, and the robot fulfills the non-holonomic constraint,

$$\dot{x}_i \sin(\theta_i) - \dot{y}_i \cos(\theta_i) = 0 \tag{2}$$

It will be set $i = L$ for the leader robot and $i = F$ for the follower one.

2.1 Problem formulation.

Consider a leader robot describing any feasible trajectory on the $X - Y$ plane. It is desired that a follower robot tracks the trajectory described by the leader robot delayed τ units of time, where τ represents the desired time gap separation between the leader and the follower robot.

The delayed leader trajectory is obtained by considering an input-delay observer based on the actual measurements of the leader robot. Under these conditions, it is intended that the follower robot tracks the delayed states provided by the observer. This time gap separation, based on a leader time delayed trajectory, is depicted in Fig. 2.

3. OBSERVER DESIGN

The development of the mentioned observer will be described taking into account the following assumptions.

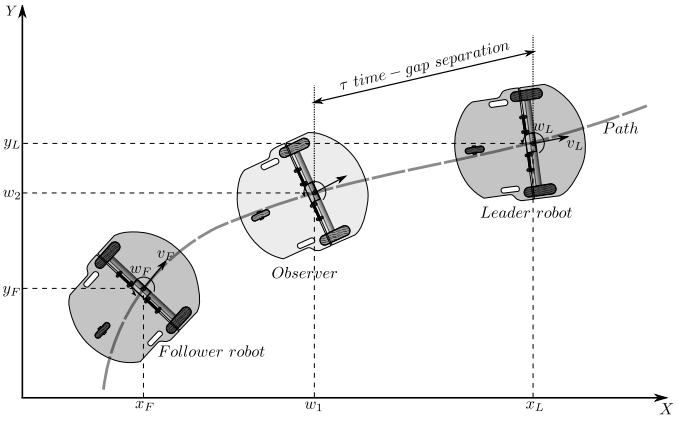


Fig. 2. Leader-follower formation with observer.

Assumption 1. The leader mobile robot variables $x_L(t)$, $y_L(t)$, $\theta_L(t)$, $v_L(t)$, $w_L(t)$ are available for measurement.

Assumption 2. The input leader robot signals $v_L(t)$, $w_L(t)$ are bounded for all t, this is

$$\sup\{v_L\} \leq \bar{v}_L, \ \sup\{w_L\} \leq \bar{w}_L \tag{3}$$

for some positive constants $\bar{v}_L$, $\bar{w}_L$.

To desing the observer for the delayed leader trajectory, consider the delayed variables,

$$w_1(t) = x_L(t-\tau), \ w_2(t) = y_L(t-\tau), \ w_3(t) = \theta_L(t-\tau) \tag{4}$$

Notice that the variables defined in (4) are available for design purposes based on Assumption 1. Hence, by taking the time derivative of (4), it is possible to obtain a virtual system that will serve as a reference for the follower robot, given by,

$$\begin{bmatrix} \dot{w}_1(t) \\ \dot{w}_2(t) \\ \dot{w}_3(t) \end{bmatrix} = \begin{bmatrix} v_L(t-\tau)\cos(w_3(t-\tau)) \\ v_L(t-\tau)\sin(w_3(t-\tau)) \\ w_L(t-\tau) \end{bmatrix}. \tag{5}$$

Based on the virtual system (5), a delayed observer can be proposed in the form,

$$\begin{bmatrix} \dot{\hat{w}}_1(t) \\ \dot{\hat{w}}_2(t) \\ \dot{\hat{w}}_3(t) \end{bmatrix} = \begin{bmatrix} v_L(t-\tau)\cos(\hat{w}_3(t)) + \lambda_1 e_{w_1}(t) \\ v_L(t-\tau)\sin(\hat{w}_3(t)) + \lambda_2 e_{w_2}(t) \\ w_L(t-\tau) + \lambda_3 e_{w_3}(t) \end{bmatrix} \tag{6}$$

where,

$$\begin{aligned} e_{w_1}(t) &= w_1(t) - \hat{w}_1(t) \\ e_{w_2}(t) &= w_2(t) - \hat{w}_2(t) \\ e_{w_3}(t) &= w_3(t) - \hat{w}_3(t) \end{aligned} \tag{7}$$

are the observation error variables, and λ_1, λ_2, λ_3 are constant positive design gains.

3.1 Observation error analysis

To analyze the observation error, a change of coordinates is done in the form,

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos(w_3) & \sin(w_3) & 0 \\ -\sin(w_3) & \cos(w_3) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} e_{w_1} \\ e_{w_2} \\ e_{w_3} \end{bmatrix} \tag{8}$$

this is,

$$
\begin{aligned}
e_1 &= e_{w_1}\cos(w_3) + e_{w_2}\sin(w_3)\\
e_2 &= -e_{w_1}\sin(w_3) + e_{w_2}\cos(w_3)\\
e_3 &= e_{w_3}.
\end{aligned} \tag{9}
$$

The observation error dynamics is obtained by taking the time derivative of (9). Taking the first line, and defining $\lambda_1 = \lambda_2 = \lambda$,

$$
\begin{aligned}
\dot{e}_1 &= \dot{e}_{w_1}\cos(w_3) - e_{w_1}\dot{w}_3\sin(w_3) + \dot{e}_{w_2}\sin(w_3)\\
&\quad + e_{w_2}\dot{w}_3\cos(w_3)\\
&= v_L(t-\tau)2\sin^2\left(\frac{e_3}{2}\right) - \lambda e_1 + e_2 w_L(t-\tau).
\end{aligned}
$$

Now with the second line of system (9),

$$
\begin{aligned}
\dot{e}_2 &= -\dot{e}_{w_1}\sin(w_3) - e_{w_1}\dot{w}_3\cos(w_3) + \dot{e}_{w_2}\cos(w_3)\\
&\quad - e_{w_2}\dot{w}_3\sin(w_3)\\
&= v_L(t-\tau)\sin(e_3) - \lambda e_2 - e_1 w_L(t-\tau)
\end{aligned}
$$

finally,

$$
\begin{aligned}
\dot{e}_3 &= w_L(t-\tau) - w_L(t-\tau) - \lambda_3 e_{w_3}\\
&= -\lambda_3 e_3.
\end{aligned}
$$

The observation error dynamics is,

$$
\begin{aligned}
\dot{e}_1 &= -\lambda e_1 + v_L(t-\tau)2\sin^2\left(\frac{e_3}{2}\right) + e_2 w_L(t-\tau)\\
\dot{e}_2 &= -\lambda e_2 + v_L(t-\tau)\sin(e_3) - e_1 w_L(t-\tau)\\
\dot{e}_3 &= -\lambda_3 e_3.
\end{aligned} \tag{10}
$$

The observation convergence properties can be formally stated in the next lemma.

Lemma 3. Suppose that Assumptions 1 and 2 are satisfied and that $\lambda_1, \lambda_2, \lambda_3 > 0$. Then, the states defined by the observer given in (6) exponentially converge to the leader robot trajectory delayed τ units of time.

Proof. Notice first that for $\lambda_3 > 0$,

$$
\dot{e}_3(t) = -\lambda_3 e_3(t) \tag{11}
$$

is exponentially stable, so, now the problem reduces to demonstrate that e_1 and e_2 also converge to the origin. With this aim, define,

$$
\bar{e}(t) = [\, e_1(t)\ e_2(t)\,]^T \tag{12}
$$

then, the dynamics of $e_1(t),\ e_2(t)$ can be expressed in the form,

$$
\dot{\bar{e}}(t) = \bar{A}\bar{e}(t) + \bar{F}(e_3, \hat{w})v_L(t-\tau) \tag{13}
$$

where,

$$
\bar{A} = \begin{bmatrix} -\lambda_1 & w_L(t-\tau) \\ -w_L(t-\tau) & -\lambda_2 \end{bmatrix} \tag{14}
$$

$$
\bar{F}(e_3, \hat{w}) = \begin{bmatrix} 2\sin^2\left(\frac{e_3}{2}\right) \\ \sin(e_3) \end{bmatrix}. \tag{15}
$$

To show the convergence to the origin of $\bar{e}(t)$, notice that,

$$
\left|\left|\bar{F}(e_3, \hat{w})v_L(t-\tau)\right|\right| \leq \left|\left|\bar{F}(e_3, \hat{w})\right|\right| |v_L| \leq \beta\,|e_3|\,\bar{v}_L \tag{16}
$$

with $\beta > 0$. Therefore, $\bar{F}(e_3, \hat{w})$ tends to zero as $e_3(t)$ converges to the origin, regardless of the evolution of $e_1(t)$ and $e_2(t)$. Therefore, $\bar{F}(e_3, \hat{w})v_L(t-\tau)$ is a vanishing signal for the system given by (13), this fact implies that the exponential convergence of the observation errors $e_1(t)$, $e_2(t)$ depend on the perturbation-free system (13) this is, on the system,

$$
\dot{\bar{e}}(t) = \bar{A}\bar{e}(t) \tag{17}
$$

A candidate Lyapunov function of the form,

$$
V = \frac{1}{2}e_1^2 + \frac{1}{2}e_2^2 \tag{18}
$$

produces

$$
\begin{aligned}
\dot{V} &= e_1\dot{e}_1 + e_2\dot{e}_2\\
&= e_1(-\lambda e_1 + e_2 w_L(t-\tau)) + e_2(-\lambda e_2 - e_1 w_L(t-\tau))\\
&= -\lambda e_1^2 - \lambda e_2^2\\
&= -2\lambda(\frac{1}{2}e_1^2 + \frac{1}{2}e_2^2)\\
&= -2\lambda V
\end{aligned} \tag{19}
$$

This concludes the proof.

In what follows, the estimated state $\hat{w}(t)$ will be used as a desired trajectory for the follower robot in order to solve the leader-follower formation problem. Notice that τ represents the time gap between the vehicles.

4. TIME GAP TRACKING CONTROL STRATEGY

Consider the follower robot dynamics,

$$
\begin{bmatrix} \dot{x}_F(t) \\ \dot{y}_F(t) \\ \dot{\theta}_F(t) \end{bmatrix} = \begin{bmatrix} \cos(\theta_F)v_F(t) \\ \sin(\theta_F)v_F(t) \\ w_F(t) \end{bmatrix} \tag{20}
$$

and take into account the reference trayectory of system (6). From system (20) it is possible to initially define,

$$
\begin{aligned}
\dot{x}_F &= v_F\cos(\theta_F) = \xi_1\\
\dot{y}_F &= v_F\sin(\theta_F) = \xi_2\\
\dot{\theta}_F &= w_F \qquad\qquad = \xi_3
\end{aligned} \tag{21}
$$

for

$$
\begin{aligned}
\xi_1 &= \dot{\hat{w}}_1 - k_1\tilde{x}\\
\xi_2 &= \dot{\hat{w}}_2 - k_2\tilde{y}\\
\xi_3 &= \dot{\hat{w}}_3 - k_3\tilde{\theta}
\end{aligned} \tag{22}
$$

with

$$
\begin{aligned}
\tilde{x} &= x_F - \hat{w}_1\\
\tilde{y} &= y_F - \hat{w}_2\\
\tilde{\theta} &= \theta_F - \hat{w}_3
\end{aligned} \tag{23}
$$

From (21) it is obtained,

$$
\cos^2(\theta_F)v_F + \sin^2(\theta_F)v_F = \xi_1\cos(\theta_F) + \xi_2\sin(\theta_F) \tag{24}
$$

so the control signals are,

$$
v_F = (\dot{\hat{w}}_1 - k_1\tilde{x})\cos(\theta_F) + (\dot{\hat{w}}_2 - k_2\tilde{y})\sin(\theta_F) \tag{25}
$$

$$
w_F = \dot{\hat{w}}_3 - k_3\tilde{\theta}. \tag{26}
$$

To analyze the tracking error, consider now the new error signals,

$$\begin{bmatrix} e_{s_1} \\ e_{s_2} \\ e_{s_3} \end{bmatrix} = \begin{bmatrix} \cos(\theta_F) & \sin(\theta_F) & 0 \\ -\sin(\theta_F) & \cos(\theta_F) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{\theta} \end{bmatrix}. \quad (27)$$

The dynamic of the tracking error can be obtained by taking the time derivative of system (27). Notice that,

$$\begin{aligned} \dot{e}_{s_1} &= \dot{\tilde{x}}\cos(\theta_F) - \tilde{x}\dot{\theta}_F \sin(\theta_F) + \dot{\tilde{y}}\sin(\theta_F) + \tilde{y}\dot{\theta}_F\cos(\theta_F) \\ &= v_F - v_L(t-\tau)\cos(e_{s_3}) + w_F(e_{s_2}) - \lambda_1 e_{w_1}\cos(\theta_F) \\ &\quad -\lambda_2 e_{w_2}\sin(\theta_F), \end{aligned}$$

The second line of (27) produces,

$$\begin{aligned} \dot{e}_{s_2} &= -\dot{\tilde{x}}\sin(\theta_F) - \tilde{x}\dot{\theta}_F\cos(\theta_F) + \dot{\tilde{y}}\cos(\theta_F) - \tilde{y}\dot{\theta}_F\sin(\theta_F) \\ &= v_L(t-\tau)\sin(e_{s_3}) - w_F e_{s_1} + \lambda_1 e_{w_1}\sin(\theta_F) \\ &\quad -\lambda_2 e_{w_2}\cos(\theta_F). \end{aligned}$$

Finally,

$$\begin{aligned} \dot{e}_{s_3} &= \dot{\tilde{\theta}} \\ &= w_F - \dot{\hat{w}}_3. \end{aligned}$$

The tracking error dynamic takes the form,

$$\begin{aligned} \dot{e}_{s_1} &= v_F - v_L(t-\tau)\cos(e_{s_3}) + e_{s_2}w_F - \lambda_1 e_{w_1}\cos(\theta_F) \\ &\quad -\lambda_2 e_{w_2}\sin(\theta_F) \\ \dot{e}_{s_2} &= v_L(t-\tau)\sin(e_{s_3}) - e_{s_1}w_F + \lambda_1 e_{w_1}\sin(\theta_F) \\ &\quad -\lambda_2 e_{w_2}\cos(\theta_F) \\ \dot{e}_{s_3} &= w_F - k_3\tilde{\theta}. \end{aligned} \quad (28)$$

Notice that the control signals, in the new coordinates (27) with $k_1 = k_2 = k$, take the form,

$$\begin{aligned} w_F &= \dot{\hat{w}}_3 - k_3 e_{s_3} \\ v_F &= -k e_{s_1} v_L(t-\tau)\cos(e_{s_3}) + \lambda_2 e_{w_2}\sin(\theta_F) \\ &\quad +\lambda_1 e_{w_1}\cos(\theta_F) \end{aligned} \quad (29)$$

4.1 Controlled system with observed states

System (28) in closed loop with the control signal (29), produces the system,

$$\begin{aligned} \dot{e}_{s_1} &= -k e_{s_1} + e_{s_2}\dot{\hat{w}}_3 - k_3 e_{s_2} e_{s_3} \\ \dot{e}_{s_2} &= -e_{s_1}\dot{\hat{w}}_3 + k_3 e_{s_1} e_{s_3} + m_1 \\ \dot{e}_{s_3} &= -k_3 e_{s_3} \end{aligned} \quad (30)$$

where,

$$m_1 = v_L(t-\tau)\sin(e_{s_3}) + \lambda(e_1\sin(e_{s_3}+e_3) - e_2\cos(e_{s_3}+e_3))$$

is a time varying term that do no explicitly depend on the tracking errors e_{s_1} and e_{s_2}. Notice from (30) that e_{s_3} is exponentially stable, and remember that e_1, e_2 and e_3 are exponentially stable, so the stability of (30) can be stated by considering the subsystem,

$$\begin{aligned} \dot{e}_{s_1} &= -k e_{s_1} + e_{s_2}\dot{\hat{w}}_3 - k_3 e_{s_2} e_{s_3} \\ \dot{e}_{s_2} &= -e_{s_1}\dot{\hat{w}}_3 + k_3 e_{s_1} e_{s_3} \end{aligned} \quad (31)$$

Remark 4. Note that the variable $m_1(t)$ depend on the observation errors $e_1(t)$, $e_2(t)$ and on the tracking error $e_{s_3}(t)$, that have previously been proven that exponentially converge to the origin. Therefore $m_1(t) \to 0$ as $t \to \infty$.

Since the functions m_1 is independent of $e_{s_1}(t)$, $e_{s_2}(t)$, it can be considered as an exogenous input signal. This fact allows to establish the stability of the closed loop system given by system (30) in terms of the free-perturbation system (31).

Lemma 5. Suppose that $k, k_3 > 0$ and that the angular input velocity of the follower robot satisfies for all t that $w_F(t) \neq 0$. Then, the closed loop tracking error dynamic given by Eqn. (30) is asymptotically stable. Consequently, the observer-based feedback (29) asimptotically solves the time-gap tracking problem associated with the leader and follower robots.

Proof. The statement of the lemma is equivalent to state the stability of system (31). With this aim consider the candidate Lyapunov function,

$$V = \frac{1}{2}e_{s_1}^2(t) + \frac{1}{2}e_{s_2}^2(t) \quad (32)$$

therefore,

$$\begin{aligned} \dot{V} &= e_{s_1}(t)\dot{e}_{s_1}(t) + e_{s_2}(t)\dot{e}_{s_2}(t) \\ &= -k e_{s_1}^2(t). \end{aligned} \quad (33)$$

Under this condition, it is clear that,

$$\dot{V} \leq 0.$$

Therefore, the states of the system given by Eqn. (31) are stable. To show that the states converge to the origin, notice that $\dot{V}$ is uniformily continuous since $\ddot{V}$ is bounded. Invoquing Barbalat's lemma Slotine et al. [1991], $\dot{V} \to 0$ and consequently, from the first line of (31) it is obtained

$$0 = e_{s_2}\dot{\hat{w}}_3 - k_3 e_{s_2} e_{s_3} \quad (34)$$

this is,

$$(\dot{\hat{w}}_3 - k_3 e_{s_3}) e_{s_2} = 0 \quad (35)$$

Considering the assumption that $w_F \neq 0$, thus $\dot{\hat{w}}_3 \neq 0$, and taking into account that $e_{s_3} \to 0$, it is clear that $e_{s_2} \to 0$ as $t \to \infty$.

5. NUMERICAL SIMULATIONS

To show the effectiveness of the observer-based solution proposed in this work a three petals geometric trajectory is proposed for the leader robot. This path involves orientation and velocity changes that influence the relative distance between the robots. The equations that describe the desired leader trajectory are given as,

$$\begin{aligned} x &= 0.2(a+b)\cos(npt)\cos(pt) \\ y &= 0.2(a+b)\cos(npt)\sin(pt) \end{aligned} \quad (36)$$

where $n = 3$ corresponds to the number of petals, $a = 13$, $b = 7$, $p = (2\pi/20)$. Equation (36) allows to define the input signals for the leader robot as,

$$v_L = \sqrt{\dot{x}^2 + \dot{y}^2}, \quad w_L = \frac{(\ddot{y}\dot{x} - \ddot{x}\dot{y})}{(\dot{x}^2 + \dot{y}^2)} \quad (37)$$

Table 1.

Robot	x[m]	y[m]	θ[rad]
Leader	4	0	$\pi/2$
Follower	4	0	$\pi/2$
Observer	3	-0.5	0

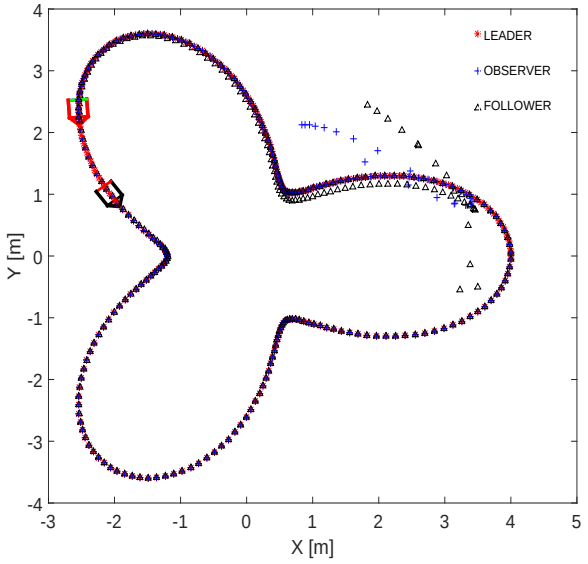


Fig. 3. Leader follower formation

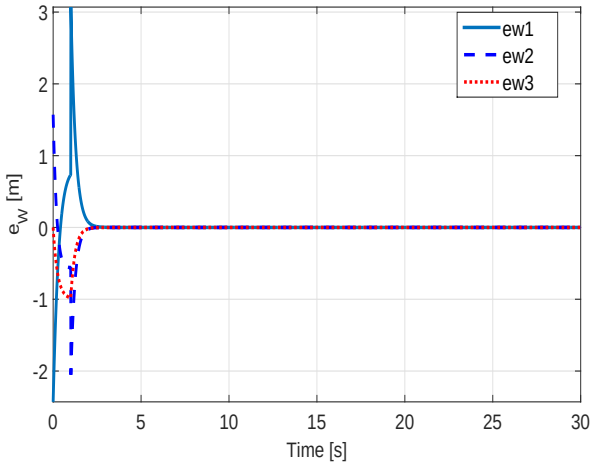


Fig. 4. Observation errors

The initial conditions for the simulation are shown in Table 1. The gain used for the observer is $\lambda = 4$, and the gains used for the control law are set as $k_1 = k_2 = 2,\ k_3 = 4$. The time-gap separation was considered as $\tau = 1sec.$ The time evolution on the plane of the leader, the follower and the estimated trajectories are shown in Fig. 3.

The observation errors $e_{w_1}(t),\ e_{w_2}(t)$ and $e_{w_3}(t)$ are depicted in Fig. 4 where it is shown their fast convergence.

The tracking position errors $e_{s_1}(t),\ e_{s_2}(t)$ and $e_{s_3}(t)$ are shown in Fig. 5 and the control signal evolution for both, the leader and the follower robots are shown in Fig. 6.

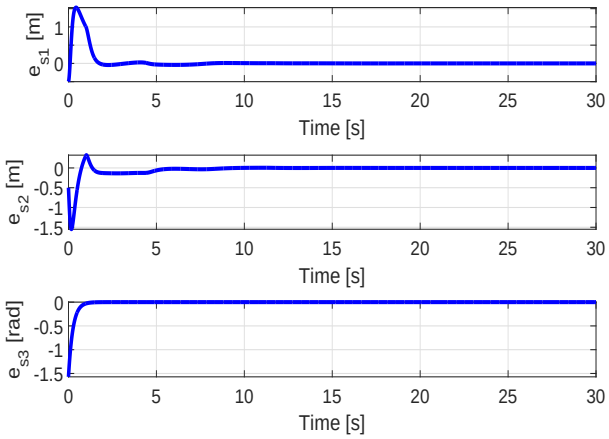


Fig. 5. Position tracking errors

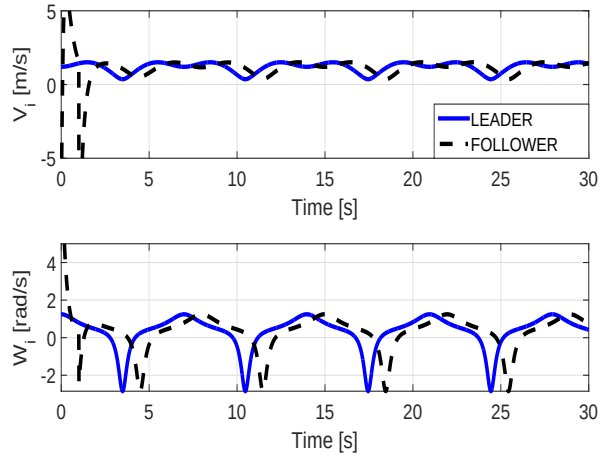


Fig. 6. Control signal evolution

Notice that despite the initial conditions errors, all the signals converge adequately as expected by the theoretical developments.

Figure 7 shows the evolution of the position in x, y of the observer, leader, and follower robot.

It is evident that the time gap separation of $\tau = 1sec.$ is kept along the simulation after the transient has passed. Meanwhile in Fig. 8, it is shown how the separation or relative distance between the robots $d(t) = \sqrt{(x_L - x_F)^2 + (y_L - y_F)^2)}$ varies accordingly to the velocity of the robots in order to satisfy the time gap separation.

6. CONCLUSIONS

Based on a delayed observer strategy, the leader-follower fomation is solved for a pair of differentially driven nonholonomic mobile robots. It is shown that under the assumption of a leader angular velocity different from zero, the observation and tracking errors exponentially converge to the origin while a desired constant time gap is maintained constant between the vehicles. Numerical simulations show the efectiveness of the proposed leader-follower

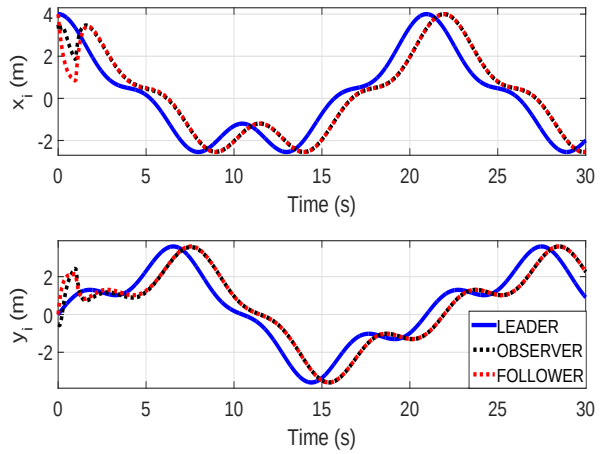


Fig. 7. x, y position evolution

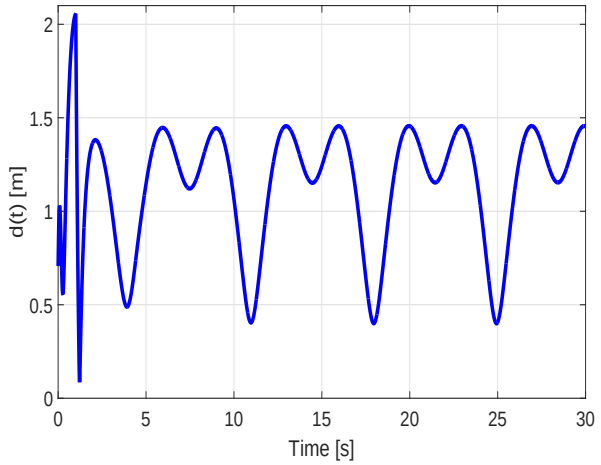


Fig. 8. Relative distance between leader and follower robots

solution. The simulations shown that the proposed control law can solve the problem of leader follower formation tracking the same path that the leader robot performed.

REFERENCES

Tucker Balch and Ronald C Arkin. Behavior-based formation control for multirobot teams. *Robotics and Automation, IEEE Transactions on*, 14(6):926–939, 1998.

Zevi Bareket, Paul S Fancher, Huei Peng, Kangwon Lee, and Charbel A Assaf. Methodology for assessing adaptive cruise control behavior. *Intelligent Transportation Systems, IEEE Transactions on*, 4(3):123–131, 2003.

DE Canudas, Bruno Siciliano, and Georges Bastin. *Theory of robot control*. GBR, 1996.

Luca Consolini, Fabio Morbidi, Domenico Prattichizzo, and Mario Tosques. Leader–follower formation control of nonholonomic mobile robots with input constraints. *Automatica*, 44(5):1343–1349, 2008.

Jaydev P Desai, Jim Ostrowski, and Vijay Kumar. Controlling formations of multiple mobile robots. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 4, pages 2864–2869. IEEE, 1998.

Taketoshi Kawabe. Vehicle spacing control with string stability using simple communication and a spacing sensor. In *Proceedings of the 7th World Congress on Intelligent Systems*, 2000.

Alexey S Matveev, Chao Wang, and Andrey V Savkin. Real-time navigation of mobile robots in problems of border patrolling and avoiding collisions with moving and deforming obstacles. *Robotics and Autonomous systems*, 60(6):769–788, 2012.

Hasan Mehrjerdi, Jawhar Ghommam, and Maarouf Saad. Nonlinear coordination control for a group of mobile robots using a virtual structure. *Mechatronics*, 21(7): 1147–1155, 2011.

Robin R Murphy, Jeffery Kravitz, Samuel L Stover, and Rahmat Shoureshi. Mobile robots in mine rescue and recovery. *Robotics & Automation Magazine, IEEE*, 16 (2):91–103, 2009.

Bobbie D Seppelt and John D Lee. Making adaptive cruise control (acc) limits visible. *International Journal of Human-Computer Studies*, 65(3):192–205, 2007.

Jean-Jacques E Slotine, Weiping Li, et al. *Applied nonlinear control*, volume 199. Prentice-Hall Englewood Cliffs, NJ, 1991.

Dong Sun and Can Wang. Controlling swarms of mobile robots for switching between formations using synchronization concept. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 2300–2305. IEEE, 2007.

Siddharth Swaminathan, Mike Phillips, and Maxim Likhachev. Planning for multi-agent teams with leader switching. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 5403–5410. IEEE, 2015.

Herbert G Tanner, George J Pappas, and Vijay Kumar. Leader-to-formation stability. *Robotics and Automation, IEEE Transactions on*, 20(3):443–455, 2004.

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

Design of preserving order observers-based controllers for discrete-time linear systems

Jesús D. Avilés [*,⋆] **Jaime A. Moreno**[‡]
Fernanda Alvarez-Mendoza[‡]

[*] *Instituto Politécnico Nacional, Sección de Estudios de Investigación y Posgrado, ESIME-UPT, C.P. 07430, Ciudad de México, México (e-mail: jesus.david.aviles@gmail.com).*
[‡] *Instituto de Ingeniería-UNAM, Coyoacán DF, 04510, Ciudad de México, México, (e-mail: {JmorenoP,MalvarezM}@iingen.unam.mx)*

Abstract:
In this work we provide a design method of preserving order observer-based dynamic output-feedback controllers for a family of discrete-time linear systems. This design considers the absence and/or presence of disturbances/uncertainties in the discrete-time systems. The control problem is focused in two purposes: (i) the stabilization of closed-loop systems by means of upper and/or lower estimations, and (ii) the state estimation preserves the partial-order with respect to the real state trajectory at each instant time. The proposed method studies the properties of Lyapunov stability and cooperativeness. The first property determines the exponential stability of the closed loop system (with controller) in combination with exponential convergence to zero of the observation errors, when there are no uncertain terms, while the second one ensures the partial ordering between upper/lower estimations with state. The observers gain can be computed through a solution of a Bilinear Matrix Inequality (BMI) and a Linear Matrix Inequality (LMI).

Keywords: Discrete-Time Systems, Preserving Order Observers, Interval Observers, Cooperative Systems.

1. INTRODUCTION

An especial class of robust state estimators, the so-called *preserving partial-order observers* have been successfully used, for instance, in biological, electric, hydraulic systems, positive and compartmental systems and many other applications, in order to estimate uncertain/disturbed during the last fifteen years (Gouze et al., 2000; Alcaraz-Gonzalez et al., 2002; Bernard and Gouze, 2004; Avilés and Moreno, 2009, 2013; Mazenc et al., 2013; Avilés and Moreno, 2014; Mazenc et al., 2014; Raïssi et al., 2012). This observers, based in the cooperativeness theory (Angeli and Sontag, 2003; Hirsch and Smith, 2004, 2005), provided upper and/or lower estimations, which can form an interval containing the real state trajectory, instead of estimating true values of the uncertain non-measureable variables. The precursor design appeared in (Gouze et al., 2000), where the *interval observers* were proposed, which are constituted by a preserving upper and lower partial-order observer; represent a interesting technique to estimate parameters and unknown variables of biotechnological processes.

Several works have been proposed for discrete-time systems (Mazenc et al., 2013; Efimov et al., 2013a; Avilés and Moreno, 2015b); they are inspired by the growing computational interest to use sample-data systems subjected to perturbations, where show that the cooperativeness property is conserved in the interval observers for the discrete-time systems (Hirsch and Smith, 2005; Mazenc et al., 2014). Noting that the cooperativeness linear condition is expressed by means of nonnegative matrices for time-discrete systems, instead of Metzler matrices for continuous-time systems.

Recently, in (Efimov et al., 2013b) the interval observers class were implemented in the design of controllers. The output stabilization was analyzed for a class of Linear Parametric Variant (LPV) continuous-time system, implementing the estimations from the interval observers. Subsequently, in (Efimov et al., 2015) it was introduced a similar analysis for the family of LPV discrete-time systems. Additionally, in (Avilés and Moreno, 2015a) have been proposed controllers based on interval observers for stabilizing a continuous-time closed-loop scheme in absence/presence of disturbances.

In the present work, we develop a design methodology of controllers for discrete-time linear systems, using the output feedback provided by the upper and/or lower preserving partial-order observers (Avilés and Moreno, 2009, 2013, 2014). The absence and presence of uncertainties/disturbances are studied in order to guarantee both the stability of the closed-loop system and the state trajectory is bounded for the estimations of the observers. This is achieved by combination of systemic properties between

⋆ J.D. Avilés is on leave from Autonomous University of Baja California. Faculty of Engineering and Business. C.P. 21460, Tecate, B.C, México.

the stability in the sense of Lyapunov and cooperativeness. Moreover, the computational implementation is considered by means of a BMI and a LMI.

This paper is organized as follows. The notations and prelimiaries are presented in Section 2. The design method of controllers based on preserving partial-order (interval) observers, for nominal and perturbed discrete-time systems, is proposed in Section 3. Several practical aspects are mentioned in Section 4. Concluding remarks are established in Section 5.

2. PRELIMINARIES

For a better understanding, we provide the following descriptions.

2.1 Notations

$M \overset{M}{\succeq} 0$	: M is a Metzler matrix, if and only if $M_{ij} \geq 0,\ \forall i \neq j,\ \ \forall i \in 1, ..., n,\ \forall j \in 1, ..., n.$
$A \succeq 0$	: A is a nonnegative matrix, sii $A_{ij} \geq 0$, $\forall i, j = 1, ..., n.$
$A \geq 0$	: A is a positive semi-definite matrix.
$A > 0$	: A is a positive definite matrix.
$B \leq 0$	: B represents a negative semi-definite matrix.
$B < 0$	: B represents a negative semi-definite matrix.
$x \succeq 0$	: x is a nonnegative vector, iff $x_i \geq 0, \forall i = 1, ..., n.$
$x \succeq z$	: represents the partial-order between two vectors, iff $x_i - z_i \geq 0, \forall i = 1, ..., n.$
$\|x\|$	: is the Euclidean norm.
$\mathbf{J}_{f(x)}$	: denotes the Jacobian matrix of the nonlinear function$f(t, x, u)$ with respect to the variable x
$\mathbf{J}_{f(u)}$	: is the Jacobian matrix of the nonlinear function $f(t, x, u)$ w.r.t the variable u.

2.2 Systems that preserver partial-order

The *cooperativeness* is a systemic property that ensures the partial order between the state trajectories and output, depending on the partial order between the initial states and inputs.

Definition 1. Consider the time-discrete nonlinear system

$$\Gamma_{\mathrm{NL}} : \begin{cases} x\left(k+1\right) = f\left(k, x(k), u(k)\right), & x(k_0) = x_{k0}, \\ y\left(k\right) = h\left(k, x(k), u(k)\right), \end{cases} \quad (1)$$

where $(x(k), u(k), y(k)) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p$ is the state, the input and the output vectors of Γ_{NL}, respectively, at each instant time $k \in \mathbb{N}$. We assume that $f : \mathbb{N} \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $h : \mathbb{N} \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^p$ are continuous functions.

The system Γ_{NL} is *cooperative* if there exists an partial ordering between two initial states and two inputs (if they exist), that is; $x_0^1 \succeq x_0^2,\ \ u^1\left(k\right) \succeq u^2\left(k\right),\ \ \forall k$. Then, the state and output trajectories will preserve the partial-order for all instant times:

$$x\left(k, k_0, x_0^1, u^1\left(k\right)\right) \succeq x\left(k, k_0, x_0^2, u^2\left(k\right)\right), \forall k \geq k_0,$$
$$y\left(k, k_0, x_0^1, u^1\left(k\right)\right) \succeq y\left(k, k_0, x_0^2, u^2\left(k\right)\right), \forall k \geq k_0,$$

where $k_0 \in \mathbb{N}$, $x_{k_0} \in \mathbb{R}^n$. □

In the following paragraphs, we present the characterization of discrete-time systems. For more details to see (Mazenc et al., 2014; Hirsch and Smith, 2005).

Proposition 2. (Hirsch and Smith (2005)). Γ_{NL} is a *discrete-time cooperative system* if and only if the following conditions are satisfied,

$$(\mathbf{a}).\ \mathbf{J}_{f(x)} \succeq 0, \quad (\mathbf{b}).\ \mathbf{J}_{f(u)} \succeq 0, \quad (\mathbf{c}).\ \mathbf{J}_{h(x)} \succeq 0. \qquad \square$$

The linear cooperative systems represent a particular case of the *definition 3*; its characterization is given in the next paragraph.

Proposition 3. The Linear Time-Invariant (LTI) system described by

$$\Gamma_{\mathrm{L}} : \begin{cases} x(k+1) = Ax(k) + Bu(k), & x\left(k_0\right) = x_{k0}, \\ y(k) = Cx(k), \end{cases} \quad (2)$$

where $(x(k), u(k), y(k)) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p$ are the state, the input and the measurement output vectors, respectively. Γ_{L} is a *cooperative system* if and only if

$$(\mathbf{i}).\ A \succeq 0, \quad (\mathbf{ii}).\ B \succeq 0, \quad (\mathbf{iii}).\ C \succeq 0$$

For the family of discrete-time linear systems, the cooperativeness is analog to positivity property, which states that all variables: input, output and state, are nonnegative at all instant times.

2.3 Stability of discrete-time linear systems

In this subsection, the stability of the linear discrete-time systems will be analyzed.

Definition 4. (Khalil (2002)). We consider the system of the form

$$x(k+1) = Ax(k), \quad x(k_0) = x_{k0}, \quad (3)$$

where $A \in \mathbb{R}^{n \times n}$ and $k \in \mathbb{N}$, if there exist the constants $\alpha > 0$ and $\rho > 0$ such that

$$\|x(k, x_{k_0})\| \leq \alpha \|x_{k0}\| exp(-\rho k) \quad (4)$$

is satisfied for all $k \in \mathbb{N}$, $x_{k0} \in \mathbb{R}^n$, then the origin is a *globally exponentially stable equilibrium point.*

Definition 5. (Jiang and Wang (2001)). The discrete-time linear system affected by the presence of additive exogenous inputs, is given by

$$x(k+1) = Ax(k) + b(k), \quad x(k_0) = x_{k0} \quad (5)$$

is (globally) Input-State Stable (ISS) w.r.t $b(k)$, if there exist a function β of class $\mathcal{KL}$, and a function γ of class $\mathcal{K}$ such that, for each initial state $x(k_0) \in \mathbb{R}^n$ and each input $b(k)$, then the solution $x(t)$ exist for all $t \geq t_0$, it holds that

$$\|x(k, x_{k_0})\| \leq \beta(\|x_{k0}\|, k) + \gamma(\|b(k))\| \quad (6)$$

where γ is known as an *ISS-gain* for Γ_{L}.

We consider the Lyapunov function candidate

$$V(x(k)) = x^T(k) P x(k), \quad P = P^T > 0. \quad (7)$$

which satisfies

$$\alpha_1(||x(k)||) \leq V(x(k)) \leq \alpha_2(||x(k)||), \quad (8)$$
$$\Delta V = V(x(k+1)) - V(x(k)) \leq -\alpha_3(||x(k)||), \quad (9)$$

$\forall\ x(k) \in \mathbb{R}^n$, for the family of Linear time-discrete nominal systems in (3). The expression ΔV is given as the Lyapunov difference. For the disturbed system in (5), an additional condition is required:

$$\Delta V = V(x(k+1)) - V(x(k)) \leq -\alpha_3(||x(k)||) + \gamma(||b(k)||), \tag{10}$$

$\forall b(k) \in \mathbb{R}^n$.

The characterizations of both definitions *5* and *6*, can be expressed in the ambient of Linear Matrix Inequalities (LMI's).

Lema 6. Consider the nominal system in (3). If there exists a positive symmetric matrix $P = P^T > 0$ and the positive constant $\epsilon > 0$, such that

$$[A^T P A - P + \epsilon P] \leq 0 \tag{11}$$

is satisfied, then $x(k) = 0$ is an *globally exponentially stable* equilibrium point. □

Now, we present the characterization of practical stability for the system Γ_{L}, when an exogenous input $b(k)$ appears in the dynamics. In this sense, we analyze the property by means of the input-to-state stability (ISS) condition for Γ_{L} in the following Lemma.

Lema 7. Consider the perturbed system in (5). If conditions from Lemma 6 are satisfied, then the system in (5) is *ISS with respect to the input* $b(k)$.

Proof. Initially the demonstration of Lemma 6 is presented. If the Lyapunov function candidate $V(x(k)) = x^T(k)Px(k)$ is defined for the system in (3), its Lyapunov difference along the trajectory of (3) is given as

$$\Delta V = V(x(k+1)) - V(x(k)) = x^T(A^T P A - P)x \leq -\epsilon x^T P x$$

Now, the same Lyapunov function candidate is proposed for the perturbed system, then its derivative is bounded by

$$\begin{aligned}\Delta V \leq &-(1-\theta)\,\epsilon V(x(k)) - \theta\epsilon V(x(k)) + b^T(k)Pb(k) \\ &+ 2x^T(k)A^T Pb(k)\end{aligned}$$

Completing squares, the property (6) holds. ■

3. PRESERVING ORDER OBSERVER-BASED CONTROL DESIGN

In this section, we present the main results of the design of preserving order observer-based dynamic output-feedback controllers for discrete-time systems. Firstly, the controller design conditions are studied for the nominal case, then we extend the notions to deal the presence of disturbances/uncertainties in the dynamical system.

3.1 Nominal System

Consider the family of nominal systems given by

$$\Pi_{\mathrm{N}} : \begin{cases} x(k+1) = Ax(k) + B_u u(k),\ x(k_0) = x_{k0}, \\ y(k) = Cx(k), \end{cases} \tag{12}$$

where $x(k) \in \mathbb{R}^n$ is the state, $y(k) \in \mathbb{R}^p$ is the measurement output, $u \in \mathbb{R}^m$ is the control input. $A \in \mathbb{R}^{n\times n}$, $B_u \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, are constant matrices such that are satisfied the next suppositions:

S1. The pair (A, B_u) is stabilizable, and
S2. The pair (A, C) is detectable.

3.1.1 Preserving partial-order observers

We propose an Luenberger state observer for the family of discrete-time systems Π_{N} in order to estimate the unknown state variables,

$$\Pi_{\mathrm{O}} : \begin{cases} \widehat{x}(k+1) = A\widehat{x}(k) + L(\widehat{y}(k) - y(k)) + B_u u(k), \\ \widehat{y}(k) = C\widehat{x}(k), \quad x(k_0) = x_{k0} \end{cases} \tag{13}$$

where $\widehat{x}(k)$ is the estimated state. $L \in \mathbb{R}^{n\times q}$ is the observer gain associated to the output injection.

The estimation error has been defined as $e(k) \triangleq \widehat{x}(k) - x(k)$, and the output error is $\widetilde{y}(k) \triangleq \widehat{y}(k) - y(k)$. Consistently, the dynamics of estimation error are given by

$$\Pi_{\mathrm{E}} : \begin{cases} e(k+1) = A_L e(k), \qquad e(k_0) = e_{k0}, \\ \widetilde{y}(k) = Ce(k), \end{cases} \tag{14}$$

where the matrix $A_L \triangleq A + LC$. From Π_{O}, the sufficient conditions are established to define the preserving partial-order observer for the nominal systems Π_{N}.

In the following paragraph we establish the cooperativeness conditions for the state observer Π_{O}.

Definition 8. (Avilés and Moreno (2014)). The observer Π_O is said an upper/lower preserving partial-order observer if

(i). The estimation error system Π_E is cooperative, that is, given $e_0 \succeq 0$ $(-e_0 \succeq 0)$, then $e(t) \succeq 0$ $(-e(t) \succeq 0)$, leading to, if $\widehat{x}_0 \succeq x_0$ $(x_0 \succeq \widehat{x}_0) \rightarrow \widehat{x}(t) \succeq x(t)$ $(x(t) \succeq \widehat{x}(t))$, $\forall t \geq 0$.

(ii). The estimation error converge asymptotically to zero, that is $\|\widehat{x}(k) - x(k)\| \to 0$ as $k \to \infty$. □

The first condition (i) is ensured, if

$$A_L \succeq 0 \tag{15}$$

is a nonnegative matrix, which is provide by the application of the cooperative property in the estimation error system (see *Proposition 2*). Thus, the condition guarantees the partial ordering between estimation and state trajectories for all instate times.

An upper and a lower preserving partial-order observer form an *interval observer*, whose characteristic behavior guarantees that the estimations always stay above and below the true state trajectory.

3.1.2 Control design method

Adding an output feedback control to the system Π_{O}, we provide a control based on a lower/upper preserving partial-order observer,

$$\Pi_{\mathrm{U}} : \begin{cases} \widehat{x}(k+1) = A\widehat{x}(k) + L(\widehat{y}(k) - y(k)) + B_u u(k), \\ \widehat{y}(k) = C\widehat{x}(k), \quad x(k_0) = x_{k0}, \\ u(k) = -K\widehat{x}(k), \end{cases} \tag{16}$$

where $K \in \mathbb{R}^{n\times m}$ is the controller gain. Consequently, the closed-loop system Π_{U} is combined with the estimation error system, which can be described by

$$\Pi_{\mathrm{LC}}: \left\{ \begin{bmatrix} x(k+1) \\ e(k+1) \end{bmatrix} = \begin{bmatrix} A_K & -B_uK \\ 0 & A_L \end{bmatrix} \begin{bmatrix} x(k) \\ e(k) \end{bmatrix} \right. \quad (17)$$

where the matrix $A_K = A - B_uK$. In the Theorem 9 is shown the stability condition of the augmented system (17).

Theorem 9. Consider the nominal system in (17). Suppose that the conditions (S1) and (S2) are satisfied. The system Π_{LC} is globally exponentially stable in the sense of Lyapunov, if there exist the matrices K, L, $P_1 = P_1^T > 0$, $P_2 = P_2^T > 0$ and the constants $\epsilon_1 > 0$, $\epsilon_2 > 0$, such that

$$\begin{bmatrix} A_K^T P_1 A_K - P_1 + \epsilon_1 P_1 & -A_K^T P_1 B_u K \\ -K^T B_u^T P_1 A_K & \Upsilon \end{bmatrix} \leq 0 \quad (18)$$

is fulfilled, where $\Upsilon = K^T B_u P_1 B_u K + A_L^T P_2 A_L - P_2 + \epsilon_2 P_2$.

Proof.

For the augmented system in (17), a Lyapunov function candidate is proposed as

$$V\left(x(k), e(k)\right) = x^T(k) P_1 x(k) + e^T(k) P_2 e(k), \quad (19)$$

and its Lyapunov difference along the trajectories of the system Π_{LC}, $\Delta V\left(x, e\right)$, is given by

$$\begin{bmatrix} x \\ e \end{bmatrix}^T \begin{bmatrix} A_K^T P_1 A_K - P_1 & -A_K^T P_1 B_u K \\ -K^T B_u^T P_1 A_K & \Upsilon_a \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix}$$
$$\leq -\epsilon_1 x^T P_1 x - \epsilon_2 e^T P_2 e$$

where $\Upsilon_a = K^T B_u P_1 B_u K + A_L^T P_2 A_L - P_2$. Then, the origin, $x(k) = 0$, $e(k) = 0$, is globally exponentially equilibrium point of the augmented system Π_{LC} in (17). ■

A main result of this work is presented in the following Theorem, which considers the design of a preserving order observer-based control.

Theorem 10. Consider the nominal system Π_{N}. Suppose that conditions (S1) and (S2) are satisfied. If conditions

- Cooperativeness in (15)
- Stability in (18)

are satisfied, then the augmented closed-loop system Π_{LC} is globally exponentially stable. Moreover, Π_{U} is a preserving order observer.

Remark 11. Note that the design method presented in Theorem *10*, establishes the stabilization of Π_{N} by means of output feedback of Π_{U}. Moreover, the Theorem *10* ensures that an upper or a lower estimation bound to the real state trajectory for all times. It is possible to eject simultaneously a pair of preserving partial-order observers depending on an upper and an lower initial state, implying that the estimations form an interval, which contains the state trajectory. This is the typical notion of the interval observers.

3.2 Disturbed systems

Consider the discrete-time linear system

$$\Phi_{\mathrm{P}}: \left\{ \begin{array}{l} x(k+1) = Ax(k) + B_u u(k) + \pi(k, x(k)), \\ y(k) = Cx(k), \quad x\left(k_0\right) = x_{k0}, \end{array} \right. \quad (20)$$

where $\pi(t, x)$ represents the disturbances and/or parametric uncertainty affecting to Π_{P}. This term satisfies the inequality

$$\pi^+(k, y(k)) \succeq \pi(k, x(k)) \succeq \pi^-(k, y(k)) \quad (21)$$

which implies that the disturbance $\pi(t, x)$ is bounded by intervals, where $\pi^+(t, y)$ and $\pi^-(t, y)$ are known Lipschitz functions.

3.2.2 Preserving partial-order observers for disturbed systems

In order to estimate the unknown state of Π_{P}, two Luenberger observers are proposed, given by

$$\Phi_{O^+}: \left\{ \begin{array}{l} \widehat{x}^+(k+1) = A\widehat{x}^+(k) + B_u u(k) + \pi^+(t, y) \\ \qquad\qquad + L^+\left(\widehat{y}^+(k) - y(k)\right), \quad \widehat{x}^+\left(k_0\right) = \widehat{x}^+_{k0}, \\ \widehat{y}^+(k) = C\widehat{x}^+(k), \end{array} \right. \quad (22)$$

$$\Phi_{O^-}: \left\{ \begin{array}{l} \widehat{x}^-(k+1) = A\widehat{x}^-(k) + B_u u(k) + \pi^-(t, y) \\ \qquad\qquad + L^-\left(\widehat{y}^-(k) - y(k)\right), \quad \widehat{x}^-\left(k_0\right) = \widehat{x}^-_{k0}, \\ \widehat{y}^-(k) = C\widehat{x}^-(k), \end{array} \right. \quad (23)$$

where $\widehat{x}^+(k)$ and $\widehat{x}^-(k)$ are the estimate states of Π_{O^+} and Π_{O^-}, respectively. The observer gains are $L^+ \in \mathbb{R}^{n \times q}$, $L^- \in \mathbb{R}^{n \times q}$.

The estimation errors are defined as $e^+(k) \triangleq \widehat{x}^+(k) - x(k)$, $e^-(k) \triangleq x(k) - \widehat{x}^-(k)$, and the output estimate errors as $\widetilde{y}^+(k) \triangleq \widehat{y}^+(k) - y(k)$, $\widetilde{y}^-(k) \triangleq y - \widehat{y}^-(k)$. Therefore, the error dynamics are given by

$$\Phi_{E^+}: \left\{ \begin{array}{l} e^+(k+1) = A_L^+ e^+(k) + w^+(k), \quad e^+(k_0) = e^+_{k0} \\ \widetilde{y}^+(k) = Ce^+(k) \\ w^+(k) = \pi^+(k, y(k)) - \pi(k, x(k)) \end{array} \right. \quad (24)$$

$$\Phi_{E^-}: \left\{ \begin{array}{l} e^-(k+1) = A_L^- e^-(k) + w^-(k), \quad e^-(k_0) = e^-_{k0} \\ \widetilde{y}^-(k) = Ce^-(k) \\ w^-(k) = \pi(k, x(k)) - \pi^-(k, y(k)) \end{array} \right. \quad (25)$$

where $w^+(k)$ and $w^-(k)$ are errors associated to the disturbance, which represent the exogenous inputs for the systems Φ_{E^+} and Φ_{E^-}.

An extension of *definition* is presented for the family of linear disturbed systems.

Definition 12. The estimator Φ_{O^+} (Φ_{O^+}) is an upper (lower) preserving partial-order observer for the disturbed system Φ_{P}, if satisfies the following conditions:

(i). The error system estimation Φ_{E^+} (Φ_{E^-}) is cooperative: if $\widehat{x}^+_{k0} \succeq x_{k0}$ ($x_{k0} \succeq \widehat{x}^-_{k0}$), then $\widehat{x}^+\left(k\right) \succeq x\left(k\right)$ ($x(k) \succeq \widehat{x}^-(k)$), $\forall k \in \mathbb{N}$.

(ii). The estimation error converges asymptotically to a ball centered in the origin with radius β, that is, $\|\widehat{x}^+(k) - x(k)\| \to \beta$ ($\|x(k) - \widehat{x}^-(k)\| \to \beta$) as $k \to \infty$.

Note that item (i) of the definition *12*, describes the characteristics of partial ordering between the state estimations and trajectories. This is provide because the system Φ_{E^+} (Φ_{E^-}) is cooperative, implying that conditions

$$A_L^+ \succeq 0 \qquad (A_L^- \succeq 0)$$

are satisfied, if the uncertain input $w^+(k) \succeq 0$ $(w^-(k) \succeq 0)$ $\forall k \in \mathbb{N}$, and the initial state $e_{k0}^+ \succeq 0$ $(e_{k0}^- \succeq 0)$.

3.1.2 Robust control design method

The purpose is to stabilize the perturbed system Φ_{P}, and to bound dynamically the real state trajectory by means of upper/lower estimations. Thus, we propose a controller described by

$$\Phi_{\mathrm{U}} : \{\, u(k) = K\widehat{x}^+(k) + K\widehat{x}^-(k) \tag{26}$$

which is based on the upper preserving partial-order observers Φ_{O^+} and lower Φ_{O^-}, where $L^+ \in \mathbb{R}^{p\times n}$, $L^- \in \mathbb{R}^{p\times n}$ and $K \in \mathbb{R}^{n\times m}$ are design matrices.

Using the estimation errors defined in the above paragraphs, the dynamics of the augmented system, composed by the closed-loop system and the estimation errors, are described of the form

$$\Phi_{\mathrm{LC}} : \left\{ \begin{bmatrix} x(k+1) \\ e^+(k+1) \\ e^-(k+1) \end{bmatrix} = \begin{bmatrix} A_{2K}, & -B_uK, & B_uK \\ 0 & A_L^+, & 0 \\ 0 & 0 & A_L^- \end{bmatrix} \begin{bmatrix} x(k) \\ e^+(k) \\ e^-(k) \end{bmatrix} + \begin{bmatrix} \pi(k) \\ w^+(k) \\ w^-(k) \end{bmatrix} \right.$$

where $A_{2K} = A - 2B_uK$.

The following stability result and output feedback controller design from Section 2 is straightforward.

Theorem 13. Consider the perturbed system Φ_P. Suppose that conditions (S1), (S2) and the inequality in (21) are satisfied. The system Φ_{LC} is globally ISS with respect to exogenous inputs, if the matrices K, L^+, L^-, $P_i = P_i^T > 0$ and the positive constants exist $\epsilon_i > 0$, $i = 1, 2, 3$ such the next conditions

- Cooperativeness: both A_L^+, A_L^- are nonnegative matrices,
- Practical stability, given by the inequality

$$\begin{bmatrix} \Theta_1, & \bigstar, & \bigstar \\ -K^TB_u^TP_1A_{2K} & \Theta_2, & \bigstar \\ KB_u^TP_1A_{2K} & -K^TB_u^TP_1B_uK, & \Theta_3 \end{bmatrix} \leq 0 \tag{27}$$

are satisfied with
$\Theta_1 = A_{2K}^TP_1A_{2K} - P_1 + \epsilon_1P_1$,
$\Theta_2 = K^TB_u^TP_1B_uK + {A_L^+}^T P_2A_L^+ - P_2 + \epsilon_2P_2$,
$\Theta_3 = K^TB_u^TP_1B_uK + {A_L^+}^T + P_3A_L^- + \left(A_L^-\right)^T P_3 + \epsilon_3P_3$
□

Proof. The demonstration of cooperativeness condition consists on the application of *Proposition 2* in the observation systems Φ_{E^+}, Φ_{E^-}, which implies,

$$A_L^- \succeq 0, \quad A_L^+ \succeq 0$$

For the second condition, its proposed a candidate Lyapunov function for the closed-loop system Φ_{LC},

$$V\left(x, e^+, e^-\right) = x^TP_1x + {e^+}^T P_2e^+ + {e^-}^T P_3e^- \tag{28}$$

The Lyapunov difference along the trajectory of Φ_{LC}, $\Delta V\left(x, e^+, e^-\right)$ is given by the expression

$$\begin{bmatrix} x \\ e^+ \\ e^- \end{bmatrix}^T \begin{bmatrix} \Theta_1, & -P_1B_uK, & P_1B_uK \\ -KB_u^TP_1 & \Theta_2, & 0 \\ KB_u^TP_1 & 0 & \Theta_3 \end{bmatrix} \begin{bmatrix} x \\ e^+ \\ e^- \end{bmatrix} \leq \begin{bmatrix} -\epsilon_1x^TP_1x \\ -\epsilon_2\left(e^+\right)^T P_2e^+ \\ -\epsilon_3\left(e^-\right)^T P_3e^- \end{bmatrix} + \begin{bmatrix} \pi^TP_1\pi \\ \left(w^+\right)^T P_2w^+ \\ \left(w^-\right)^T P_3w^- \end{bmatrix}$$

Re-arraying terms to complete squares. the augmented system Φ_{LC} is ISS with respect to the inputs $\pi(t)$, $w^+(t)$, $w^-(t)$. ■

The Theorem 13 represents a generalization of the Theorem 9, when the estimations of an interval observer are considered in the output feedback controller. However, this design can be applied to the family of nominal discrete systems, obtaining an exponential stability in closed-loop system; Additional the upper and lower estimation errors converges to zero.

4. COMPUTATIONAL IMPLEMENTATION

The Theorems *9* and *13* state the design of output feedback controllers based on preserving order (or interval) observers for a family of discrete-time linear systems, taking into account absence/presence of disturbances.

The conditions in (18) and in (27) guarantee the stability property for the augmented systems Π_{LC} and Φ_{LC}, respectively. The matrix inequality (18) is a special case for the condition (27), when is only considered a upper/lower preserving order observer. In general, the inequality in (27) is described as nonlinear matrix inequalities in the variables $(P_i, \epsilon_i, L^+, L^-, K)$ with $i = 1, 2, 3$. There exist cases in order to convert the nonlinear matrix inequality to the ambient of LMI's. For instance, changing the bilinear term ϵ_iP_i, $i = 1, 2, 3$ by the term ϵ_iI, and if (K, L^+, L^-) are fixed, then (27) becomes a LMI in (ϵ_i, P_i), $i = 1, 2, 3$. Using the Schur's Complement, the other results are obtained. The study of this matrix inequality represents analysis for further research.

The cooperativeness conditions, given by $A_L^+ \succeq 0$, $A_L^- \succeq 0$, $A_L \succeq 0$, nonnegative matrices, represent the partial-ordering the upper/lower estimations and state trajectory. This conditions are leaded to the LMI's; They are LMI's in the variables (L^+, L^-, L).

5. CONCLUSION

In this paper, we develop a design method of preserving order observer-based dynamic output-feedback controllers for a family of discrete-time linear systems, considering the absence and/or presence of disturbances/uncertainties.

It is shown that the resulting augmented system, which combines the closed-loop systems with observer error dynamics, is globally (ISS) exponentially stable using the upper/lower estimations, depending on the (presence) absence of disturbances. Additionally, it is ensured the partial ordering between estimations and state trajectories. The observers gain can be obtained by solving a Bilinear Matrix Inequality (BMI) and a Linear Matrix Inequality (LMI).

ACKNOWLEDGEMENTS

J.D. Avilés acknowledge the support of CONACyT Mexico in the frame of National Posdoctorals Stays (CVU 208496). J. Moreno gratefully acknowledge the financial support from Fondo de Colaboración del II-FI, UNAM, project IISGBAS-122-2014, PAPIIT-UNAM under grant IN113614, and CONACyT project 241171.

REFERENCES

Alcaraz-Gonzalez, V., Harmand, J., Rapaport, A., Steyer, J., Gonzalez-Alvarez, V., and Pelayo-Ortiz, C. (2002). Software sensors for highly uncertain wwtps: a new approach based on interval observers. *Water Res*, 36(2515).

Angeli, D. and Sontag, D. (2003). Monotone control systems. *IEEE Transactions Automatic*, 48(10), 1684–1698.

Avilés, J. and Moreno, J. (2009). Cooperative observers for nonlinear systems. *Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference, Shanghai, China*, 6125–6130.

Avilés, J. and Moreno, J. (2013). Preserving order observers for nonlinear systems. *International Journal of Robust and Nonlinear Control*, doi: 10.1002/rnc.2975.

Avilés, J. and Moreno, J. (2014). Diseño de observadores que preservan el orden para sistemas no lineales usando transformación de coordenadas. *Congreso Latinoamericano de Control Automático.*

Avilés, J. and Moreno, J. (2015a). Método para disenar controladores basados en observadores que preservan el orden parcial para sistemas lineales. *Congreso Nacional del AMCA, Cuernavaca, Morelos.*

Avilés, J. and Moreno, J. (2015b). Observadores que preservan el orden parcial para sistemas en tiempo discreto. *Congreso Nacional del AMCA, Cuernavaca, Morelos.*

Bernard, O. and Gouze, J. (2004). Closed loop observers bundle for uncertain biotechnological models. *Journal of Process Control*, 14(3), 765–774.

Efimov, D., Perruquetti, W., Raïssi, T., and Zolghadri, A. (2013a). Interval observers for time-varying discrete-time systems. *IEEE Transactions on Automatic Control*, 58(12), 3218–3224.

Efimov, D., Raissi, T., Perruquetti, W., and Zolghadri, A. (2015). Design of interval observers for estimation and stabilization of discrete-time lpv systems. *IMA Journal of Mathematical Control and Information*, dnv023.

Efimov, D., Raissi, T., and Zolghadri, A. (2013b). Control of nonlinear and lpv systems: interval observer-based framework. *IEEE Transactions on Automatic Control.*

Gouze, J.L., Rapaport, A., and Hadj-Sadok, M.Z. (2000). Interval observers for uncertain biological systems. *Ecol Modelling*, 133(1-2), 45–56.

Hirsch, M. and Smith, H. (2005). Monotone maps: a review. *Journal of Difference Equations and Applications*, 11(4-5), 379–398.

Hirsch, M. and Smith, H. (2004). *Monotone Dynamical Systems.* Handbook of differential equations: ordinary differential equations. Vol. II, Elsevier B. V., Amsterdam 239-357.

Jiang, Z.P. and Wang, Y. (2001). Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37(6), 857–869.

Khalil, H.K. (2002). *Nonlinear Systems.* Prentice Hall, New York, USA, third edition.

Mazenc, F., Dinh, T., and Niculescu, S. (2013). Robust interval observers for discrete time systems of luenberger type. *ACC, Washington USA*, 2484–2489.

Mazenc, F., Dinh, T., and Niculescu, S. (2014). Interval observers for discrete time systems. *International Journal of Robust and Nonlinear Control*, (17), 2867–2890.

Raïssi, T., Efimov, D., and Zolghadri, A. (2012). Interval state estimation for a class of nonlinear systems. *IEEE Trans. on Aut. Control*, 57(1), 260–265.

Detection and Diagnosis of Breast Tumors using Deep Convolutional Neural Networks

J. D. Gallego-Posada. * **D. A. Montoya-Zapata.** ** **O. L. Quintero-Montoya.** ***

* *Research Group on Mathematical Modeling, Universidad EAFIT, Medellín, Colombia (e-mail: jgalle29@eafit.edu.co)*
** *Research Group on Mathematical Modeling, Universidad EAFIT, Medellín, Colombia (e-mail: dmonto39@eafit.edu.co)*
*** *Research Group on Mathematical Modeling, Universidad EAFIT, Medellín, Colombia (e-mail: oquinte1@eafit.edu.co).*

Abstract: We present an application of Deep Convolutional Neural Networks (CNN) for the detection and diagnosis of breast tumors. The images used in this study have been extracted from the mini-MIAS database of mammograms. The proposed system has been implemented in three stages: (a) crop, rotation and resize of the original mammogram; (b) feature extraction using a pretrained CNN model (AlexNet and VGG); (c) training of a Support Vector Machine (SVM) at the classification task using the previously extracted features. In this research, the goal of the system is to distinguish between three classes of patients: those with benign, malign or without tumor. Experiments show that feature extraction using pretrained models provides satisfactory results, achieving a 64.52% test accuracy. It is worth noting the impact of the data augmentation process and the balance of the number of examples per class on the performance of the system.

Keywords: Breast tumor, classification, convolutional neural network, mammogram, support vector machine

1. INTRODUCTION

Breast cancer is the most common cancer in women and is commonly thought to be a disease of the developed world but nearly 50% of breast cancer cases and 58% of deaths occur in less developed countries. It is estimated that around the world over 508.000 women died in 2011 due to this condition. According to the World Health Organization (2016), detection of breast cancer in its early stages dramatically increases the chances of establishing a successful treatment plan.

As part of the current efforts to control this condition, the development of computer-aided diagnosis systems which can assist medical personnel with the early detection of tumors pose a crucial alternative. In such systems a high reliability in the accuracy of the classifier is a top priority.

In the study by Suckling et al. (1994), the diagnosis was performed employing a SVM trained with features extracted using AlexNet and VGG pretrained models fed with preprocessed mammograms. Our data source is the database of the Mammographic Image Analysis Society (MIAS).

The paper is structured as follows: in Section 2 we provide a review of the application of Deep Learning techniques to the image classification problem. Section 3 presents an outline of previous studies of breast cancer detection and classification using Deep Learning and Artificial Intelligence-based approaches. In Section 4, the employed methodologies are described. Next, in Section 5 the specifications of the implemented system are presented. Finally, Section 6 contains the main conclusions of this work and some possibilities for future improvements on this research.

2. RELATED WORK

The study of computer-aided breast cancer diagnosis has been addressed from several perspectives. The aim of this section is to briefly illustrate the state of the art in this field using artificial intelligence and additionally using strictly Deep Learning-related techniques.

2.1 Analysis of Breast Cancer using Artificial Intelligence Techniques

Alolfe et al. (2009) used a SVM and linear discriminant analysis to distinguish between benign and malign tumors on the MIAS database. Using this approach, they classified 90% and 87.5% of benign and malignant images correctly, respectively. A region of interest (ROI) of 32×32 pixels was selected from the images and 224 features were extracted. These features were divided into five groups: wavelet, first order statistics, second order statistics, shape and fractal dimension data. Finally, 13 features were selected with the forward stepwise linear regression method.

Wang et al. (2014) used the mammographies from 482 patients to compare the accuracies from an extreme learning machine (ELM) and a SVM to classify between images with and without tumors. In the preprocessing stage a median filter was used to reduce the noise and the wavelet

transformation of local modulus maxima in conjunction with the region growing algorithm were used as edge segmentation method. Finally, five textural features and five morphological features were extracted from the resulting image and these were used at the classification task. The ELM classifier exhibited better performance than the SVM classifier.

Dheeba et al. (2014) obtained an accuracy of 93.67% classifying between normal and abnormal tissues with an optimized neural network using Particle Swarm Optimization. The experiment was carried out with their private database of mammograms and the classification was done with the Laws Texture Energy Measures extracted from a ROI of dimension 15×15 pixels.

Peng et al. (2016) obtained an accuracy of 96% using an artificial neural network to classify the mammograms from MIAS database. They defined three different categories to carry out the experiment: normal, with presence of a benign tumor and with presence of a malign tumor. A median filter and the seeded region growing algorithm were used to remove the noise of the original images. Then, they extracted 16 features related to the texture properties of the images and five of them were selected. The feature selection algorithm, which is based on the rough-set theory, was developed by the authors.

Mahersia et al. (2016) achieved recognition rates of 97.08% and 95.42% on the MIAS database using a neural network with a Bayesian back-propagation algorithm and an ANFIS system as classifiers, respectively. The breasts were classified into two categories: normal and cancerous. The mammograms from this database were first enhanced, removing the noise and details that may interfere with the recognition of the tumors. Then a generalized Gaussian density model for wavelet coefficients was used as feature extractor.

2.2 Analysis of Breast Cancer using Deep Learning

Ertosun and Rubin (2015) used three different architectures of CNNs to locate masses in mammography images. They selected 2420 images from the DDSM dataset and divided these images into training, validation and test sets, containing 80%, 10% and 10% of the images, respectively. They also used cropping, translation, rotation, flipping and scaling techniques to get an augmented training set, in order to improve the generalization ability of the system. The experiment was divided into two stages: the first consisted in the classification of a mammography as containing or not masses and the second in the localization of masses in the images.

Arevalo et al. (2015) obtained 86% of area under the Receiver Operating Characteristic (ROC) curve by classifying mammography mass lesions using a CNN as feature extractor and a SVM as classifier. The data to carry out the experiment was the BCDR-F03 dataset, which is part of the BCDR database. This data was composed by 736 images, 426 containing benign mass lesions and the rest containing malignant lesions. The data augmentation was achieved by flipping and rotating the images. In addition, the mammography images were normalized by the use of global and contrast normalization. The CNN was trained using both dropout and max-norm regularization techniques.

Jiao et al. (2016) obtained an accuracy of 96.7% classifying the breast masses between benign and malign from the DDSM database using a CNN as feature extractor and a SVM as classifier. The images were previously normalized and whitened. On the other hand, the CNN was trained with a subset of ImageNet, dataset produced by Russakovsky et al. (2015), and the features to perform the classification were extracted from two different layers of the CNN.

Abdel-Zaher and Eldeib (2016) developed a classifier using the weights of a previously trained deep belief network as the initial parameters for a neural network with Liebenberg Marquardt learning function. This model was tested on the Wisconsin Breast Cancer Dataset, obtaining an accuracy of 99.68%.

3. DEEP LEARNING FOR IMAGE CLASSIFICATION

This section is based on the works from Guo et al. (2015) and LeCun et al. (2015).

Around 2006, the results obtained by a group of researchers working together in parallel projects in the Canadian Institute for Advanced Research renovated the interest of the community for the deep neural networks. The main four works Bengio et al. (2006); Hinton (2005); Hinton et al. (2006); Marc'Aurelio Ranzato et al. (2006), introduced unsupervised learning procedures to pure supervised learning procedures. The objective of each layer in the neural network was to learn the inputs of the previous layer, as stated by LeCun et al. (2015). This approach performed well in comparison with the existent artificial intelligence techniques in tasks such as recognizing handwritten digits, specially when the amount of labeled data was limited, as mentioned by Sermanet et al. (2013).

Since the rise of Deep Learning, the CNN model outperformed the fully connected neural networks in tasks related to natural image classification. However, this approach was not seriously used at classification problems until 2012. During six years in which CNNs were laid aside, the methods based on the *Bag of Visual Words* model, that were the state of the art techniques for image classification, were improved by Lazebnik et al. (2006) with the incorporation of spatial geometry, through the use of spatial pyramids.

The turning point for image classification was 2012. In this year, AlexNet, a CNN with five convolutional layers and three fully connected layers developed by Krizhevsky et al. (2012), outperformed the existing methodologies and won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012, almost halving the error rate of the model in second place from Russakovsky et al. (2015). According to LeCun et al. (2015), this success reflected the new developments in graphic hardware and algorithms: the increased chip processing abilities (GPU units), the use of Rectified Linear Unit (ReLU) as neural activation functions, a novel regularization technique developed by Srivastava et al. (2014) and the advances in algorithms for data augmentation.

Since the success of AlexNet in 2012, several improvements of this model have been performed. Zeiler and Fergus (2014) established a technique to analyze the responses of intermediate layers, what enabled them to implement Clarifai, winning the ILSVRC.

In 2014, deeper architectures were finally used. VGG from Simonyan and Zisserman (2015) and GoogLeNet from Szegedy et al. (2015) networks obtained the second and first place in ILSVRC, respectively. The VGG network from had 13-16 convolutional layers, while GoogLeNet had 21 convolutional layers.

He et al. (2015b) proposed a model that surpassed for the first time human-level performance on the ImageNet 2012 test dataset, with a network with the same architecture of VGG. In addition, He et al. (2015a) also established a new framework to train deeper networks called the residual learning. They developed ResNet, a 152-layers network and won ILSVRC.

For Guo et al. (2015), researchers are focusing in three main aspects to further improve the performance of Deep Learning models: (a) the implementation of larger networks: ResNet, GoogLeNet and VGG models have shown that the networks with a larger number of layers outperform the simpler ones; (b) the use of multiple networks, where every network can execute all the process independently, so the responses of all the networks are combined in order to obtain the final result; and (c) the introduction of external information from other resources and the use of shallow structures. In this aspect, one of the most important developments is *Regions with CNN Features* method by Girshick et al. (2014), in which the features extracted from a CNN feed a SVM.

Other research projects have focused their efforts on getting a further understanding of what deep neural networks learn, addressing the problem from both a theoretical and a empirical perspective. For instance, Li et al. (2016) have recently studied convergent learning, aiming to analyze cases in which different neural networks learn similar representations. In this work, they propose a method for quantifying the similarity between deep neural networks and showed that there exist basic features which are learned by multiple networks with the same architectures but different random initialization.

4. THEORETICAL BACKGROUND

4.1 Convolutional Neural Networks

CNNs are a type of biologically-inspired feed-forward networks characterized by a sparse local connectivity and weight sharing among its neurons. A CNN can also be seen as a sequence of convolutional and subsampling layers in which the input is a set of $H \times W \times D$ images, where H is the height, W is the width and D is the number of channels which, in the case of RGB images corresponds to $D = 3$.

Following Ng et al. (2010), a typical convolutional layer (volume) is formed by K filters (kernels) of size $F \times F \times D$, where $F \leqslant H$ and $F \leqslant W$. These filters are usually randomly initialized and are the parameters to be tuned in the training process. Since the size of the filter is generally strictly smaller than the dimensions of the image, this leads to a local connectivity structure among the neurons. Each of this convolutional volumes has an additional hyper-parameter, S, which corresponds with the stride that the filter is going to slide spatially in the image.

Let's denote a particular training example as $X_{H\times W\times D}$ and a convolution filter $W_{F\times F\times D}$. As it is familiar from the usual Multi-Layer Perceptron, it is customary to add a bias term b to each of the linear combinations formed. Finally, a (commonly non-linear) activation function, for example ReLU, is applied to the convolution between the input image and the kernels, which yields an activation map A of the dimensions $1 + \frac{N-F}{S} \times 1 + \frac{N-F}{S} \times 1$:

$$A = f(X \circledast W + b)$$

where $\circledast$ represents the *valid* convolution between the operands and f is the activation function.

Appending the activation maps found by applying K diferent kernels to the input example, an activation volume of dimensions $1+\frac{N-F}{S}\times 1+\frac{N-F}{S}\times K$ is obtained. Note that depending on the dimensions of the image, the filter and the size of the stride, the resulting activation volume may reduce its spatial dimensions very quickly. An alternative to control this situation in advance is the use of *padding* techniques to the original image.

Finally, in order to perform dimensionality reduction directly on the data, pooling layers are applied to an activation volume or even the input image itself. These layers subsample its inputs, typically with mean or max pooling, over contiguous regions of size $P \times P$.

Figure 1 shows an example of a typical architecture for a CNN in which two convolutional and two pooling layers are applied to the original image. In this case, the extracted features obtained as are fed into a fully connected layer to perform the classification task. Note that it is possible to change the classifier set up at the end of the network with, for example, a SVM or a softmax classifier.

4.2 Back-propagation Algorithm

The summary presented in this section is heavily based on the Unsupervised Feature Learning and Deep Learning Tutorial from Ng et al. (2010). For simplicity, we will illustrate the algorithm assuming that we have a CNN with the input layer followed by a convolutional volume, a pooling layer and finally a fully connected layer.

Let's denote by $\delta^{(l+1)}$ the error term in the $(l + 1)$-th layer in the network with labeled training data (x, y), parameters (W, b) and cost function $J(W, b; x, y)$. If the l-th layer is densely connected to the former, the error for this layer can be computed by:

$$\delta^{(l)} = \left((W^{(l)})^t \delta^{(l+1)}\right) \bullet f'(z^{(l)})$$

where $\bullet$ represents element-wise multiplication and f is the activation function.

The gradients are:

$$\nabla_{W^{(l)}} J(W, b; x, y) = \delta^{(l+1)}(a^{(l)})^t$$

$$\nabla_{b^{(l)}} J(W, b; x, y) = \delta^{(l+1)}$$

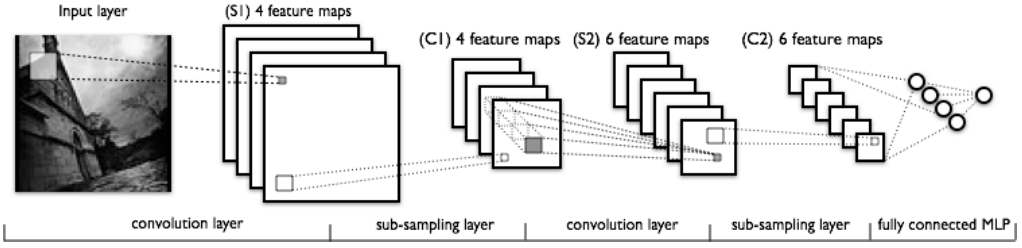


Fig. 1. Example of a CNN architecture. Taken from: LISA Lab (2016).

If the l-th layer is a convolutional and subsampling layer, then the error is propagated through as:

$$\delta_k^{(l)} = upsample\left((W_k^{(l)})^t \delta_k^{(l+1)}\right) \bullet f'(z_k^{(l)})$$

where k indexes the filter number and the *upsample* function propagates the error through the pooling layer by calculating the error related to each input unit.

Finally, the gradient for each filter map can be found by:

$$\nabla_{W_k^{(l)}} J(W, b; x, y) = \sum_{i=1}^{m} (a_i^{(l)}) \circledast rot90(\delta_k^{(l+1)}, 2)$$

$$\nabla_{b_k^{(l)}} J(W, b; x, y) = \sum_{a,b} \left(\delta_k^{(l+1)}\right)_{a,b}$$

where $a^{(l)}$ is the input to the l-th layer and $rot90(A, k)$ rotates the input array A counterclockwise by $k * 90$ degrees.

4.3 Linear Support Vector Machines

Suppose we are given a training data set of size n examples of the form:

$$\{(X_1, y_1), (X_1, y_1), ..., (X_n, y_n)\}$$

where each y_i is either 1 or -1 and each X_i is a p-dimensional vector. Thus, assuming that the data is linearly separable, we want to find the hyperplane that separates the group of $\{X_i\}$ for which $y_i = 1$ from those for which $y_i = -1$ so that the distance between the hyperplane and the nearest point from either group is maximized. For that reason, it is also called a *maximum-margin classifier*. This can be formally expressed as:

$$\min_{w \neq 0, b} \frac{1}{2} ||w||^2$$

$$\text{s.t.} \quad y_i(w^t X_i + b) \geqslant 1 \quad (i = 1, 2, ..., n)$$

. Recall that $\frac{b}{||w||}$ represents the separation of the hyperplane from the origin along the normal vector w when the hyperplane is expressed as $wX - b = 0$.

4.4 Confusion Matrix

Consider a classification problem with only two classes: positive (P) and negative (N). For every training example, there are only four possible outcomes. If the training example is positive and the prediction is positive, we call it a *true positive*; and if the prediction is negative, it is called a *false negative*. On the other hand, if the training example is negative and it is classified as negative, it is called a *true negative*; otherwise, it is a *false positive*. Table 1 displays an example of a confusion matrix for a two-class problem.

Table 1. Confusion Matrix

		Predicted Class	
		P	**N**
Actual Class	**P**	True Positives	False Negatives
	N	False Positives	True Negatives

Fawcett (2006) defines a confusion matrix as a tool that allows to visualize the performance of a classifier in a supervised learning problem. By means of this matrix it is possible to asses whether the system is commonly confusing pairs of classes. In the aforementioned problem, the confusion matrix summarizes the four possible outcomes from the classifier.

5. MAMMOGRAMS CLASSIFICATION

5.1 Data

The mammograms used for the commitment of this work were retrieved from the database of the MIAS, collected by Suckling et al. (1994), which is known as mini-MIAS since the images of the original MIAS database has been reduced to 200 micron pixel edge and the dimension of the mammograms has been fixed to 1024×1024 pixels. This database contains 322 mammograms and the intensity of every pixel is between 0 and 255. This database also includes information about the class and the severity of abnormalities that may be present in the mammograms, as well as the coordinates of the center of these abnormalities.

It must be mentioned that we only used the mammogram images and the required information to divide the mammograms into three categories: patients with benign, malign or without tumor.

5.2 Data Preprocessing

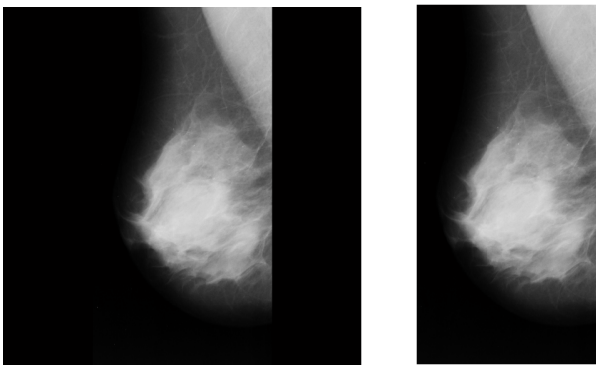
Mammogram Cropping Mammograms contain black zones in the borders which may difficult the classification task. For this reason, we designed an algorithm to eliminate these black zones based on the sum of the pixels over the column. The algorithm finds the first column, say C_l, on the left of the mammogram in which the sum of the pixels exceeds a given threshold P. Now, from this point, the algorithm finds the first column, C_r, in which the sum of the pixels is not greater than P. Then, the new image is the one enclosed between C_l and C_r. This algorithm was applied to every mammogram of the 322 retrieved from the aforementioned database, taking $P = 500$. An example of

the images obtained at this stage is illustrated in Figure 2, in which Figure 2a is an original mammogram of the mini-MIAS database and Figure 2b is the resultant image after the application of the cropping algorithm.

Data Augmentation Due to the lack of mammograms corresponding to malign tumors (51 out of 322), it was necessary to perform a data augmentation operation in order to get a balanced dataset with at least 600 mammograms. For this purpose, after the cropping procedure, every resultant mammogram was rotated $-90°$, $90°$ and $180°$. The label assigned to the three artificially generated mammograms corresponded with the label of the original image.

5.3 Feature Extraction

As mentioned in Sections II and III, CNNs are being widely used to carry out image classification tasks because of their outstanding performance in comparison with other classification techniques. For this reason, they have become an emerging alternative in the computer-aided diagnosis field.



(a) Original. (b) Cropped.

Fig. 2. Mammograms obtained after the cropping stage.

In this work, two different experiments were carried out using a CNN previously trained on the ImageNet database as feature extractor. In the first experiment, the CNN used was AlexNet from Krizhevsky et al. (2012) while in the second experiment the CNN used was VGG-F from Chatfield et al. (2014). The features selected to perform the classification of mammograms were the activations of the last convolutional layer of the CNN. Then, in both cases, 4096 features have been extracted for each image.

In order to feed both pretrained CNNs with the cropped images, it was necessary to convert every mammogram into a three channel image by repeating the single channel three times. Then, the resulting image was resized depending on the input dimension of the CNN (227×227 pixels for AlexNet and 224×224 pixels for VGG-F). Finally, the average image (which is included with the tuned parameters of the pretrained models used) was subtracted from the resized image.

5.4 Classification

The goal of the system was to distinguish between three classes: patients with benign, malign or without tumor. Therefore, based on the works from Alolfe et al. (2009); Arevalo et al. (2015); Jiao et al. (2016), we decided to adopt a SVM as our classifier.

In order to evaluate our methodology, 120 and 80 mammograms of each category were selected from the augmented dataset to define the training and test stages of the SVM, respectively. Hence, our training set was composed by 360 mammograms and our testing set by 240 mammograms.

To carry out the training of the SVM, each of the 360 mammograms selected was given as the input for the CNN and the features obtained at this step became the inputs for the SVM. Then, using the Statistics and Machine Learning Toolbox from MATLAB, the SVM was trained.

The classification accuracy of the trained SVM was evaluated with the 240 mammograms belonging to the test set, following the same process described above for the extraction of the features for every mammogram.

5.5 Results

In Table 2 the confusion matrix obtained using AlexNet as feature extractor without augmenting the dataset is shown. This experiment was carried out with a training set of 30 mammograms per category and a test set of 20 per category. This confusion matrix is based on the response of the system on the test set and this low accuracy rate of only 35%, which corresponds to 21 well classified mammograms of the 60 that conformed the test set, is an evidence of the necessity of performing a data augmentation operation.

Table 2. Confusion matrix for MIAS test set predictions and feature extraction using AlexNet.

		Target			
		Benign	**Malign**	**Normal**	**Total**
Output	**Benign**	**36.53**	48.12	15.35	**36.53**
	Malign	27.39	**56.12**	16.49	**56.12**
	Normal	31.34	56.29	**12.36**	**12.36**
	Total	**38.35**	**34.96**	**27.97**	**35.01**

In Tables 3 and 4 the confusion matrices corresponding to the response of the system when AlexNet and VGG-F CNNs are used as feature extractors in conjunction with a SVM as classifier are exhibited. Table 3 shows the accuracy of the system on the test set after performing the data augmentation when the CNN used is AlexNet.

Table 3. Confusion matrix for Augmented MIAS test set predictions and feature extraction using AlexNet.

		Target			
		Benign	**Malign**	**Normal**	**Total**
Output	**Benign**	**61.79**	20.33	17.87	**61.79**
	Malign	18.79	**61.75**	19.46	**61.75**
	Normal	22.88	20.67	**56.46**	**56.46**
	Total	**59.73**	**60.10**	**60.20**	**60.01**

On the other side, Table 4 shows the response of the system when VGG-F is the feature extractor. It can be noted that the performance of the system has dramatically increased after artificially augmenting the dataset: from 35% to 60.01% and 64.52% using AlexNet and VGG-F, respectively.

Table 4. Confusion matrix for Augmented MIAS test set predictions and feature extraction using VGG.

		Target			
		Benign	**Malign**	**Normal**	**Total**
Output	**Benign**	**63.63**	18.45	17.92	**63.63**
	Malign	17.86	**64.37**	17.77	**64.37**
	Normal	16.91	17.54	**65.55**	**65.55**
	Total	**64.66**	**64.14**	**64.75**	**64.52**

6. CONCLUSIONS

Based on the results obtained in this work, the Deep Learning approach, particularly using pretrained CNNs as feature extractors, is a promising methodology when addressing the problem of diagnosing breast cancer with mammogram images. Since in this context the reliability of the system is highly relevant, it is desirable to increase the achieved 64.52% test accuracy. This outcome could be improved via fine-tuning of the final layers or training the whole network parameters. Additionally, it is worth noting the impact of the data augmentation process and the balance of the number of examples per class on the performance of the system.

The implemented system has three main advantages: (a) the mammograms are classified directly as with benign or malign tumor and without tumor, (b) it is not necessary to define a specific area in which the tumor is located and (c) apart from the mammogram, additional information must not be provided.

Future research could be focused on the evaluation of the following techniques:

- To extract features from multiple layers of the CNN instead of only using the activations obtained from the last convolutional layer.
- To use different pretrained CNNs as feature extractors, such as GoogLeNet from Szegedy et al. (2015) or ResNet from He et al. (2015a).
- To include a feature selection phase in which the best extracted features from a CNN could be selected to perform the classification of the mammograms.
- To test other classifier structures: neural networks, fuzzy inference systems or clustering techniques.

REFERENCES

Abdel-Zaher, A.M. and Eldeib, A.M. (2016). Breast cancer classification using deep belief networks. *Expert Systems with Applications*, 46, 139 – 144. doi: http://dx.doi.org/10.1016/j.eswa.2015.10.015.

Alolfe, M.A., Mohamed, W.A., Youssef, A.B.M., Mohamed, A.S., and Kadah, Y.M. (2009). Computer aided diagnosis in digital mammography using combined support vector machine and linear discriminant analysis classification. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2609–2612. doi: 10.1109/ICIP.2009.5413992.

Arevalo, J., González, F.A., Ramos-Pollán, R., Oliveira, J.L., and Lopez, M.A.G. (2015). Convolutional neural networks for mammography mass lesion classification. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 797–800. doi:10.1109/EMBC.2015.7318482.

Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. (2006). Greedy Layer-Wise Training of Deep Networks. *Advances in neural information processing systems*, 19(1), 153. doi:citeulike-article-id:4640046.

Chatfield, K., Simonyan, K., Vedaldi, A., and Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. In *British Machine Vision Conference*.

Dheeba, J., Singh, N.A., and Selvi, S.T. (2014). Computer-aided detection of breast cancer on mammograms: A swarm intelligence optimized wavelet neural network approach . *Journal of Biomedical Informatics*, 49, 45 – 52. doi:http://dx.doi.org/10.1016/j.jbi.2014.01.010.

Ertosun, M.G. and Rubin, D.L. (2015). Probabilistic visual search for masses within mammography images using deep learning. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, 1310–1315. doi:10.1109/BIBM.2015.7359868.

Fawcett, T. (2006). An introduction to ROC analysis . *Pattern Recognition Letters*, 27(8), 861 – 874. ROC Analysis in Pattern Recognition .

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 580–587. doi: 10.1109/CVPR.2014.81.

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., and Lew, M.S. (2015). Deep learning for visual understanding: A review. *Neurocomputing*. doi: 10.1016/j.neucom.2015.09.116.

He, K., Zhang, X., Ren, S., and Sun, J. (2015a). Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*.

He, K., Zhang, X., Ren, S., and Sun, J. (2015b). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *Proceedings of the ICCV*, 1–11. doi:10.1109/ICCV.2015.123.

Hinton, G.E. (2005). What kind of graphical model is the brain? In *International Joint Conference on Artificial Intelligence*, volume 5, 1765–1775.

Hinton, G.E., Osindero, S., and Teh, Y.W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7), 1527–54. doi:10.1162/neco.2006.18.7.1527.

Jiao, Z., Gao, X., Wang, Y., and Li, J. (2016). A deep feature based framework for breast masses classification. *Neurocomputing*, 197, 221 – 231. doi: http://dx.doi.org/10.1016/j.neucom.2016.02.060.

Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances In Neural Information Processing Systems*, 1–9. doi: http://dx.doi.org/10.1016/j.protcy.2014.09.007.

Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, 2169–2178. doi:10.1109/CVPR.2006.68.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. doi: 10.1038/nature14539.

Li, Y., Yosinski, J., Clune, J., Lipson, H., and Hopcroft, J. (2016). Convergent Learning: Do different neural networks learn the same representations? In *ICLR*, 1–21.

LISA Lab (2016). *My LeNet*. Retrieved 2016-5-26.

Mahersia, H., Boulehmi, H., and Hamrouni, K. (2016). Development of intelligent systems based on Bayesian regularization network and neuro-fuzzy models for mass detection in mammograms: A comparative analysis . *Computer Methods and Programs in Biomedicine*, 126, 46 – 62. doi:http://dx.doi.org/10.1016/j.cmpb.2015.10.017.

Marc'Aurelio Ranzato, C.P., Chopra, S., and LeCun, Y. (2006). Efficient learning of sparse representations with an energy-based model. In *Advances in Neural Information Processing Systems*, volume 19, 1137–1144.

Ng, A., Ngiam, J., Foo, C.Y., Mai, Y., and Suen, C. (2010). UFLDL tutorial. *http://ufldl.stanford.edu/wiki/index.php/UFLDL_Tutorial*.

Peng, W., Mayorga, R., and Hussein, E. (2016). An automated confirmatory system for analysis of mammograms. *Computer Methods and Programs in Biomedicine*, 125, 134 – 144. doi: http://dx.doi.org/10.1016/j.cmpb.2015.09.019.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3), 211–252. doi:10.1007/s11263-015-0816-y.

Sermanet, P., Kavukcuoglu, K., Chintala, S., and LeCun, Y. (2013). Pedestrian detection with unsupervised multi-stage feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3626–3633.

Simonyan, K. and Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *Proceedings of the ICLR*, 1–14. doi: 10.1016/j.infsof.2008.09.005.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1), 1929–1958.

Suckling, J., Parker, J., Dance, D., Astley, S., Hutt, I., Boggis, C., Ricketts, I., Stamatakis, E., Cerneaz, N., Kok, S., et al. (1994). The mammographic image analysis society digital mammogram database. In *Exerpta Medica. International Congress Series*, volume 1069, 375–378.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9.

Wang, Z., Yu, G., Kang, Y., Zhao, Y., and Qu, Q. (2014). Breast tumor detection in digital mammography based on extreme learning machine. *Neurocomputing*, 128, 175 – 184. doi: http://dx.doi.org/10.1016/j.neucom.2013.05.053.

World Health Organization (2016). Breast cancer: prevention and control. URL `http://www.who.int/cancer/detection/breastcancer/en/`. [Accessed: 19- May- 2016].

Zeiler, M.D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *Proceedings of the ECCV International Workshop on Statistical Learning in Computer Vision*, 818–833. Springer.

Dissolved Oxygen Dynamic Model for Endospore-Forming Bacteria batch bioprocess

Adriana N. Amicarelli, Alex Alzate and Fernando A. di Sciascio

Instituto de Automática UNSJ - CONICET, Av. San Martín (Oeste) 1109 Argentina (Tel.: +54 264 4213303 / Fax: +54 264 4213672; e-mail: inaut@inaut.unsj.edu.ar).

Abstract: This paper extends and generalizes a model of Dissolved Oxygen (DO) dynamic developed for the δ - endotoxins of *Bacillus thuringiensis* (*Bt*) batch production process to a DO dynamical model that can be used for a broad class aerobic Endospore-forming Bacteria (AEFB). The most significant feature of this type of microorganism is the endospores generation in their life cycle. The generalization of the model allows to use the same parametric model structure for process control and state estimation purposes in a considerable variety of AEFB batch bioprocesses by selecting an appropriate set of parameters for each specific bioprocess.

Keywords: Bioprocess Model, Aerobic Endospore-Forming Bacteria (AEFB), Batch Process, Control Process, State Estimation.

1. INTRODUCTION

In a previous work (Amicarelli et al., 2010) the authors developed a dissolved oxygen (DO) dynamics model to complete a pre-existing model (Atehortúa et al., 2007) for the δ - endotoxins of *Bacillus thuringiensis* (*Bt*) batch production process. The inclusion of this dynamic is useful for control purposes since the dissolved oxygen in the culture medium is a key variable to maintain the aerobic microorganism population in adequate levels and conditions for the effectiveness of the fermentation process. This mentioned model was successfully considered in subsequent works to propose different DO control strategies through different approaches i) control based on **Lyapunov theory** (Amicarelli et al., 2010), ii) control based on **nonlinear dynamic inversion** (Rómoli et al., 2016), iii) **Predictive Control Based Model** (CPBM) (Alzate et al., 2016) and **classical controllers** as PID, PI. Is important also to mention that dissolved oxygen concentration can also be used for estimating the microorganisms concentration in the culture medium when the oxygen consumption for the microorganism is known (Amicarelli et al., 2015; A Amicarelli et al., 2014; di Sciascio & Amicarelli, 2008; Rómoli et al., 2016). As was mentioned early, model was initially developed for *Bt* batch fermentation but is possible to extend the use of this model for a more general class of microorganism named: Aerobic Endospore-forming Bacteria (AEFB). The Endospore-forming Bacteria (EFB) are microorganism as *Bacillus* and *Clostridium* that can survive in hostile environments by producing endospores and then rapidly germinating to vegetative cells and growth when they encounter favorable environmental conditions. *Bacillus* genus are Gram-positive, aerobic or facultative Endospore-forming Bacteria, that is, *Bacillus* are organism capable of growth in the presence of oxygen, and forms a type of resting cell called endospore. *Bacillus* is a large group, including *Bacillus cereus*, *Bacillus clausii*, *Bacillus halodenitrificans*, *Bacillus subtilis*, *Bacillus thuringiensis*, among others. *Bacillus* spores, also called endospores, are resistant to harsh chemical and physical conditions. *Clostridium*, *Sporolactobacillus* are group of anaerobic spore forming bacteria. *Clostridium* genus consists of more than a hundred known species such as *Clostridium perfringens*, *Clostridium botulinum*, *Clostridium difficile*, *Clostridium tetani* and *Clostridium sordellii*. *Clostridium thermocellum* are used commercially to produce ethanol, *Clostridium acetobutylicum* to produce acetone, and *Clostridium diolis* to convert fatty acids to yeasts and propanediol, among others. The processes that involve AEFB have two classes of cells: vegetative cells and sporulated cells. The proposed dissolved oxygen model (Amicarelli et al., 2010) allows estimating or quantifying how much oxygen is consumed for vegetative growth and for cell maintenance for *Bt* fermentations. When the EFB are in the vegetative state, their growth, as well as their substrate utilization and decay processes are almost the same as for common bacteria. When the culture medium exhibit nutritional limitation, the EFB sporulate (Atehortúa et al., 2006; Errington, 2003; Hoch, 1993; Sonenshein, 2000). In general, the bioprocess models available in literature normally include cellular dynamics: the main substrate and generation of product dynamics. However, dissolved oxygen is not considered as a part of the model. Nevertheless, it is feasible to include this dynamic since it can be easily measured online. Furthermore, this is a useful variable for the manipulation as control action of bioprocess.

In this work, the authors extend the use of the DO model for control and estimation purposes, to other batch bioprocess with similar characteristics with the aim to achieve other products. Example of bioprocess with other finalities can be

the production of biosurfactant by *Bacillus subtilis* (Davis et al., 2001; Yeh et al., 2006). The genus *bacillus* have general features that include the degradation of substrates derived from vegetal and animal sources including cellulose, starch, pectin, proteins, agar, hydrocarbons, and others; antibiotic production; nitrification; denitrification; nitrogen fixation; facultative lithotrophy; autotrophy; acidophily; alkaliphily; psychrophily; thermophily; and parasitism (Todar, 2006). Three microorganisms are currently in use for industrial riboflavin production. The *hemiascomycetes Ashbya gossypii*, a *lamentous fungus*, and *Candida famata* are naturally producers of this vitamin (Stahmann et al., 2000). Vitamins, C, B2 (riboflavin), vitamin B12, linolenic acid, and ergosterol are produced by the use of microorganisms. Gram positive bacteria, producing inhibitory substances like cyclic peptides and bacteriocins, with a broad antimicrobial spectrum and a history of safe use in food (Cleveland et al., 2001; Holzapfel et al., 1995). A relevant related paper is the work of Park, Rittmann, and Bae (Park et al., 2009) when it is developed a model called Life-Cycle Kinetic Model for Endospore – Forming Bacteria. This model includes Germination, Sporulation and dissolved oxygen dynamics. However, the expression and balances are impractical for control an estimation purposes due to their complexity and the difficulties to measure each involved variable online. Motivated in this fact, in the present work, the authors report the use of a simple parametric dynamic model for these objectives allowing better control of dissolved oxygen concentration in the culture medium of this class of bioprocesses.

This paper is organized as follows: Section 2 presents the model and it generalization for AEFB. Section 3 presents validations model through experimental data and model comparisons. Finally, in Section 4 conclusions are stated.

2. GENERALIZATION OF THE PROPOSED DISSOLVED OXYGEN DYNAMIC MODEL

The standard model for dissolved oxygen balance is:

$$\frac{dC_{DO}(t)}{dt} = K_L a\left(C_{sat} - C_{DO}(t)\right) - q_{O_2} X(t) \;= OTR(t) - OUR(t) \tag{1}$$

Where X is the total cells concentration, C_{DO} is the dissolved oxygen concentration, C_{sat} the oxygen saturation in equilibrium with the oxygen partial pressure of the gaseous phase, $K_L a$ is the volumetric oxygen mass transfer rate, and q_{O_2} is the net specific oxygen uptake rate. The first term in the second member is the rate of aeration or OTR (oxygen transfer rate from air bubble to liquid phase), and the second term is the rate of oxygen consumption by cells or OUR (oxygen uptake rate of cells per volume of broth).

The aeration term (OTR) can be written as follows (Amicarelli et al., 2010):

$$OTR(t) = K_{air} F_{air}\left(C_{sat} - C_{DO}(t)\right) \tag{2}$$

where K_{air} is an oxygen consumption parameter by growth (constant for each fermentation), and F_{air} is the inlet air flow rate that enters the bio-reactor. For a given bioreactor configuration, F_{air} is mainly a function of the impeller agitation speed. Based on experimental evidence (Atehortúa et al, 2007), it is assumed that the oxygen consumption rate (OUR) depends on the total cells X, that is, both vegetative and sporulated cells consume oxygen at different rates, therefore:

$$\mathrm{OUR}(t) = q_{O_2} X(t) \tag{3}$$

Because dissolved oxygen is considered as a second substrate q_{O_2} is:

$$q_{O_2} = \frac{\mu_X}{Y_{X/O_2}} = \frac{\mu_X}{Y_{X/O_2}^{\max}} + m_{O_2} \tag{4}$$

Where Y_{X/O_2} the observed biomass yield is based on oxygen consumed, $Y_{X/O_2}^{\max}$ is the true biomass yield based on oxygen used for growth, m_{O_2} is the oxygen consumption coefficient for respiration maintenance, and μ_X is the net specific growth rate of total cells.

$$\mathrm{OUR}(t) = q_{O_2} X(t) = \underbrace{\frac{\mu_X}{Y_{X/O_2}^{\max}} X(t)}_{\substack{O_2 \text{ consumption} \\ \text{for biomass growth}}} + \underbrace{m_{O_2} X(t)}_{\substack{O_2 \text{ consumption} \\ \text{for cell maintenance}}} \tag{5}$$

With the aim to define μ_X , from the total cells balance:

$$X(t) = X_v(t) + X_s(t) \tag{6}$$

where X_v is the vegetative cells concentration, X_s is the sporulated cells concentration and the vegetative cell balance:

$$\frac{dX_v(t)}{dt} = \underbrace{\mu X_v(t)}_{\text{Growth rate of vegetative cell}} - \underbrace{k_d X_v(t)}_{\substack{\text{death rate of} \\ \text{vegetative cell}}} - \underbrace{k_s X_v(t)}_{\substack{\text{sporulation rate of} \\ \text{vegetative cell}}} \tag{7}$$

where μ is the gross specific growth rate of vegetative cells, k_d is the relative death rate of vegetative cells, k_s is a kinetic constant representing the spore formation rate, and μ_v is the net specific growth rate of vegetative cells.

$$\mu_v = \mu - k_d - k_s \tag{8}$$

Operating with (6) and (7):

$$\begin{aligned} \frac{dX(t)}{dt} &= \frac{dX_v(t)}{dt} + \frac{dX_s(t)}{dt} = (\mu - k_d) X_v(t) \\ &= (\mu - k_d) \underbrace{\frac{X_v(t)}{X(t)}}_{f_v} X(t) = (\mu - k_d) f_v X(t) = \mu_X X(t) \end{aligned} \tag{9}$$

where f_v is a fraction of vegetative cells, then:

$$\mu_X = \frac{1}{X(t)} \frac{dX(t)}{dt} = \frac{d[\log X(t)]}{dt} = (\mu - k_d) f_v \tag{10}$$

$$k_d^{(n)} = k_{d_{\max}}\left(\frac{1}{1+e^{G_d\,(t_0+nT_0-P_d)}}\right) - k_{d_{\max}}\left(\frac{1}{1+e^{G_d\,(t_0-P_d)}}\right) \tag{11}$$

As was stated before, dissolved oxygen C_{DO} is considered a second limited growth substrate, therefore μ is modeled using a double Monod kinetic expression (Ryder & Sinclair, 1972).

$$\mu = \mu(S, C_{DO}) = \mu_{\max}\left(\frac{S(t)}{(K_S + S(t))}\frac{C_{DO}(t)}{(K_O + C_{DO}(t))}\right), \quad 0 \le \mu \le \mu_{\max} \tag{12}$$

Now, replacing (10) in (5) $\mu_X X(t) = (\mu - k_d) X_v(t)$ and taking into account that $X(t) = X_v(t) + X_s(t)$, we obtain:

$$OUR(t) = \left[\frac{\mu - k_d}{Y_{X/O_2}^{\max}} + m_{O_2}\right] X_v(t) + m_{O_2} X_s(t) = q_{O_2}^v X_v(t) + q_{O_2}^s X_s(t)$$
$$q_{O_2}^v = \frac{\mu - k_d}{Y_{X/O_2}^{\max}} + m_{O_2}, \quad q_{O_2}^s = m_{O_2} \tag{13}$$

Where $q_{O_2}^v$, and $q_{O_2}^s$ are the specific oxygen uptake rate for vegetative and sporulated cells respectively. Equation (13) shows explicitly that both vegetative and sporulated cells consume oxygen at different rates. This is a similar model to the model of oxygen uptake rate proposed by Park et al. (p. 1023, Equation (A.9)), (Park et al., 2009) for endospore-forming bacteria.

We now analyse if the net specific oxygen uptake rate q_{O_2} does become negative.

Taking into account that $0 \le \mu \le \mu_{\max}$, $k_d \ge 0$, $Y_{X/O_2}^{\max} > 0$, $f_v \ge 0$, and $m_{O_2} \ge 0$. Comparing (5) and (13) it is obvious that $q_{O_2} \ge 0$ if always $q_{O_2}^v \ge 0$ because $q_{O_2}^s = m_{O_2}$ is positive. Then, we analyze if $q_{O_2}^v$ does become negative.

$$q_{O_2}^v = \frac{\mu - k_d}{Y_{X/O_2}^{\max}} + m_{O_2} \ge 0 \;\rightarrow\; k_d - Y_{X/O_2}^{\max} m_{O_2} \le \mu$$

For *Bacillus thuringiensis* $q_{O_2}^v$ is positive because the inequality $0 \le k_d - Y_{X/O_2}^{\max} m_{O_2} \le \mu \le \mu_{\max}$ always holds, therefore q_{O_2} is also positive.

$$\begin{cases} Y_{X/O_2}^{\max} = 263.15 \\ m_{O_2} = 0.00073 \\ k_d = 0.1 \end{cases} \Rightarrow \quad \mu \ge \underbrace{k_d}_{0.1} - \underbrace{Y_{X/O_2}^{\max} m_{O_2}}_{0.192} = -0.092$$

Alternatively, by replacing ($\mu_X X(t) = dX(t)/dt$) in (5), we obtain an equivalent model of oxygen uptake rate:

$$\text{OUR(t)} = \frac{1}{Y_{X/O_2}^{\max}}\frac{dX(t)}{dt} + m_{O_2} X(t) \tag{14}$$

This form of implementation of the oxygen consumption model has been widely reported in the literature; see the review by Garcia-Ochoa et al, 2010, p. 296, Equation (12) (Garcia-Ochoa et al., 2010), and references therein. This model form has also been specifically applied to modelling *Bacillus thuringiensis* oxygen uptake rate (Amicarelli et al., 2010; Ollis, 1983; Rowe et al., 2003).

The oxygen uptake rate models (5), (13), and (14) are mathematically equivalent. The model (13) was significantly more explanatory power than model (14), but from a computational point of view, the last one is better than the others. This is so, because the time-discretized model (15) is a linear difference equation with constant coefficients.

$$\text{OUR(t)} = \frac{1}{Y_{X/O_2}^{\max}}\frac{dX(t)}{dt} + m_{O_2} X(t) \;\xrightarrow[T_0]{}\; OUR^{(n)} = \left[\frac{1}{Y_{X/O_2}^{\max} T_0} + m_{O_2}\right] X^{(n)} - \frac{1}{Y_{X/O_2}^{\max} T_0} X^{(n-1)} \tag{15}$$

Consequently, the proposed model for Dissolved Oxygen Balance:

$$\frac{dC_{DO}(t)}{dt} = K_{air} F_{air}\left(C_{sat} - C_{DO}(t)\right) - \frac{1}{Y_{X/O_2}^{\max}}\frac{dX(t)}{dt} - m_{O_2} X(t) \tag{16}$$

3. VALIDATIONS AND EXPERIMENTATIONS

3.1 Data pre-processing

In this section, we will compare batch experiments of bacillus fermentations with the results achieved with the proposed model (16). The variables of the dissolved oxygen model are presented in Table 1 and the parameters values for *Bacillus thuringiensis* (*Bt*) are presented in Table 2. With the aim to generalize the model for others microorganism, there are tacked experimental data of batch fermentations of *Bacillus subtilles* (*Sb*) from the work of Park Park, Rittmann, and Bae (Park et al., 2009). The parameters values identified in this work for the dissolved oxygen model for *Sb* fermentations are presented in Table 3. The saturation concentration of oxygen (C_{sat}) is also a function of time, because the composition of the gas phase (in equilibrium with the liquid phase) varies with time. However, to simplify the model, the time dependency was not considered in the present formulation.

Table 1: Variables in the Dissolved Oxygen model

Symbol	Description
X	Total cell concentration $\left[g\ L^{-1}\right]$
C_{DO}	Dissolved oxygen concentration $\left[g\ L^{-1}\right]$
K_O	Oxygen saturation constant $\left[g\ oxygen\ L^{-1}\right]$
C_{sat}	Oxygen saturation concentration (DOC concentration in equilibrium with the oxygen partial pressure of the gaseous phase) $\left[g\ L^{-1}\right]$
Y_{X/O_2}	Observed biomass growth yield based on oxygen consumed $\left[g\,cells\ g\,oxygen^{-1}\right]$
$\frac{1}{Y_{X/O_2}^{\max}}$	Oxygen consumption constant by growth [dimensionless] $\left[g\,oxygen\ g\,cells^{-1}\right]$

K_{air}	Ventilation constant $\left[L^{-1}\right]$
$K_L a$	Volumetric oxygen mass transfer rate $\left[h^{-1}\right]$
q_{O_2}	Net specific oxygen uptake rate $\left[g\ oxygen\ \ g\ cells^{-1}h^{-1}\right]$
F_{air}	Inlet air flow rate that enters the bioreactor $\left[L\,h^{-1}\right]$
OTR	Oxygen transfer rate $\left[g\ oxygen\ \ L^{-1}h^{-1}\right]$
OUR	Oxygen uptake rate $\left[g\ oxygen\ \ L^{-1}h^{-1}\right]$
T_S	Sample time $\left[h\right]$
m_{O_2}	Oxygen consumption constant for maintenance [h^{-1}] $\left[g\ oxygen\ \ g\ cells^{-1}h^{-1}\right]$

Table 2: Parameters in the model for *Bt* fermentations

Symbol	Value
C_{sat}	0.00745
m_{O_2}	0.729 10^{-3}
$1/Y^{\max}_{X/O_2}$	3.795 10^{-3}
K_{air}	2.114 10^{-3}

Table 3: Parameters in the model for *Sb* fermentations

Symbol	Value
C_{sat}	0.0090
m_{O_2}	20.77 10^{-3}
$1/Y^{\max}_{X/O_2}$	6.2 10^{-3}
K_{air}	2.1

Agitation speed was set as high as possible to obtain better mass transfer between air bubbles and liquid. Obviously, a natural limit to this speed is the shear forces caused during liquid agitation; thus, the K_{air} value is obtained from DO experimental data at the nominal agitation speed (N= 400 rpm). The value of the air inlet flow was 1320 Lh^{-1}. The collected data from the B*t* fermentations is a set of measurements of biomass concentration and dissolved oxygen concentration. A sampling time Ts = 0.1 hours was selected by using Fourier frequency analysis. The biomass concentration data record must be increased to have the same experimental data record length as the dissolved oxygen concentration (approximately from 18 to 180 samples).

In this paper, Gaussian Process regression has been used as an imputation method for filling in the missing values (di Sciascio & Amicarelli, 2008), see Fig.1 and Fig.2 for *Bt* and *Sb* respectively. The collected data from the *Sb* fermentations is a set of biomass measurements and in the work of Park (Park et al., 2009) there are not reported experimental DO data in correspondence with the biomass concentration, despite of this, we compare the DO obtained by the model of Park and our model (see Figure 4).

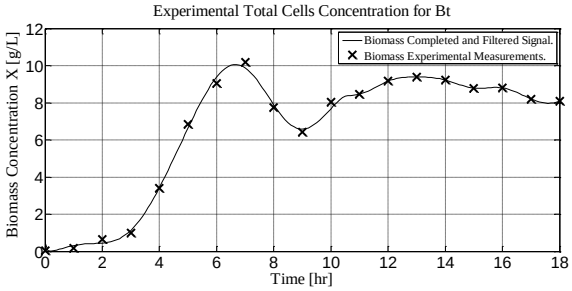


Fig 1. Experimental measurements of biomass concentration and completed filtered biomass for *Bacillus thuringiensis* fermentation.

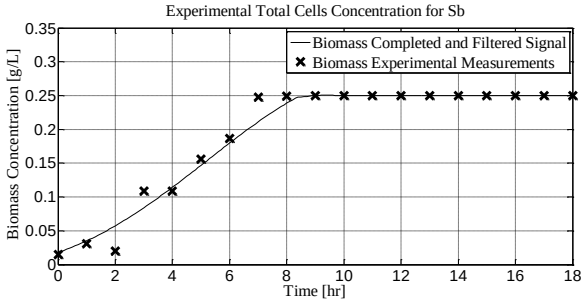


Fig 2. Experimental measurements of biomass concentration and completed filtered biomass for *Bacillus subtiles* fermentation (experimental data from (Park et al., 2009)).

3.2 Model Parameter Estimation for Sb fermentation.

The estimation of the dissolved oxygen model parameters m_{O_2} , $\frac{1}{Y^{\max}_{X/O_2}}$, and K_{air} for fermentations of *Bacillus thuringiensis* was developed in (Amicarelli et al., 2010). Now it is necessary to estimate the same parameters for *Bacillus Subtiles (Sb).* The proposed continuous DO dynamic model for a batch system is:

$$\frac{dC_{DO}(t)}{dt} = K_{air}F_{air}\left(C_{sat} - C_{DO}(t)\right) - \frac{1}{Y^{\max}_{X/O_2}}\frac{dX(t)}{dt} - m_{O_2}X(t) \tag{17}$$

The approximate discrete-time model of continuous-time dynamic model (17) is:

$$\frac{\left(C_{DO}(t_k) - C_{DO}(t_{k-1})\right)}{T_s} = -\frac{1}{Y^{\max}_{X/O_2}}\cdot\frac{\left(X(t_k) - X(t_{k-1})\right)}{T_s} - m_{O_2}\cdot X(t_k) + K_{air}\cdot F_{air\,in}(C_{sat} - C_{DO}(t_k)) \tag{18}$$

$$t_k = k\cdot T_s,\ 1\le k\le N,$$
$$C_{DO}(t_0) = C_{DO}(0),\ X(t_0) = X(0)$$

Operating algebraically with (18):

$$C_{sat}-C_{DO}(t_k)=\frac{1}{1+K_{air}\cdot F_{air\,in}\cdot T_s}\cdot\left(C_{sat}-C_{DO}(t_{k-1})\right)$$
$$+\frac{\frac{1}{Y^{\max}_{X/O_2}}}{1+K_{air}\cdot F_{air\,in}\cdot T_s}\cdot\left(X(t_k)-X(t_{k-1})\right)+\frac{m_{O_2}\cdot T_s}{1+K_3\cdot F_{air\,in}\cdot T_s}\cdot X(t_k) \quad (19)$$
$$t_k=k\cdot T_s,\ 1\le k\le N,$$
$$C_{DO}(t_0)=C_{DO}(0),\ X(t_0)=X(0)$$

Notice that (19) can be written compactly in the following form:

$$\hat{y}(t_k|\boldsymbol{\theta})=\theta_1\varphi_1(t_k)+\theta_2\varphi_2(t_k)+\theta_3\varphi_3(t_k)=\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(t_k) \quad (20)$$

This last equation (20) represents a discrete-time linearly parameterized model where:

$$\hat{y}(t_k|\boldsymbol{\theta})=C_{sat}-C_{DO}(t_k),$$
$$\boldsymbol{\varphi}(t_k)=\begin{bmatrix}\varphi_1(t_k)\\ \varphi_2(t_k)\\ \varphi_3(t_k)\end{bmatrix}=\begin{bmatrix}C_{sat}-C_{DO}(t_{k-1})\\ X(t_k)-X(t_{k-1})\\ X(t_k)\end{bmatrix},$$
$$\boldsymbol{\theta}=\begin{bmatrix}\theta_1\\ \theta_2\\ \theta_3\end{bmatrix}=\frac{1}{1+K_{air}\cdot F_{in}\cdot T_S}\cdot\begin{bmatrix}1\\ \frac{1}{Y^{\max}_{X/O_2}}\\ m_{O_2}\cdot T_S\end{bmatrix} \quad (21)$$

A column vector $\boldsymbol{\theta}$ featuring the unknown parameters are the parameters vector, whereas $\boldsymbol{\varphi}(t_k)$ is the regression vector formed by known signals. From a set of N input-output experimental data, sampled at equally-spaced time intervals Ts, it is possible to estimate the parameters vector $\boldsymbol{\theta}$ and, there from the physical parameters estimates $\frac{1}{Y^{\max}_{X/O_2}}$, m_{O_2}, and K_{air} (see (21)). Then, the estimated parameters of the model are obtained directly from (21):

$$\boldsymbol{K}=\begin{bmatrix}\frac{1}{Y^{\max}_{X/O_2}}\\ m_{O_2}\\ K_{air}\\ \ \end{bmatrix}=\begin{bmatrix}\frac{\theta_2}{\theta_1}\\ \frac{\theta_3}{\theta_1\cdot T_S}\\ \frac{1-\theta_1}{F_{air\,in}\cdot T_S\cdot\theta_1}\end{bmatrix} \quad (22)$$

3.3 Model Results

The results obtained with the proposed model for the *Bt* and *Sb* fermentations are shown in Fig. 3 and 4 respectively. Results for *Bt* fermentations were previously reported and discussed in (Amicarelli et al., 2010). However, an example of the behavior represented by the model in (16) for *Bacillus thuringiensis* can be seen in Fig. 1. In the case of *Bacillus subtiles*, we considered the experimental results reported by (Park et al, 2009). With this information, we obtained the DO output values with our proposed model and we finally compared these results with the ones reports by Park et al. in Fig. 4.

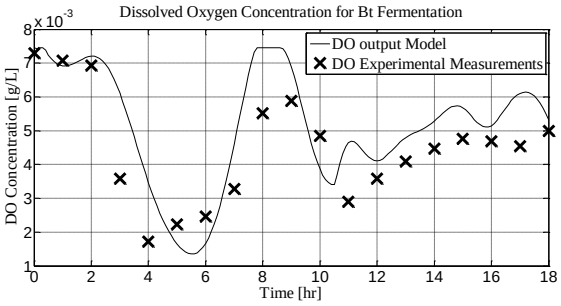


Fig. 3. DO Concentration Approximation Model for *Bacillus thuringiensis* fermentation.

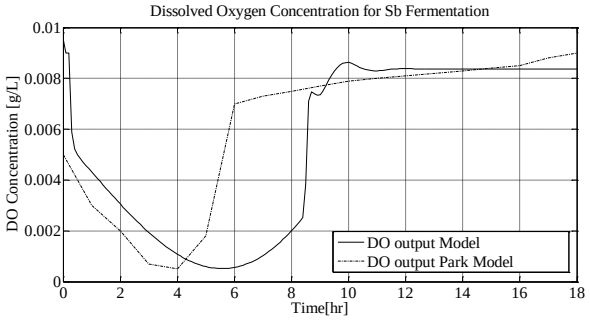


Fig. 4. DO Concentration Approximation Model for *Bacillus subtiles* fermentation.

With reference to Fig. 4, it can be observed that our DO model output adequately represent the dynamic behavior according to the considered biomass data. Oxygen consumption is evident during the first hours of fermentation (approximately 8 h) since the process is on the microorganism exponential growth stage. Next, the oxygen requirements decrease in the sporulation stage. Park's model predicts a DO level recovery two hours before (see Fig. 6). This is may be due to the affection of the others terms involved in such model. See, equation A.9 in (Park et al, 2009). It can be concluded that for practical aims and control purposes for the DO variable for this process. Our proposed model (16) satisfactory represents the behavior of *Bt* and *Sb* batch fermentations. Moreover, the model is based on the biomass values only. This is clear advantage when is compared with more complex models that require more variables, which are in many cases hard to obtain. Biomass data that can be sensed provided the specific sensors, o by a proper estimators design as proposed and validated in the previous papers: (di Sciascio & Amicarelli, 2008), (Adriana Amicarelli et al., 2014), (Amicarelli et al., 2015), (Rómoli et al., 2016).

4. CONCLUSIONS

This paper presented an extension or generalization of a dissolved oxygen dynamic model for batch fermentations. The generalization was performed from a particular previous model for *Bacillus thuringiensis* (Amicarelli et al., 2010) and was now extended for a more general class of aerobics microorganism named Aerobic Endospore-Forming Bacteria

(AEFB). This model generalization allows the scientific community to use the same model for control and state estimation purposes not only in *Bt* but also in a considerable variety of bioprocesses. The model was validated with experimental data of batch fermentations with *Bacillus Subtiles* and *Bacillus thuringiensis*. The model was also compared against more complex obtaining similar results, making easy its practical implementation for related applications.

ACKNOWLEDGMENT

The authors wish to thank the following organizations who contributed to the completion of this work: Universidad Nacional de San Juan (UNSJ, Argentina), and Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET, Argentina).

REFERENCES

Alzate, A., Amicarelli, A., Gómez, L., & di Sciascio, F. (2016). Métodos de Quasi-Monte Carlo para el Cálculo de Conjuntos en Teoría de Conjuntos en Control. In Argencon 2016. Buenos Aires Argentina: IEEE Argentina.

Amicarelli, A., di Sciascio, F., Toibero, J., & Alvarez, H. (2010). Including dissolved oxygen dynamics into the Bt δ-endotoxins production process model and its application to process control. Brazilian Journal of Chemical Engineering, 27, 41-62.

Amicarelli, A., Montoya, L. Q., & di Sciascio, F. (2015). Substrate Feeding Strategy Integrated with a Biomass Bayesian Estimator for a Biotechnological Process. International Journal of Chemical Reactor Engineering, Accepted for publication.

Amicarelli, A., Quintero, O., & Sciascio, F. (2014). Behavior comparison for biomass observers in batch processes. Asia-Pacific Journal of Chemical Engineering, 9, 81-92.

Amicarelli, A., Quintero, O., & Sciascio, F. (2014). Behavior comparison for biomass observers in batch processes. Asia-Pacific Journal of Chemical Engineering, 9, 81-92.

Atehortúa, P., Álvarez, H., & Orduz, S. (2006). Comments on: "A Sporulation Kinetic Model for Batch Growth of B. thuringiensis". The Canadian Journal of Chemical Engineering, 84, 386-388.

Atehortúa, P., Álvarez, H., & Orduz, S. (2007). Modeling of growth and sporulation of Bacillus thuringiensis in an intermittent fed batch culture with total cell retention. Bioprocess and Biosystems Engineering, 30, 447-456.

Cleveland, J., Montville, T. J., Nes, I. F., & Chikindas, M. L. (2001). Bacteriocins: safe, natural antimicrobials for food preservation. International journal of food microbiology, 71, 1-20.

Davis, D., Lynch, H., & Varley, J. (2001). The application of foaming for the recovery of surfactin from B. subtilis ATCC 21332 cultures. Enzyme and microbial technology, 28, 346-354.

di Sciascio, F., & Amicarelli, A. N. (2008). Biomass estimation in batch biotechnological processes by Bayesian Gaussian process regression. Computers & Chemical Engineering, 32, 3264-3273.

Errington, J. (2003). Regulation of endospore formation in Bacillus subtilis. Nature Reviews Microbiology, 1, 117-126.

Garcia-Ochoa, F., Gomez, E., Santos, V. E., & Merchuk, J. C. (2010). Oxygen uptake rate in microbial processes: an overview. Biochemical engineering journal, 49, 289-307.

Hoch, J. A. (1993). The phosphorelay signal transduction pathway in the initiation of Bacillus subtilis sporulation. Journal of cellular biochemistry, 51, 55-61.

Holzapfel, W., Geisen, R., & Schillinger, U. (1995). Biological preservation of foods with reference to protective cultures, bacteriocins and food-grade enzymes. International journal of food microbiology, 24, 343-362.

Ollis, D. F. (1983). A Simple Batch Fermentation Model: Theme and Variations. Annals of the New York Academy of Sciences, 413, 144-156.

Park, S., Rittmann, B. E., & Bae, W. (2009). Life-cycle kinetic model for endospore-forming bacteria, including germination and sporulation. Biotechnology and Bioengineering, 104, 1012-1024.

Rómoli, S., Amicarelli, A., Ortiz, O., Scaglia, G., & di Sciascio, F. (2016). Nonlinear Control of the Dissolved Oxygen Concentration Integrated With a Biomass Estimator for Production of Bacillus Thuringiensis δ-Endotoxins. Computer & Chemical Engineering, Accepted for publication.

Rowe, G. E., Margaritis, A., & Wei, N. (2003). Specific Oxygen Uptake Rate Variations during Batch Fermentation of Bacillus thuringiensis Subspecies kurstaki HD-1. Biotechnology progress, 19, 1439-1443.

Ryder, D., & Sinclair, C. (1972). Model for the growth of aerobic microorganisms under oxygen limiting conditions. Biotechnology and Bioengineering, 14, 787-798.

Sonenshein, A. L. (2000). Control of sporulation initiation in Bacillus subtilis. Current opinion in microbiology, 3, 561-566.

Stahmann, K.-P., Revuelta, J., & Seulberger, H. (2000). Three biotechnical processes using Ashbya gossypii, Candida famata, or Bacillus subtilis compete with chemical riboflavin production. Applied Microbiology and Biotechnology, 53, 509-516.

Todar, K. (2006). Todar's online textbook of bacteriology: University of Wisconsin-Madison Department of Bacteriology.

Yeh, M.-S., Wei, Y.-H., & Chang, J.-S. (2006). Bioreactor design for enhanced carrier-assisted surfactin production with Bacillus subtilis. Process Biochemistry, 41, 1799-1805.

Drive System Development for Gait Rehabilitation Exoskeleton

Sergey González-Mejía*, José Miguel Ramírez-Scarpetta**

** School of Electrical and Electronic Engineering, Universidad del Valle, Cali, Colombia (Tel: +57-2-4028239; e-mail: sergey.gonzalez@correounivalle.edu.co).*
*** School of Electrical and Electronic Engineering, Universidad del Valle, Cali, Colombia (e-mail: jose.ramirez@correounivalle.edu.co)*

Abstract: The research considers an assisted rehabilitation platform with a lower limbs exoskeleton, it has a crane to support the weight of patient and exoskeleton, and also, it includes a control system to synchronize the patient´s position on treadmill. This paper focuses on development of drive system for trajectory tracking control based on angular positions in a lower limb exoskeleton walking on a treadmill. It describes the logic states for servo-controllers algorithm and human-machine interface development to manage the drive system. It is presented experimental results under different operating conditions and performance analysis of the system.

Keywords: Assisted, drive, exoskeleton, gait, human-machine interface, joint, platform, rehabilitation, tracking, trajectory.

1. INTRODUCTION

The trajectory tracking control based on the joint angle position is important at early stage of rehabilitation, which can help the affected limb to achieve continuous and repetitive training. The main problem to address in the position control is the generation of an appropriate path that can be used for passive training (Meng et al. 2015). Robotic gait training is an emerging technique for retraining walking ability following spinal cord injury. A challenge is determining an appropriate gait trajectory and level of assistance for each patient, since patients have a wide range of sizes and impairment levels. In (Emken et al. 2008), it demonstrates how a lightweight and powerful robot can record subject-specific, trainer-induced leg trajectories during manually assisted gait, then replay those trajectories for an assisted therapy. Vallery et al. (2009) proposed a method for online trajectory generation that can be applied for hemiparetic patients; desired states for one disabled leg are generated online based on the movements of the other leg. An instantaneous mapping between legs is performed by exploiting physiological interjoint couplings. In this way, the patient generates the reference motion for the affected leg autonomously. The paper (Duschau-Wicke et al. 2010) presents a patient-cooperative strategy that allows patients to influence the timing of their leg movements along a physiologically meaningful path; the path control strategy, generates a compliant virtual that keeps the patient legs within a tunnel around the desired spatial path. In (Hussain et al. 2013), a trajectory tracking controller based on a chattering-free robust variable structure control law was implemented in joint space to guide the patient limbs on physiological gait trajectories; the performance of the robotic orthosis was evaluated during two gait training modes, namely, trajectory tracking mode with maximum compliance and trajectory tracking mode with minimum compliance.

In this research is considered a lower limbs exoskeleton (Ramírez-Scarpetta & Caicedo 2012); it is a rehabilitation orthosis for balancing and gait in patients with motor disabilities. The exoskeleton imposes torques in joints, respecting the motion ranges, in order to develop gait therapies (Ramírez-Scarpetta 2016).

This paper details the development of a tracking system based on angular positions in sagittal plane for the lower limbs exoskeleton. In second section the assisted rehabilitation platform description is presented; the third section displays the drive system development for trajectory tracking control and the last section the performance achieved by the tracking system is showed.

2. ASSISTED REHABILITATION PLATFORM DESCRIPTION

The Industrial Research Group, GICI, of Universidad del Valle developed an assisted rehabilitation platform for human gait, Fig. 1, which consists of three stages: the first stage is the user interface that it is installed on a PC and it is managed by therapist only, to manage the therapy, patient information and gait parameters; the outputs of this stage are the angular paths describing a natural gait, logic controls of the interface buttons and values of gait speed and step length; the second stage is the engineering level to manage the angular trajectories tracking; the information from the previous stage goes into a PC that has a human-machine interface installed, with a logic button panel for movements execution of gait and sends the trajectories towards a development system based on processor (National Instruments 2012); this system transmits the values of angular positions towards a drive system of exoskeleton by RS-485 communication. This orthosis is actuated by servo-motors (Harmonic Drive AG 2010c) in hips and knees joints,

driven by servo-controllers SC-610 (Harmonic Drive AG 2010a), ankles are not actuated. The position control of the joints is implemented in the servo-controllers, which is a cascade control structure for position, speed and current (González-Mejía & Ramírez-Scarpetta 2014); the weight of the exoskeleton and the patient is supported by a crane; the third stage is the control system for synchronizing position on a treadmill, which uses a kinect to know the patient position on the band.

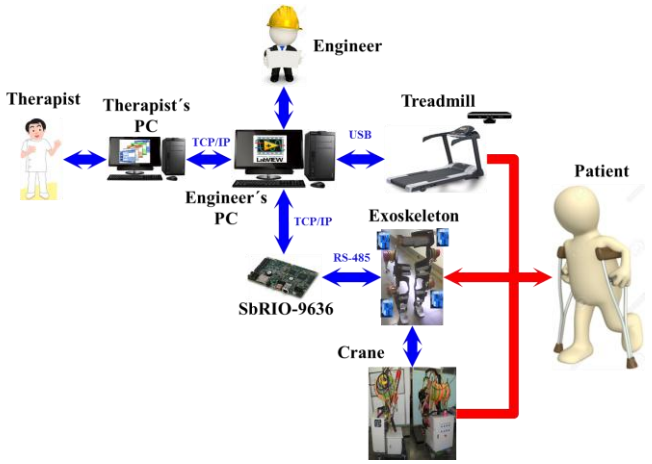


Fig. 1. Assisted rehabilitation platform for human gait.

This paper focuses on development of the drive system for trajectory tracking control based on angular positions in a lower limb exoskeleton carrying a mannequin and walking on a treadmill, Fig. 2.

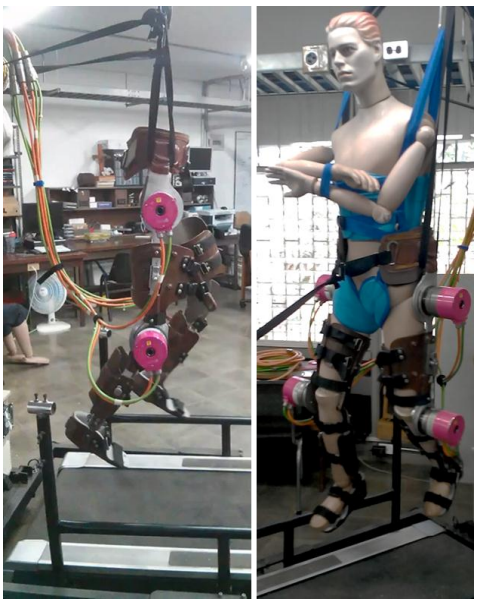


Fig. 2. Lower limbs exoskeleton on treadmill.

The Fig. 3 shows in general way, the drive system structure for rehabilitation platform; it details the interaction between the therapist PC, engineer PC, development system and the exoskeleton servo-controllers, also, the process of the data transmission.

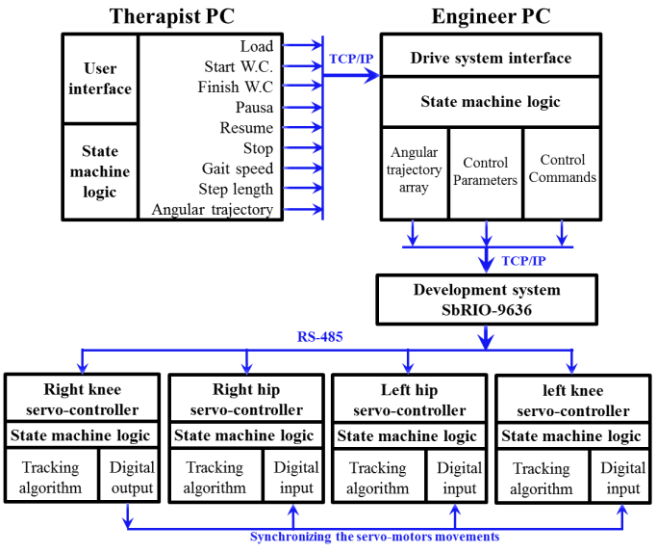


Fig. 3. Drive system structure for rehabilitation platform.

3. DRIVE SYSTEM DEVELOPMENT FOR TRAJECTORY TRACKING CONTROL

The research analyzes the results obtained in the master thesis (González-Mejía 2014), for selecting a path pattern that generates a natural gait in the sagittal plane; the selected database (Troje 2002), are displacement data (x, y, z) with an external reference frame, then the system origin is moved to the position of total mass center of system, COM, and then is applied the inverse kinematics algorithm to calculate the angles generated in lower limbs joints. In Fig. 4 it is shown that the origin of coordinate system has been positioned on the total COM of exoskeleton-patient. The purpose in the thesis (González-Mejía 2014), is the design and deployment of a control for static balance in exoskeleton of lower limbs on sagittal plane, which it is based on the concept of support area and the horizontal deflection of total COM of system. Further, a control strategy is designed to compensate the loss of balance when the system undergoes to large deviations.

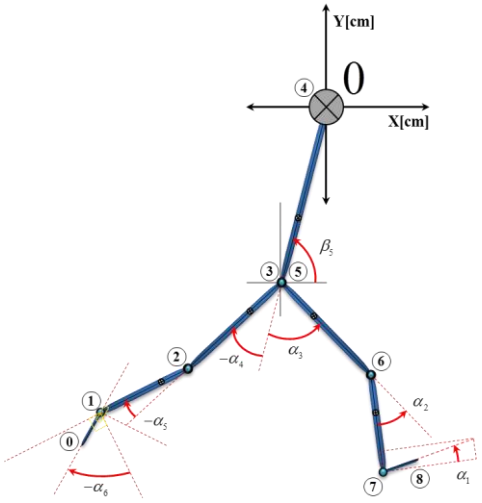


Fig. 4. Reference system in total COM .

The angular trajectories with origin on total COM are physically implementable in the tracking control system of servo-motors.

In the drive system development, a human-machine interface is created under the LabView® platform; the process of sending angular trajectories towards the drive system in the exoskeleton is managed. The angular trajectory assigned to each joint must be organized in an array of one hundred data. These arrays are decomposed and sent via RS-485 communication to the specified servo-controller, in order to rebuild the array by an algorithm programmed into the servo-controller.

In communication frame, three data are sent to the servo-controller: the angle value *Array[i]*, the position in the array to store the data, *i*, and the amount of data in the trajectory array, *length(Array)*. With this information into servo-controller, the algorithm can reconstruct and store in a new array the path that was sent. Then, the servo-controller of right knee is specified as master and the other slaves. The master role is to lead the synchronization of movement of the other servo-motors, since, the slaves servo-controllers begin to read the trajectories arrays and to apply the angles values when the master indicates.

In the routine for reading trajectories arrays, the program looks up the angle value in the array and replace it in Eq. (1), then it executes the movement, $Angulo_{motor}$, and moves to the next angle of the array. To this process, it is configured a runtime called sampling time or reading, t_m.

The movement of the servo-motor is given by Eq. (1).

$$Angulo_{motor}[i] = \left(AngularGain\left(Array[i] * \frac{100}{360} \right) \right) + Offset \quad (1)$$

The relationship between i and gait cycle time, t_{gait}, is given by Eq. (2).

$$t_{gait} = (t_m)i \quad (2)$$

AngularGain, *Offset* and sample time are control parameters that can be modified from interface and then are sent to the servo-controllers via RS-485 communication. The *AngularGain* parameter varies the step length, the Offset corrects the posture angle of the patient and sample time governs the cycle gait.

The time of cycle gait, t_{gait}, is Eq. (3).

$$t_{gait} = \frac{2L_p}{V_m} \quad (3)$$

Where, L_p is the step length and V_m is gait speed and in compliance with limits, t_m, it has Eq. (4).

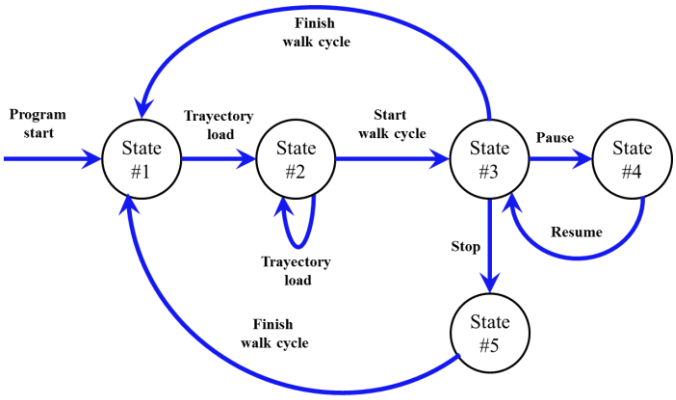
$$V_m \leq L_p \leq \left(\frac{t_{gait_{max}}}{t_{gait_{min}}} \right) V_m \quad (4)$$

The output variables of the therapist interface are the inputs of human-machine interface in engineering level; the output variables are the events caused by the logic of button panel, and the parameterized angular trajectories for natural gait, the gait speed and the step length, which lead to calculate the cycle time of gait, Eq. (3).

3.1 Programming of Servomotors' Controllers

The program into servo-controllers has a structure of state machine, Fig. 5.

The states are activated by the action on the button panel of engineering interface. Following, the states are described:



State #1: Software initialized.
State #2: Angular trajectory loaded.
State #3: Exoskeleton is walking.
State #4: Exoskeleton paused.
State #5: Exoskeleton stopped suddenly.

Fig. 5. State machine for servo-controllers program.

***Software initialized*:** The servo-controllers are turned on and the human-machine interface is running. It is loaded the operation settings of the servo-controllers, such as digital inputs and outputs, parameters velocity and acceleration, home position, etc. In this state, when servo-controllers are turned on, the angular position of joints is calibrated to zero and thus, servo-motors are positioned in an initial condition before turn. Also, it is allowed to activate the command *Trajectory load*.

Angular trajectory loaded: When the control command, *Trajectory load*, is executed, the angular trajectories are sent from the interface to the servo-controllers via RS-485. The arrays are reconstructed by servo-controller´s program and is waited the full send of trajectories. It is allowed to activate the commands *Trajectory load* and *Start walk cycle*.

Exoskeleton is walking: When executing the command, *Start walk cycle*, the exoskeleton performs walking cycles and only in this state, the user can send control parameters*: walking time, walk cycles, offsets* and *angular gains*.

To perform gait movements, the Fig. 6 shows the algorithm executed inside of servo-controllers.

The first three steps in algorithm, the servo-motors move to an initial position; this is necessary for smooth movement and ramp up without generating the *follow error* (Harmonic Drive AG 2010b) in the servo-controller, and avoid strong changes in the angular acceleration. This generates the gait initial step in the exoskeleton. It is allowed to activate the commands *Finish walk cycle*, *Pause* and *Stop*.

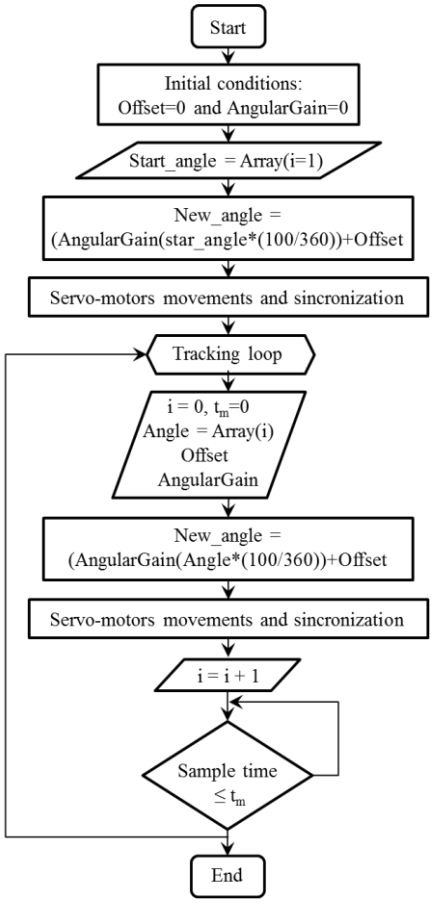


Fig. 6. Angular position tracking algorithm.

Exoskeleton paused: It is activated by a request, *Pause*, from the interface. The servo motors are positioned in the closest trajectory for double support phase Fig. 7. It is allowed to activate the command *Resume* to return to the state #3.

Exoskeleton stopped suddenly: When executing the *Stop* command, the exoskeleton is suddenly stopped by user decision due to an emergency. It is allowed to activate the command Finish walk cycle to return to the state #1, which generates a smooth ramp-down motion without generating an error in the servo-controller. This generates the gait final step in the exoskeleton.

Reading commands and data received by serial communication is performed through interruptions.

For the servo-motors synchronization a master-slave topology is used, where the master is the servo-controller of right knee and the others are slaves. The master handles a digital output and slaves handle a digital input, the connection is shown in Fig. 3.

The synchronization is done through a digital pin and its status is evaluated: the master and slave load angle value in a motion buffer (Harmonic Drive AG 2010b); then, the master generates a high state to be read by slaves and thus apply movement. In this way, it is achieved that the servo motors move synchronously due to management master.

Fig. 7. Exoskeleton in double support phase.

3.2 Human-Machine Interface Development

The human-machine interface is divided into three stages: The first stage sends the corresponding angular trajectory to each servo-motor, the second sends the control parameters to all servo-controllers and the third handles the control commands. Following, stages of the interface developed in the LabView® platform are explained, Fig. 8.

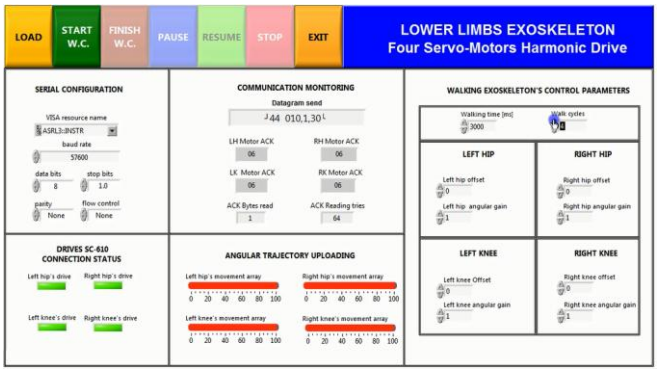


Fig. 8. Human machine interface.

The control parameters are on the right side of the interface; for each joint actuated is applied: *Offsets* and *angular gains* in angular positions, and in general way, it sets *walking time*

and *walk cycles* to generate the movements; the upper left side shows the control commands, which are under a logic button panel: *Load*, *Start walking cycle*, *walk Finish cycle*, *Pause*, *Resume*, *Stop* and *Exit*; the right side shows the settings parameters for serial communication; the lower right side and center shows the connectivity between the interface and the drive system, and indicators for transmit/receive data and evolution of loading for angular trajectories, respectively.

Sending angular trajectory array: The conception of general code implemented for sending the trajectory array for each servo-controller is shown in Fig. 3. Mainly, it takes a data in the array, the position of this data into array, the amount of array data and then those values are included into writing datagram of servo-controller; then the datagram is sent and it is expected to read an ACK character from the servo-controller. This algorithm is contained in three *While* cycles: the internal cycle re-transmits the datagram even if the algorithm cannot read the ACK; the middle one is for transmitting each array value of trajectory and the external is for perform array transmission for each servo-controller.

Sending control parameters: This stage is activated after sending all the angular trajectories and the same concept explained in the previous section is applied, with the difference that this code does not have the intermediate while loop and it sends the control parameters periodically.

Sending control commands: The management of these sending commands is given by the state machine programmed into servo-controllers.

4. TRACKING SYSTEM PERFORMANCE

The trajectory tracking system was tested by referring to the detailed database in the third section; it contains one hundred data for hip and knee joints, and with a sampling time of $15\,(ms)$. With different sampling times, the tracking trajectory of each servo-motor with the reference is compared. The Fig. 9 and Fig. 10 show that angular trajectories tracking without load by each servo-motor is successful.

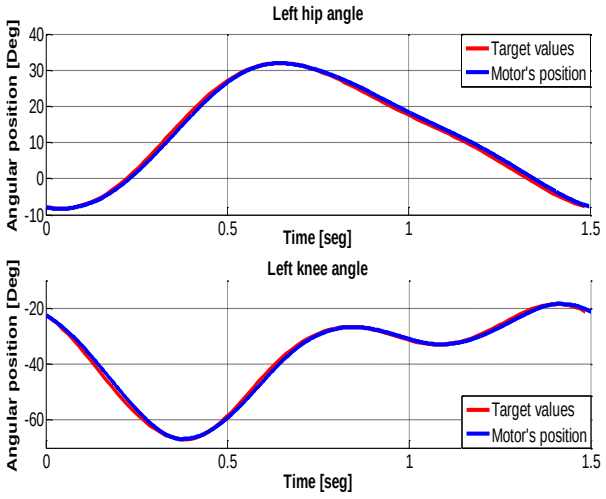


Fig. 9. Angular tracking trajectories of left side servo-motors.

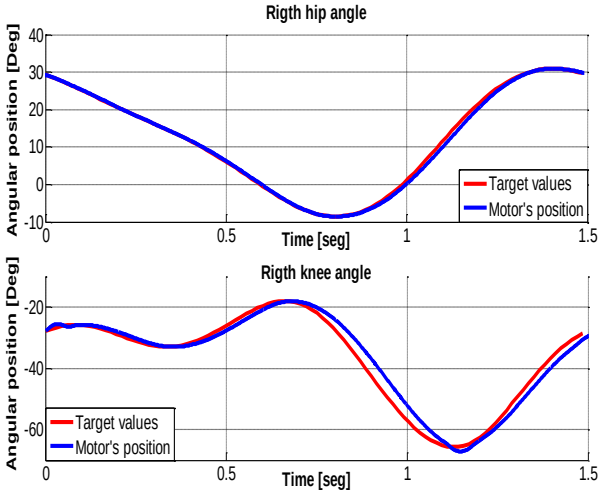


Fig. 10. Angular tracking trajectories of servo-motors right side.

Varying the sampling time, t_m, and operating the system without load, it achieved a walking time from $1(s)$ to $3.5(s)$, the results are shown in Fig. 11.

Furthermore, with the servo-motor assembly in knee joint of exoskeleton, it achieved a walking time from $2(s)$ to $5(s)$, the difference is due to the servo motor executes the last command given by the servo-controller even if it has not reached the desired position. Smooth and stepless movements are achieved, Fig. 12.

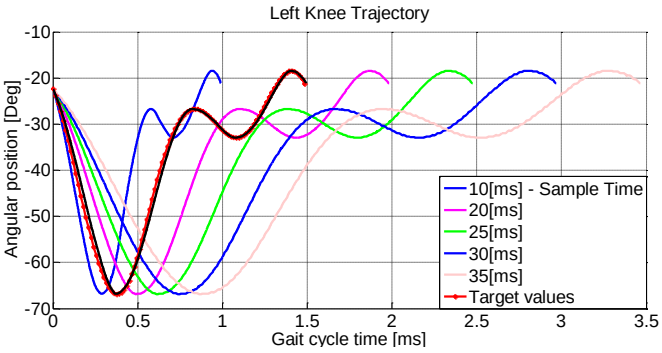


Fig. 11. Trajectory tracking of left knee servo-motor without load.

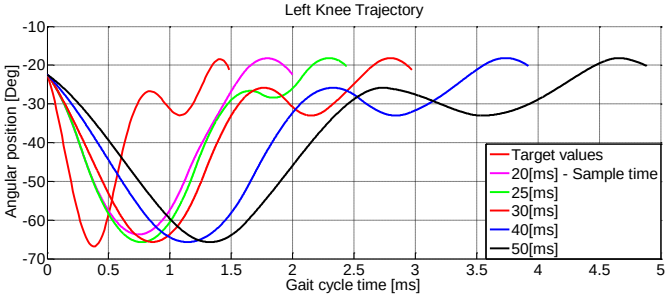


Fig. 12. Trajectory tracking of servo-motor of left knee with load.

5. CONCLUSIONS

The performance of drive system for trajectory tracking control based on angular positions in a lower limb exoskeleton was successful; the system fulfilled the required actions, timing, positioning, good tracking and a correct communication; the conception of logic states allowed to develop an appropriated human-machine interface to manage the drive system of rehabilitation platform; an algorithm for angular trajectories tracking was implemented in the servo-controllers; the master-slave topology developed generates synchronized and smooth movements in the gait of exoskeleton; the deployment of a deterministic sampling time in the platform, allowed to impose a correct cycle gait time.

For future works, an assist-as-needed control will be developed to measure the effectiveness of the patient rehabilitation.

6. ACKNOWLEDGEMENTS

This work is part of doctoral research of Sergey González, funded by Colciencias; he is on paid leave given by Universidad Santiago de Cali, and the research project: "Gait assisted with an exoskeleton", funded by Colciencias, Code 1106-521-28248, Contract 0462-2012.

7. REFERENCES

Duschau-Wicke, A. et al., 2010. Path Control: A Method for Patient-Cooperative Robot-Aided Gait Rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(1), pp.38–48.

Emken, J.L. et al., 2008. Feasibility of Manual Teach-and-Replay and Continuous Impedance Shaping for Robotic Locomotor Training Following Spinal Cord Injury. *IEEE Transactions on Biomedical Engineering*, 55(1), pp.322–334.

González-Mejía, S., 2014. *Diseño e implementación de una estrategia de control para equilibrio en posición bípeda de un exoesqueleto de miembros inferiores*. Master thesis, Universidad del Valle.

González-Mejía, S. & Ramírez-Scarpetta, J.M., 2014. Design and Implementation of a control strategy for static balance of a lower limbs exoskeleton. *Ingeniería y Competitividad*, 16(1), pp.229 – 238.

Harmonic Drive AG, 2010a. Product Documentation SC-610 Series Servo Controllers. Available at: http://www.harmonicdrive.de/english/products/servoproducts/servo-controller/sc-610/product-documentation.html.

Harmonic Drive AG, 2010b. Set-up-Software Work Bench v5 for SC-610-Series Servo Controllers. , pp.1 – 35. Available at: http://www.treffer.com.br/produtos/harmonicdrive/PDF/setup_software_sc_610.pdf [Accessed June 1, 2016].

Harmonic Drive AG, 2010c. The CHA Series digital AC hollow-shaft servo actuator. Available at: http://www.harmonicdrive.de/english/products/servoproducts/servoactuators/cha/ [Accessed February 8, 2014].

Hussain, S., Xie, S.Q. & Jamwal, P.K., 2013. Robust Nonlinear Control of an Intrinsically Compliant Robotic Gait Training Orthosis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(3), pp.655–665.

Meng, W. et al., 2015. Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation. *Mechatronics*.

National Instruments, 2012. sbRIO-9636 Support - National Instruments. Available at: http://sine.ni.com/psp/app/doc/p/id/psp-1029/lang/es [Accessed February 19, 2014].

Ramírez-Scarpetta, J.M., 2016. *Marcha asistida con un exoesqueleto*, Cali.

Ramírez-Scarpetta, J.M. & Caicedo, E., 2012. *Exoesqueleto para rehabilitación asistida de pacientes con pérdida parcial o completa del movimiento de los miembros inferiores. Fase 1: equilibrio*, Cali: Universidad Del Valle.

Troje, N.F., 2002. The little difference: Fourier based gender classification from biological motion. *Dynamic Perception R P Wrtz and M Lappe eds Aka Press Berlin*, pp.115–120.

Vallery, H. et al., 2009. Reference Trajectory Generation for Rehabilitation Robots: Complementary Limb Motion Estimation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(1), pp.23–30.

Dynamic Characterization of Typical Electrical Circuits via Structural Properties ⋆

Sofía Avila-Becerril * **Gerardo Espinosa-Pérez** *

* *Facultad de Ingeniería, Universidad Nacional Autónoma de México, Ciudad Universitaria, Ciudad de México (e-mail: soavbec@comunidad.unam.mx; gerardoe@unam.mx).*

Abstract: In this paper the characterization of a class of electrical circuits is carried out in terms of both stability properties and steady state behavior. The main contribution is the interpretation of the electrical topology in terms of mathematical properties derived from the structure of their models. In this sense, it is explained at what extent the topology by itself defines the dynamic behavior of the systems.

Keywords: Electrical circuits, Hamiltonian, Passivity, Graph Theory.

1. INTRODUCTION

The large number of studies on electrical circuits (Weiss and Mathis (1997), Maschke et al. (1995), Jeltsema and Scherpen (2003)) has enabled to recognize general and particular properties that in turn has allowed to generate new perspectives and methodologies for analysis and control. It is of interest to modify the dynamic behavior of the circuit so that some variables reach a desired operating point. The works that have been reported in this context, range from the characterization (see for example Van der Schaft and Maschke (2013)) to control (Ortega et al. (2003)) and applications (Jeltsema and Scherpen (2003)).

An interesting perspective on control circuits is based on the modification of its structure; this compensation means the addition of new elements like capacitors, inductors and resistors in specific locations such that the desired behavior of the electrical system is achieved. The methodology is attractive in correspondence with the study of intrinsic properties of the interconnection that can be applied for example in compensators of Electrical Power Systems. In this context, contributions have appeared where the aim is to exploit the structure of a given circuit. Van der Schaft (2010) studied the problem of shaping a resistive circuit behavior through the interconnection of another resistive circuit, viewing the second as a "controller" and leading to a methodology denoted as "partial synthesis by interconnection" which, in its turn, belong to a more general controller design called *Control by Interconnection* (CbI) (Ortega et al. (2008)). Roughly speaking, under this perspective the objective is to look at the controller as one dynamical system that interconnected with other (the plant) generates a new system with desired properties.

The situation described above, has encouraged the study of the structural properties of electrical circuits in order to recognize the dynamic behavior that can be achieved. The aim of this paper is to identify structural properties of the mathematical models of networks most used in practice, called typical circuits, and use them to characterized both their stability properties and their steady state behavior.

This approach is based on arguments of Graph theory (see Bollobás (1998)). First, the Kichhoff's laws are formulated in terms of basic cutsets and loopsets (for details see Wellstead (1979)). The stability analysis of these behaviors is developed going from simple (linear) to more complex (nonlinear) structures. The last part of the paper concerns to the characterization of the model components that influences the steady state behavior of typical topologies. These analyses are related with the operation of a Mesh network equipped with Direct Current (DC) or Alternate Current (AC) voltage sources. Some technical proofs and numerical testing are deferred to a journal article follow.

The rest of the paper is organized as follows: In Section 2 Kirchhoff's laws are stated in terms of basic cutsets and loopsets and it is presented the dynamic model of a class of electrical circuits followed by its stability analysis. The bases for the characterization of the steady state behavior of typical networks are included in Section 4 while the usefulness of the presented results is illustrated in Section 5.

2. ELECTRICAL CIRCUIT DYNAMIC

An electrical network can be defined as a oriented graph G consisting of a finite set of *nodes* $\mathcal{V}(G)$ and a finite subset $\mathcal{E}(G)$ of pairs of $\mathcal{V}$, called *edges*, where no self-loops are allowed. In electrical circuits nodes are the interconnection points of elements whereas the edges are associated to lumped one-port elements, with a voltage v *across* its terminals and a current i that flows *through* it.

Once the lumped elements are interconnected, their port variables must satisfy the Kirchhoff Current and Voltage Laws (KCL and KVL, respectively). A spanning tree (Bollobás (1998)) is a connected sub-graph containing the n nodes of the graph and $n-1$ edges such that no loops

⋆ Sponsor and financial support acknowledgment goes here.

are formed, the remaining $b-(n-1)$ edges form the corresponding co-tree. A *basic cutset* is a set of edges whose elements are one branch and some or all the chords. A *basic loopset* is a set formed by one chord and some or all branches such that a closed loop is formed. The KCL and the KVL are given in terms of the basic cutset matrix $C_b \in \mathbb{R}^{n-1\times b}$ and the basic loopset matrix $B_b \in \mathbb{R}^{b-n+1\times b}$ as

$$C_b i = 0 \ ; \ B_b v = 0 \tag{1}$$

And if $i \in \mathbb{R}^b$ and $v \in \mathbb{R}^b$ are ordered in such a way that

$$i = \begin{bmatrix} i_t \\ i_c \end{bmatrix} \in C_1; \quad v = \begin{bmatrix} v_t \\ v_c \end{bmatrix} \in C^1$$

with $i_t \in \mathbb{R}^{(n-1)}, v_t \in \mathbb{R}^{(n-1)}$ the currents and voltages of the tree and $i_c \in \mathbb{R}^{b-(n-1)}, v_c \in \mathbb{R}^{b-(n-1)}$ the currents and voltages of the co-tree, respectively, it is possible Bollobás (1998) to write down the current constraints as

$$i_t = -Hi_c, \quad v_c = H^T v_t. \tag{2}$$

where $H \in \mathbb{R}^{(n-1)\times b-(n-1)}$ is called the *fundamental loop matrix*.

In this paper it is considered a *complete* circuit (Brayton and Moser (1964)) so that it is not admitted capacitor-only loops and inductor-only cutsets. Let the circuit elements be grouped such that voltage sources, all the capacitors and some (voltage-controlled) resistors appear at the tree, while inductors and the rest (current-controlled) resistors are in the co-tree, [1] leading to

$$i_t = \begin{bmatrix} i_1 \\ i_C \\ i_{Rt} \end{bmatrix} ; \ v_c = \begin{bmatrix} v_{Rc} \\ v_L \end{bmatrix} ; \ v_t = \begin{bmatrix} v_1 \\ v_C \\ v_{Rt} \end{bmatrix} ; \ i_c = \begin{bmatrix} i_{Rc} \\ i_L \end{bmatrix}$$

where $v_1, i_1 \in \mathbb{R}^{n_1}$, $v_C, i_C \in \mathbb{R}^{n_2}$, $v_{Rt}, i_{Rt} \in \mathbb{R}^{n_3}$, such that $n_1+n_2+n_3 = n-1$, and $v_{Rc}, i_{Rc} \in \mathbb{R}^{n_4}$, $v_L, i_L \in \mathbb{R}^{n_5}$, with $n_4+n_5 = b-(n-1)$.

If the total stored energy of the circuit $H_a : \mathbb{R}^{n_2\times n_5} \to \mathbb{R}_{\geq 0}$ is defined as $H_a(q,\phi) = V_q(q) + V_\phi(\phi)$ and the port variables can be represented as

$$\dot{q} = i_C, \quad v_C = \frac{\partial H_a(q,\phi)}{\partial q} = \nabla_q H_a \tag{3a}$$

$$\dot{\phi} = v_L, \quad i_L = \frac{\partial H_a(i,\phi)}{\partial \phi} = \nabla_\phi H_a \tag{3b}$$

$$i_{Rt} = -f_t(v_{Rt}), \quad v_{Rc} = -f_c(i_{Rc}) \tag{3c}$$

where f_t and f_c are assumed to be bijective functions. With the partition introduced above the matrix H, in its turn, can be divided as

$$H = \begin{bmatrix} H_{1R} & H_{1L} \\ H_{CR} & H_{CL} \\ H_{RR} & H_{RL} \end{bmatrix} \tag{4}$$

where the subscript stand for the interconnections between tree and co–tree elements. As already reported in the literature, substitution of (3a–3c) into (2) leads to the dynamical model given by

$$\dot{x} = \mathbb{J}\nabla_x H_a(x) + \mathbb{F}(x, v_1, v_{Rt}, i_{Rc}) + \mathbb{G}E_1 \tag{5}$$

under the definitions

$$x = \begin{bmatrix} q \\ \phi \end{bmatrix} ; \ \nabla_x H_a(x) = \begin{bmatrix} \nabla_q H_a(x) \\ \nabla_\phi H_a(x) \end{bmatrix} ; \ E_1 = \begin{bmatrix} v_1 \\ 0 \end{bmatrix} \tag{6}$$

[1] For the sake of simplicity presentation current sources will be omitted of the analysis.

with matrices

$$\mathbb{J} = \begin{bmatrix} 0 & -H_{CL} \\ H_{CL}^T & 0 \end{bmatrix} ; \ \mathbb{G} = \begin{bmatrix} 0 & 0 \\ H_{1L}^T & 0 \end{bmatrix} \tag{7}$$

$$\mathbb{F}(x, v_1, v_{Rt}, i_{Rc}) = \begin{bmatrix} 0 & -H_{CR} \\ H_{RL}^T & 0 \end{bmatrix} \begin{bmatrix} v_{Rt} \\ i_{Rc} \end{bmatrix} \tag{8}$$

where

$$\begin{bmatrix} v_{Rt} \\ i_{Rc} \end{bmatrix} = \begin{bmatrix} -f_t^{-1}(-H_{RR}i_{Rc} - H_{RL}\nabla_\phi H_a(x)) \\ -f_c^{-1}(H_{1R}^T v_1 + H_{RR}^T v_{Rt} + H_{CR}^T \nabla_q H_a(x)) \end{bmatrix} \tag{9}$$

and complemented by the algebraic constraint

$$i_1 = -H_{1R}i_{Rc} - H_{1L}i_L \tag{10}$$

It is important to notice that if we concentrate on circuits with *linear* resistive elements, then

$$i_{Rt} = -R_t^{-1}v_{Rt}, \quad v_{Rc} = -R_c i_{Rc} \tag{11}$$

where $R_t = R_t^T > 0$ and $R_c = R_c^T > 0$ are diagonal matrices with the resistances of the tree and co-tree resistors, respectively.

Remark 1. The constraint (10) represents the current demanded to the voltage sources. Though, along the paper it is considered that there are ideal sources.

Remark 2. The time derivative of $H_a(x)$ along the trajectories of (5), considering $E_1 = 0$, is given by

$$\dot{H}_a(x) = \left(\nabla_x H_a(x)\right)^T \mathbb{F}(x, v_1, v_{Rt}, i_{Rc})$$

so that the stability of the network depends on the matrices H_{CR}, H_{RL}, H_{RR} and H_{1R}.

3. STRUCTURAL PROPERTIES FOR STABILITY

The steady state trajectories achievable by the system, denoted as *admissible* trajectories, are solution of

$$\dot{x}^\star = \mathbb{J}\nabla_{x^\star} H_a(x^\star) + \mathbb{F}(x^\star, v_1^\star, v_{Rt}^\star, i_{Rc}^\star) + \mathbb{G}E_1^\star \tag{12}$$

where it has been implicitly assumed the existence of an input $v_1^\star$ that generates the behavior $x^\star$. When $x^\star$ is an equilibrium point, $v_1^\star$ is constant and the steady state operation (equilibrium point) is input dependent and determined by

$$\mathbb{J}\nabla_{x^\star} H_a(x^\star) + \mathbb{F}(x^\star, v_1^\star, v_{Rt}^\star, i_{Rc}^\star) + \mathbb{G}E_1^\star = 0. \tag{13}$$

We first consider linear elements, so we identify conditions on H to guarantee *tracking* of a time-varying solution of (12) but the result applies only when the circuit is composed by linear elements, *i.e.*, when the total stored energy $H : \mathbb{R}^{n_2} \times \mathbb{R}^{n_5} \to \mathbb{R}_{>0}$ takes the form

$$H_a(x) = \frac{1}{2}x^T P x; \ \ P = diag\{C^{-1}, L^{-1}\} = P^T > 0 \tag{14}$$

with diagonal matrices $L \in \mathbb{R}^{n_5\times n_5} > 0$ and $C \in \mathbb{R}^{n_2\times n_2} > 0$ of inductances and capacitances. The resistors satisfy (11) and it is clear that

$$\nabla_x H_a(x) = Px.$$

Proposition 1. Consider a linear electrical network described by (5) and (14) with $v_1(t)$ a time-varying input such that its steady state behavior

$$\dot{x}^\star = [J-R]\,Px^\star + GE_1^\star \tag{15}$$

is well posed. Under these conditions

$$\lim_{t\to\infty} \tilde{x} = 0$$

with $\tilde{x} = x - x^\star$ if

$$ker\{H_{CR}^T\} = ker\{H_{RL}\} = 0. \tag{16}$$

Proof. The error dynamic is given by

$$\dot{\tilde{x}} = [J - R]\, P\tilde{x} + G\tilde{E}_1 \tag{17}$$

with the energy–like function

$$H_a(\tilde{x}) = \frac{1}{2}\tilde{x}^T P\tilde{x} \tag{18}$$

and the identities $\tilde{v}_C = \nabla_{\tilde{x}_1} H_a(\tilde{x}) = C^{-1}\tilde{q}$, $\tilde{i}_L = \nabla_{\tilde{x}_2} H_a(\tilde{x}) = L^{-1}\tilde{\phi}$. Take the function (18) as Lyapunov function candidate, its time derivative along (17) satisfies

$$\dot{H}_a = -zR_T^{-1}z \leq 0 \tag{19}$$

with $R_T = diag\{R_{11}, R_{22}\}$ and R_{11}, R_{22} symmetric positive definite matrices defined by

$$R_{11} = R_c + H_{RR}^T R_t H_{RR}$$
$$R_{22} = R_t^{-1} + H_{RR} R_c^{-1} H_{RR}^T$$

while

$$z = \begin{bmatrix} H_{CR}^T \tilde{v}_C \\ H_{RL}\tilde{i}_L \end{bmatrix} \tag{20}$$

The proof is completed by noting that the maximal invariant set where $\dot{\tilde{H}}_a = 0$ is $z = 0$. To guarantee that $\tilde{v}_C = \tilde{i}_L = 0$ are the only solutions, brings out $ker\{H_{CR}^T\} = ker\{H_{RL}\} = 0$ as the sufficient conditions that assure asymptotic stability of $(\tilde{x}, \tilde{v}_1) = (0, 0)$.

□

The proposition below shows that, in the case of equilibrium points, the same conditions still guarantee asymptotic stability for nonlinear capacitors and inductors.

Proposition 2. Consider an electrical network described by model (5) with $v_1^\star$ a constant input. In addition, assume that

A.1 The resistors involved in the circuit are characterized by linear constitutive relationships.

A.2 The equilibrium point $x^\star$ that correspond to $v_1^\star$ locally satisfies $x^\star = argmin\{H_a(x)\}$.

Then, the equilibrium point $(x^\star, v_1^\star)$ is locally asymptotically stable if

$$ker\{H_{CR}^T\} = ker\{H_{RL}\} = 0. \tag{21}$$

Proof. The total stored energy $H_a(x)$ is a nonlinear function. Therefore, under the assumption **A.1** the equilibria is characterized by the solutions of

$$[J - R]\, \nabla_{x^\star} H_a(x^\star) + GE_1^\star = 0$$

If assumption **A.2** holds, then, it is possible to consider the Lyapunov function candidate $H_0 : \mathbb{R}^{n_2+n_5} \to \mathbb{R}_{\geq 0}$ reported by Jayawardhana et al. (2007)

$$H_0(x) = H_a(x) - x^T \nabla_{x^\star} H_a(x^\star) - (H_a(x^\star) - x^{\star T} \nabla_{x^\star} H_a(x^\star)) \tag{22}$$

whose time derivative along the trajectories of (5), under **A.1** and $v_1 = v_1^\star$, yields

$$\dot{H}_0(x) = -z^T R_T^{-1} z.$$

□

Remark 3. The result above considers just the stability of equilibrium points. The tracking problem, up to the authors knowledge, imposes an open problem.

Motivated by Proposition 1 and Proposition 2, the final result of this section focuses in a particular class of circuits characterized by two properties.

Property 1. *If the number of tree resistors is equal to the number of inductors and they are one to one series connected then*

$$H_{RL} = I_2 \in \mathbb{R}^{n_3 \times n_3};\;\; H_{RR} = 0_1 \in \mathbb{R}^{n_3 \times n_4} \tag{23}$$

with I_2 as already defined and 0_1 a zero matrix.

Property 2. *If the number of co–tree resistors is equal to the number of capacitors and they are one to one parallel connected then*

$$H_{CR} = I_3 \in \mathbb{R}^{n_2 \times n_2};\;\; H_{1R} = 0_2 \in \mathbb{R}^{n_1 \times n_2} \tag{24}$$

with I_3 as already defined and 0_2 a zero matrix.

Remark 4. Typical networks exhibit these properties (see Avila-Becerril et al. (2015)). But is also found when modeling of inductors and capacitors losses are considered.

The dynamical behavior of the electrical network (5) under Properties 1 and 2 is described by

$$\dot{x} = \mathbb{J}\nabla_x H_a(x) + \mathbb{F}_1(x) + \mathbb{G}E_1 \tag{25}$$

where

$$\mathbb{F}_1(x) = \begin{bmatrix} -f_c^{-1}(\nabla_q H_a(x)) \\ -f_t^{-1}(\nabla_\phi H_a(x)) \end{bmatrix}$$

Under Properties 1 and 2 the dissipation terms only depend on the state; this allows to relax the assumptions about $f_c(\cdot)$ and $f_t(\cdot)$.

Proposition 3. Consider the electrical network described by (25) with $v_1^\star$ a constant input such that the steady state behavior is well posed. Assume **A.2** holds and that

A.3 The maps $f_c(\cdot)$ and $f_t(\cdot)$ define incremental output strictly passive operators in the sense that

$$(x_1 - x_2)^T \left[f_c^{-1}(x_1) - f_c^{-1}(x_2)\right] > 0$$
$$(x_1 - x_2)^T \left[f_t^{-1}(x_1) - f_t^{-1}(x_2)\right] > 0$$

hold for $x_1 \neq x_2$ and considering x_i as input.

Under these conditions the equilibrium point $(x^\star, v_1^\star)$ is locally asymptotically stable.

Proof. The equilibria of the system is characterized by

$$\mathbb{J}\nabla_{x^\star} H_a(x^\star) + \mathbb{F}_1(x^\star) + \mathbb{G}E_1^\star = 0$$

Thus, if **A.2** holds it is possible to consider the Lyapunov function candidate $H_0 : \mathbb{R}^{n_2+n_5} \to \mathbb{R}_{\geq 0}$, defined in (22), which leads, under the condition $v_1 = v_1^\star$, to

$$\dot{H}_0(x) = -(\nabla_x H_a(x) - \nabla_{x^\star} H_a(x^\star))^T \left[\mathbb{F}_1(x) - \mathbb{F}_1(x^\star)\right]$$

Using (3a) and (3b) and since **A.3** holds it is clear that $\dot{H}_0(x) < 0$ with the maximal invariance set defined as

$$\varepsilon = \{(q, \phi)\,|\,(v_C - v_C^\star) = 0, (i_L - i_L^\star) = 0\}. \tag{26}$$

□

4. TOPOLOGICAL STRUCTURE OF TYPICAL NETWORKS

In this section it is stated the structure of the matrix H for a generic network that captures, in an unified way, the characteristics of typical networks (For details on typical networks see the work of Fernández-Carrillo et al. (2015)). The generic network satisfies Properties 1 and 2, hence, its dynamic behavior is represented by model (25) leaving $H_{CL} \in \mathbb{R}^{n_2 \times n_5}$ and $H_{1L} \in \mathbb{R}^{n_1 \times n_5}$ to be characterized. Consider that the n_5 inductors are divided into three types, namely: n_r r-inductors, that

belong to a trajectory that connects a source with a capacitor, n_s s-inductors that belong to a trajectory that connects a source with another source and n_p p-inductors that connects a capacitor with another capacitor, such that $n_r + n_s + n_p = n_5$. Moreover, assume that the capacitors belong at least to one of the following classes

C.1 The $i-th$ capacitor, $i \in \{1, ..., n_2\}$, shares cutset with r_i r-inductors.

C.2 The $i-th$ capacitor, $i \in \{1, ..., n_2\}$, shares cutset with p_i p-inductors.

while the voltages sources satisfies that

C.3 The $i-th$ voltage source, $i \in \{1, ..., n_1\}$, shares cutset with $m_i \in \{1, ..., n_5\}$ inductors.

So that, the rows of matrix H_{CL} can be divided into: i) capacitors that hold simultaneously conditions **C.1** and **C.2** and ii) capacitors that hold only with condition **C.2**. The columns of H_{CL} are divided into three blocks each one corresponding to r, s and p inductors, respectively.

Following the stated organization, matrix H_{CL} takes the form

$$-H_{CL} = \begin{bmatrix} \mathbf{1}_{r_1}^T & 0 & \cdots & 0 & 0_s & N_1 \\ 0 & \mathbf{1}_{r_2}^T & \cdots & 0 & 0_s & N_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \mathbf{1}_{r_z}^T & 0_s & N_z \\ 0 & 0 & \cdots & 0 & 0_s & N_{z+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0_s & N_{n_2} \end{bmatrix}; \qquad (27)$$

where

- $\mathbf{1}_{r_i}^T \in \mathbb{R}^{1\times r_i}$, $i = 1, \ldots, z$, are vectors filled with ones denoting the condition stated in **C.1**. In this case it has been assumed that there exist z capacitors of this kind. It holds that $r_1 + r_2 + \cdots + r_z = n_r$.
- The zero columns $0_s \in \mathbb{R}^{1\times n_s}$ reflects the fact that any capacitor can be connected to s type inductors.
- Row vectors $N_i \in \mathbb{R}^{1\times n_p}$ include the possibility that a given capacitor can be simultaneously connected to r and p type inductors. If the $i-th$ capacitor is connected to one of the p-inductors, appears a 1, otherwise is 0. If the capacitor only hold **C.1**, $N_i = 0$.
- In the rows that go from $z+1$ to n_2 only appear vectors N_i since they correspond to **C.2** class capacitors.

Property 3. *Each column of the matrix* $N = col\{N_1, \ldots, N_{n_2}\} \in \mathbb{R}^{n_2\times n_p}$ *is composed by one* 1, *one* -1 *and the rest of the entries equal to zero.*

Property 4. *The vector* $\mathbf{1}_{n_2} \in \mathbb{R}^{n_2}$, *i.e., the vector filled with ones of dimension* n_2, *is a left eigenvector of matrix* N *satisfying* $\mathbf{1}_{n_2}^T N = 0$.

Concerning the structure of matrix $H_{1L} \in \mathbb{R}^{n_1\times n_5}$, it is obtained from (10), which is given by

$$i_1 = -H_{1L} i_L$$

taking into account that $H_{1R} = 0$. Their columns are divided into three blocks corresponding to r, s and p type inductors, respectively, where the third one is zero due to the fact that sources do not belong to cutsets where p-inductors are involved. This matrix takes the form

$$H_{1L} = [\, M_r \;\; M_s \;\; 0_p \,] \qquad (28)$$

- The entries different from zero of each row of $M_r \in \mathbb{R}^{n_1\times n_r}$ shows the connection of sources with r-inductors.
- The $i-th$ row of M_r, $i = 1, 2, \ldots, n_1$ is divided into z sections leading to

$$\beta_i = [\, \beta_{ir_1} \;\; \beta_{ir_2} \;\; \cdots \;\; \beta_{ir_z} \,] \qquad (29)$$

  where each $\beta_{ir_j} \in \mathbb{R}^{1\times r_j}$, $j = 1, 2, \ldots, z$, has only one entry equal to 1 if the $i-th$ source is connected to the $j-th$ capacitor. Otherwise, the vector is zero.
- Since two sources can not be connected to the same r-inductor, each column of M_r has only one entry different from zero.
- The sum of the entries different from zero of the $i-th$ row of M_r equals ρ_i.
- The entries different from zero of each column of $M_s \in \mathbb{R}^{n_1\times n_s}$ stand for the connection of sources with other source.
- The sum of the entries different from zero of the $i-th$ row of M_s equals γ_i. Therefore, $\rho_i + \gamma_i = m_i$ of **C.3**.
- Matrix $0_p \in \mathbb{R}^{n_1\times n_p}$ is a zero matrix that exhibits the fact that sources can not be related with p-inductors.

Property 5. *The vector* $\mathbf{1}_{n_1} \in \mathbb{R}^{n_1}$ *is a right eigenvector of matrix* M_s^T *satisfying* $M_s^T \mathbf{1}_{n_1} = 0$.

This general network can be specialized to the typical circuits, for example the Mesh Network.

Mesh Network In this topology each load is connected to all sources, then $r_i = n_1$ for all $i \in \{1, 2, \ldots, r\}$. Also, the number of sources equals the number of capacitors, then $n_1 = n_2$. Finally, there are not neither p-inductors nor **C.2** class capacitors.

The matrices that topologically characterize a Mesh network are

$$H_{CL} = \begin{bmatrix} -\mathbf{1}_{n_1}^T & 0 & \cdots & 0 & 0_s \\ 0 & -\mathbf{1}_{n_1}^T & \cdots & 0 & 0_s \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & -\mathbf{1}_{n_1}^T & 0_s \end{bmatrix} \in \mathbb{R}^{n_1\times n_5}, \qquad (30)$$

where n_1 is the number of sources and $n_s < n_1$ is the number of sources connected to other source, while

$$H_{1L} = [\, M_r \;\; M_s \,] \qquad (31)$$

with the particular feature that all the partitions β_{ir_j} of the $i-th$ row of M_r include an element different from zero since all the sources are connected to all capacitors.

5. STEADY–STATE CHARACTERIZATION: TWO CASE STUDY

The aim of this section is to exploit the structure of the Fundamental Loop Matrix H under two different scenarios.

5.1 Lossless DC network

Consider a network operating under constant voltage sources, with possibly nonlinear capacitors and inductors, with linear co–tree resistances and assuming that the tree resistances $R_t = 0$. This scenario is frequently considered in practice, for example, in Electrical Power Systems under the assumption that the lines are dominantly inductive.

Properties 1 and 2 hold. Hence, model (5) reduces to

$$\dot{x} = \begin{bmatrix} R_c^{-1} & -H_{CL} \\ H_{CL}^T & 0 \end{bmatrix} \nabla_x H_a(x) + \begin{bmatrix} 0 \\ H_{1L}^T \end{bmatrix} v_1 \tag{32}$$

For stability properties of the network, Proposition 2 can be directly applied, since

$$\dot{H}_0(x) = -(v_C - v_C^\star)^T R_c^{-1}(v_C - v_C^\star) \leq 0$$

and by direct substitution of $v_C^\star$ in (32), the maximal invariant set of the system corresponds to $i_L = i_L^\star$, condition that proves that asymptotic stability is attained.

The second part is devoted to the characterization of the steady state behavior defined by the equilibria of the system, which are the solutions of

$$-R_c^{-1} v_C^\star - H_{CL} i_L^\star = 0 \tag{33a}$$

$$H_{CL}^T v_C^\star + H_{1L}^T v_1^\star = 0, \tag{33b}$$

for a given $v_1^\star$, the characterization can be carried out for the capacitor voltages or the inductor currents. If the capacitor voltages are chosen, it is possible to write that

$$v_C^\star = -\left[H_{CL} H_{CL}^T\right]^{-1} H_{CL} H_{1L}^T e_1^\star \tag{34}$$

since, from Property 3, H_{CL} is row full rank.

Departing from this last expression, in the following proposition it is illustrated how exploiting the structures for H_{CL} and H_{1L} it is straightforward to concluded the steady state operation that is achieved by a given network. The result is illustrated for the case of a Mesh circuit.

Proposition 4. Consider a Mesh electrical circuit characterized by (30) and (31). Assume that

- The network is lossless, i.e., $R_t = 0$.
- Propositions 1 and 2 hold.
- The vector of voltage sources v_1 is composed by n_1 constant values.

Under these conditions capacitor voltages achieves average consensus (Bai et al. (2011)) in the sense that

$$v_C^\star = \alpha \mathbf{1}_{n_1}. \tag{35}$$

with

$$\alpha = \frac{1}{n_1} \sum_{m=1}^{n_1} v_{1m}^\star$$

the steady state average value of v_1.

Remark 5. A direct corollary of the last proposition refers to the case when $v_1^\star = \bar{v}_1 \mathbf{1}_{n_1}$, with $\bar{v}_1 \in \mathbb{R}$. Under this condition voltage capacitor consensus (see Bai et al. (2011)) is achieved in the sense that $v_C^\star = v_1^\star$.

Remark 6. Interestingly enough, if inductive loses are included, $R_t \neq 0$, consensus is not longer preserved. Instead of, the entries of the matrix $R_c^{-1} + H_{CL} R_t^{-1} H_{CL}^T$ depends on the values of the tree resistors. However, it seems that this situation allows for designing compensation techniques, *i.e.*, adding new lumped elements to the circuit, such that the desired behavior is accomplished.

5.2 AC steady–state behavior

In this section, we study an electric network operating under sinusoidal voltage sources. In this case all the passive elements are considered linear although inductive loses are included. Hence, the circuit dynamic is described by (5) under (14) and is represented as

$$P^{-1}\dot{z} = [J - R]\, z + G e_1$$

where $z = \left[\, v_C^T \;\; i_L^T \,\right]^T$ and J, R and G previously defined.

Since the steady state behavior is now time–varying, the admissible trajectories are given as solution of

$$P^{-1}\dot{z}^\star = [J - R]\, z^\star + G v_1^\star \tag{36}$$

As usual (Desoer and Kuh (1969)), it is assumed that voltage and currents are of the form

$$f(t) = F cos(\omega t + \phi) = Re(\mathbb{F} e^{j\omega t})$$

with the phasor $\mathbb{F} = F e^{j\phi}$. Therefore, admissible trajectories are defined by

$$Re\left[j\omega P \mathbb{Z}^\star e^{j\omega t} - (J - R)\mathbb{Z}^\star e^{j\omega t}\right] = Re(G)(\mathbb{V}_1^\star e^{j\omega t})$$

leading to

$$\mathbb{Z}^\star = \left[j\omega P - (J - R)\right]^{-1} G \mathbb{V}_1^\star \tag{37}$$

Considering that the circuit satisfies Properties 1 and 2, *i.e.*, identities (23) and (24) hold, the model reduces to

$$sP - (J - R) = \begin{bmatrix} sC + R_c^{-1} & H_{CL} \\ -H_{CL}^T & sL + R_t \end{bmatrix}$$

where $s = j\omega$, $C \in \mathbb{R}^{n_2 \times n_2}$ is the capacitance matrix and $L \in \mathbb{R}^{n_5 \times n_5}$ is the inductance matrix. Hence, it is obtained that

$$\mathbb{Z}^\star = \begin{bmatrix} \mathbb{V}_C^\star \\ \mathbb{I}_L^\star \end{bmatrix} = \begin{bmatrix} A_1 H_{1L}^T \mathbb{V}_1^\star \\ A_2 H_{1L}^T \mathbb{V}_1^\star \end{bmatrix} \tag{38}$$

with

$$A_1 = -\left[Y_{CR} + H_{CL} Y_{LR} H_{CL}^T\right]^{-1} H_{CL} Y_{LR} \tag{39}$$

$$A_2 = Y_{LR} - Y_{LR} H_{CL}^T \left[Y_{CR} + H_{CL} Y_{LR} H_{CL}^T\right]^{-1} H_{CL} Y_{LR}$$

and diagonal admitance matrices

$$Y_{CR} = sC + R_c^{-1} \tag{40}$$

$$Y_{LR} = (sL + R_t)^{-1} = s^{-1}(L + s^{-1} R_t)^{-1} = s^{-1} D \tag{41}$$

For illustrative purposes, the steady state behavior of capacitor voltages is characterized assuming a Mesh network. In accordance with the partitions of H_{CL} and H_{1L} previously introduced, the inductance takes the form

$$L = diag\{L_r, L_s\}$$

with $L_r \in \mathbb{R}^{n_r \times n_r}$, $L_s \in \mathbb{R}^{n_s \times n_s}$. In addition, the former must be divided into z matrices as

$$L_r = diag\{L_{ri}\}; \;\; i = 1, 2, \ldots, z$$

where $L_{ri} \in \mathbb{R}^{n_1 \times n_1}$ since for a Mesh network $n_r = n_1$.

From Properties 1 and 2, the dimension of R_c is $n_1 \times n_1$, due to the fact that $n_1 = n_2$, while $R_t \in \mathbb{R}^{n_5 \times n_5}$, *i.e.*, equals the dimension of L.

Under the aforementioned partitions, matrix D introduced in (41) is given by

$$D = diag\{D_r, D_s\} \tag{42}$$

where $D_r = diag\{D_{ri}\}$ with $D_{ri} = (L_{ri} + s^{-1} R_{ti})^{-1}$, $i = 1, 2, \ldots, z$, and R_{ti} submatrices of R_t of dimension corresponding to L_{ri}. Concerning matrix H_{1L}, whose structure is presented in (31), the sub–matrix $M_r \in \mathbb{R}^{n_1 \times n_r}$ is represented as

$$M_r = \left[\, M_{r1} \;\; M_{r2} \;\; \cdots \;\; M_{rz} \,\right]$$

with $M_{ri} \in \mathbb{R}^{n_1 \times r_i}$, $i = 1, 2, \ldots, z$, matrices composed by the vectors introduced in (29) exhibiting only one entry different from zero on each of their columns.

Proposition 5. Consider a Mesh electrical circuit characterized by (30) and (31). Assume that

- Propositions 1 and 2 hold.
- The vector of voltage sources v_1 is composed by n_1 sinusoidal functions.

Under these conditions the input/output relationship between the capacitor voltage phasors $\mathbb{V}_C^\star$ and the voltage source phasors $\mathbb{V}_1^\star$ is given by

$$\mathbb{V}_C^\star = -\mathbb{M}^{-1} \begin{bmatrix} \mathbf{1}_{n_1}^T D_{r1} M_{r1}^T \\ \mathbf{1}_{n_1}^T D_{r2} M_{r2}^T \\ \vdots \\ \mathbf{1}_{n_1}^T D_{rr} M_{rz}^T \end{bmatrix} \mathbb{V}_1^\star \tag{43}$$

where $\mathbb{M} = \left[s^2 C + s R_c^{-1} + diag\left\{ \mathbf{1}_{n_1}^T D_{ri} \mathbf{1}_{n_1} \right\} \right]$, $i = 1, 2, \ldots, z$.

Remark 7. The idea to manipulate the capacitance, inductance and resistance values to attain a given steady state behavior is not new. In many applications this procedure is recognized as *compensation* and is related with the addition of new elements parallel or series connected with the originals. The advantage offered by the approach presented in this paper, is that the compensation analysis can be carried out in a systematic way.

6. CONCLUDING REMARKS

In this paper a dynamic characterization of a class of electrical circuits has been presented. The characterization contemplates both stability properties and steady behavior; its main feature is that it is based on the structural properties of the networks which have been obtained by using arguments from the Graph theory. It has been shown that the structure of these circuits strongly defines their dynamic behavior. The usefulness of the results reported in this paper lies in the possibility to carry out the characterization in a very systematic way. In addition, it has been shown that they offer an alternative to deal with problems like compensator location to attain a prescribed behavior.

ACKNOWLEDGEMENTS

Part of this work was supported by DGAPA-UNAM under grant IN116516.

REFERENCES

Avila-Becerril, S., Espinosa-Pérez, G., and Fernández-Carrillo, P. (2015). Stability and consensus of electrical circuits via structural properties. *IFAC-PapersOnLine*, 48(13), 111–116.

Bai, H., Arcak, M., and Wen, J. (2011). *Cooperative control design: a systematic, passivity-based approach.* Springer Science & Business Media.

Bollobás, B. (1998). *Modern graph theory*, volume 184. Springer Science & Business Media.

Brayton, R. and Moser, J. (1964). A theory of nonlinear networks i. *Quart. Appl. Math*, 22(1), 1–33.

Desoer, C. and Kuh, E. (1969). *Basic Circuit Theory.* McGraw-Hill.

Fernández-Carrillo, P., Avila-Becerril, S., and Espinosa-Pérez, G. (2015). Consenso en sistemas eléctricos de potencia mediante compensación. *Congreso Nacional de Control Automtico (AMCA) 2015*, 527 – 532.

Jayawardhana, B., Ortega, R., García-Canseco, E., and Castanos, F. (2007). Passivity of nonlinear incremental systems: Application to pi stabilization of nonlinear rlc circuits. *Systems & control letters*, 56(9), 618–622.

Jeltsema, D. and Scherpen, J. (2003). A dual relation between port-hamiltonian systems and the brayton–moser equations for nonlinear switched rlc circuits. *Automatica*, 39(6), 969–979.

Maschke, B.M., Van der Schaft, A.J., and Breedveld, P.C. (1995). An intrinsic hamiltonian formulation of the dynamics of lc-circuits. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 42(2), 73–82.

Ortega, R., Jeltsema, D., and Scherpen, J. (2003). Power shaping: A new paradigm for stabilization of nonlinear rlc circuits. *Automatic Control, IEEE Transactions on*, 48(10), 1762–1767.

Ortega, R., van der Schaft, A., Castanos, F., and Astolfi, A. (2008). Control by interconnection and standard passivity-based control of port-hamiltonian systems. *Automatic Control, IEEE Transactions on*, 53(11), 2527–2542.

Van der Schaft, A. (2010). Characterization and partial synthesis of the behavior of resistive circuits at their terminals. *Systems & Control Letters*, 59(7), 423–428.

Van der Schaft, A. and Maschke, B. (2013). Port-hamiltonian systems on graphs. *SIAM Journal on Control and Optimization*, 51(2), 906–937.

Weiss, L. and Mathis, W. (1997). A hamiltonian formulation for complete nonlinear rlc-networks. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 44(9), 843–846.

Wellstead, P. (1979). *Introduction to physical system modelling.* Academic Press London.

Dynamic clustering for process supervision ⋆

Nathalie A. Barbosa [*,**] **Louise Travé-Massuyès** [**] **Victor H. Grisales** [***]

[*] *Universidad Nacional de Colombia, Department of Electrical and Electronics Engineering, Bogotá, Colombia (e-mail: nabarbosa@unal.edu.co)*
[**] *LAAS-CNRS, Université de Toulouse, CNRS, UPS, Toulouse, France. (e-mail: {nabarbos,louise}@laas.fr)*
[***] *Universidad Nacional de Colombia, Department of Mechanical and Mechatronics Engineering, Bogotá, Colombia (e-mail: vhgrisalesp@unal.edu.co)*

Abstract: In the context of data-driven monitoring, evolving environments challenge researchers with non-stationary data flows where the concepts (or states) being tracked can change over time. This issue is common in real industrial environments that suffer wear over time, implying for instance that the "normal" state undergoes drift. This requires monitoring algorithms suited to represent the evolution of the system behavior and in real industrial environments, also suited to represent time dependent features. This paper proposes a unified clustering approach to monitoring evolving environments using a two-stages distance-based and density-based algorithm. In this approach the system measurements trends, found on the fly, are characterized and then used as input to the clustering algorithm that provides as output clusters representing the system current state. Due to a forgetting process, clusters may emerge, drift, merge, split or disappear, hence following the evolution of the system. The dynamic clustering algorithm shows good outlier rejection capability when tested on an industrial benchmark suffering faults with varying magnitude.

Keywords: Dynamic clustering, Fault Detection, Supervision, Evolving environments

1. INTRODUCTION

Technological advances of past decades have resulted in systems highly adaptive to a constantly changing environment. In addition, they have changed the way enterprises get information about the state of their systems. Huge amounts of data, arising from various sources, are generally collected and they are available for further analysis. These two facts have promoted the success of machine learning approaches for diagnostics tasks, although they must face new challenges, in particular building efficient classification algorithms that adapt the targeted model – or classifier – to the evolution of the system and that scale to big data.

Many researchers have used pattern recognition, neural networks and clustering techniques for fault diagnosis (Maurya et al., 2010; Hedjazi et al., 2010; Eduardo Mendel et al., 2011). To carry out diagnosis using a pattern recognition method, there are two principal stages: training (or learning) and recognition. One known disadvantage of classic data-driven techniques is that they often address only anticipated fault conditions (Vachtsevanos et al., 2007), missing new or unknown data of which it was not aware of during training. Other disadvantage is that training examples usually follow static distributions that remain unchanged over time. If the performance of the algorithm decreases below a given threshold it should be re-learned, but it does not adapt dynamically (its structure and sometimes even its parameters).

The classification techniques can establish a model of the system functional states by extracting knowledge from various attributes. This knowledge is related to a particular behavior, without being represented by a set of analytic relations. The modifications of these characteristics enable the detection of abnormal operations (Isaza et al., 2009).

In the context when new objects submitted to the classifier during the recognition stage do not imply novelty detection nor a change of the classifier the classification is called *static classification* ((Joentgen et al., 1999)). If the classifier changes in time (dynamically), following the system, the classification task becomes a *dynamic classification problem*.

Dynamism in the classifier is achieved when not only the parameters but the classifier structure changes according to input data in an automatic way. Abrupt changes in the data can be captured by cluster creation or elimination. Smooth changes are usually reflected as cluster drifts and less frequently as cluster merging and splitting.

Among the techniques that have been used for dynamic classification, we can mention: evolving clustering ((Angelov, 2011)), Self-Adaptive feed-forward neural network (SAFN) ((Li et al., 2011)), LAMDA (Learning Algorithm for Multivariate Data Analysis) ((Kempowsky et al., 2006)), Growing Gaussian Mixture Models (2G2M) ((Bouchachia and Vanaret, 2011)), CluStream (Aggarwal et al., 2003), ClusTree (Kranen et al., 2011) and DenStream (Cao et al., 2006). Some of these alternatives

⋆ This work was supported in part by the Colombian administrative department of science, technology and innovation COLCIENCIAS, the Universidad Nacional de Colombia and the Laboratory for Analysis and Architecture of Systems LAAS-CNRS

are really complex and hence not suited to handle online large amounts of process data, such as data arriving in a stream. Their requirements in terms of memory and processor power are too high. Two-stages clustering (online/offline) has emerged as an alternative to deal with large amounts of data arriving at fast rates. Examples of this approach can be found in (Aggarwal et al., 2003; Kranen et al., 2011; Cao et al., 2006).

Qualitative trends (QT) is a user friendly representation of features that have been successfully applied in the fields of process monitoring and fault diagnosis (Dash et al., 2004; Maurya et al., 2007, 2010; Gamero et al., 2014). This non model-based technique exploits historical data of a process to characterize its behavior using a qualitative language. This representation reduces the complexity of system states by allowing only a finite set of descriptors (Gamero et al., 2014). Polynomial fit-based methods have been used to extract time-series trends owing to its shorter computational time and higher robustness to noise. The main advantages of the QT representation are its interpretability, the complexity reduction and the robustness in presence of noise. On the contrary, one main drawback is the lack of differentiation of episodes following the same qualitative trend, i.e. the concept of magnitude is completely lost.

In (Barbosa et al., 2015) we proposed a monitoring algorithm which couples a dynamic clustering method with an on-line trend extraction algorithm that works incrementally on the incoming data. The algorithm was used to achieve on-line diagnosis on the continuous stirred tank heater (CSTH) model developed by (Thornhill et al., 2008). In this paper we improve our previous work by: (1) introducing a global and local density analyses that improves the system tracking specially in presence of novel unknown behavior. (2) Introducing a variable representation that allows quick detection by an operator. (3) Proposing several functions to represent the forgetting process. This **Dy**namic **Cl**ustering algorithm for tracking **E**volving **E**nvironments is called ***DyClee***. We especially want to address the outlier rejection and the evolution characterization capabilities of our algorithm both in a local and global sense.

This paper is organized as follows: Section 2 introduces our algorithm including the dual global and local density analysis. Section 3 presents ***DyClee*** forgetting process including the new proposed functions. Section 4 presents outlier detection and its application to detect unknown system behaviors. Section 5 shows some capabilities of our algorithm in toy examples and Section 6 shows the same capabilities in the CSTH benchmark. Finally the conclusions are presented in section 7

2. DYNAMIC CLUSTERING ALGORITHM

This paper uses the distance- and density-based clustering approach introduced in (Barbosa et al., 2015) improved to be able to detect local and global outliers and to follow different evolution dynamics. The algorithm description is shown in figure 1.

System data is considered to arrive in stream. Data streams take the form of time series providing the values of the signals measured on a given process at each sampled time. In order to extract trend information, this work proposes to process each time series x_i into *episodes*, to generate an abstraction of the original signal into a simpler *qualitative-like*, yet quantitative, representation.

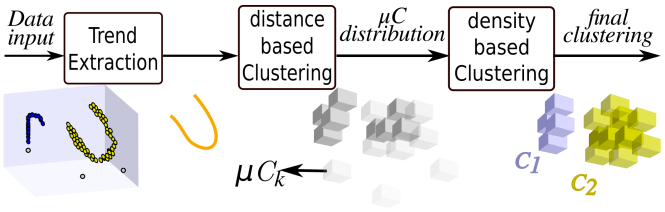


Fig. 1. Principle of ***DyClee***

Episodes are defined by three elements: a trend context TC, a set of auxiliary variables AV and a time interval T_i leading to (1):

$$e(x_i) = \{TC, AV, T_i\} \tag{1}$$

As introduced before, to find the trend context polynomial fit can be used. Instead of using an entirely qualitative representation using an alphabet of primitives like Maurya et al. (2007) or Gamero et al. (2014), we use the polynomial coefficients as trend context.

When data arrives, the preprocessing stage performs a polynomial fit in order to find the underlying trend. If the polynomial representation is considered as good, i.e. the polynomial fitting error is lesser than the signal noise variance, the polynomial coefficients are used as TC and the polynomial start and end time-stamps define T_i.

The distance-based clustering stage creates μ-clusters that are summarized representations of the data set made by using some statistical and temporal information. Formally, a μ-cluster is a hyper box representing a group of data points **close in all dimensions** and whose information is summarized in the tuple:

$$\mu C_k = (n_k, LS_k, SS_k, t_{lk}, t_{sk}, D_k, Class_k) \tag{2}$$

where n_k is the number of objects in the μ-cluster k, $LS_k \in \Re^d$ is the vector containing the linear sum of each feature over the n_k objects, $SS_k \in \Re^d$ is the square sum of feature over the n_k objects, t_{lk} is the time when the last object was assigned to that μ-cluster, t_{sk} is the time when the μ-cluster was created, D_k is the μ-cluster density and $Class_k$ is the μ-cluster label if known. In order to maintain an up-to-date structure allowing to track system evolution, μ-clusters are weighted with a forgetting function.

The density-based stage analyses the distribution of those μ-clusters whose density is considered as medium or high and creates the final clusters by a density based approach, that is, dense μ-clusters that are close enough (connected) are said to belong to the same cluster. A μ-cluster is qualified as one of three options: dense μ-cluster ($D\mu$C), semi-dense μ-cluster ($S\mu$C) or low density (outlier) $O\mu$-cluster ($O\mu$C).

In our previous work the dense character of a μ-cluster was found using a user specified parameter named α. Specifically, being K the total number of μ-clusters, $D\mu$Cs are the μ-clusters with $D_j \geq \alpha\ avg\,(D_1 \cdots D_K)$, $S\mu$C are the μ-clusters with $D_j \geq \frac{\alpha}{2} avg\,(D_1 \cdots D_K)$, and $O\mu$C are the μ-clusters with densities lower than that. The problem with this approach is that it demands user knowledge about the density distribution of the clusters and it does not guarantee cluster homogeneity.

In this paper we implements two different automatic approaches to establish the dense character of the μ-clusters, named global-density approach and local-density approach. The former approach allows to detect clusters with similar

densities while the later allows the detection of clusters with varied densities. ***DyClee*** global- and local-density approaches are further explained in the following subsections. These approaches are the first contribution of this paper.

2.1 Global-density analysis

In the global-density approach density is considered as a μ-cluster characteristic regarding all the μ-clusters. In this sense, two measures are considered as representative of the μ-clusters in a global sense, the average of μ-cluster's density and the median. These measures will work as thresholds for establishing the dense character of a μ-cluster. The intuition behind the selection of these measures is that the median and average densities of an heterogeneous group are significantly different, although, if the group is uniformly dense, these two quantities are equal.

Formally, a μ-cluster μC_z is said to be dense at time t if its density is greater than or equal to both global measures, i.e. the median and the average. On the contrary if its density is bigger or equal to one of the two measures and lower than the other the μ-cluster is said to be semi-dense. Finally if the μ-cluster μC_k have a density bellow both thresholds it is said to be an OμC. These conditions are represented in inequalities 3 to 5, where D_i is the density of the μ-cluster i and K is the total number of μ-clusters.

$$D\mu C \Leftrightarrow D_z \geq median\left(D_1 \cdots D_K\right) \wedge D_z \geq avg\left(D_1 \cdots D_K\right) \quad (3)$$

$$S\mu C \Leftrightarrow D_z \geq median\left(D_1 \cdots D_K\right) \vee D_z \geq avg\left(D_1 \cdots D_K\right) \quad (4)$$

$$O\mu C \Leftrightarrow D_z < median\left(D_1 \cdots D_K\right) \wedge D_z < avg\left(D_1 \cdots D_K\right) \quad (5)$$

As stated before, in order to find the final clusters, the dense character of the μ-clusters and its connections are analyzed. A set of connected μ-clusters is said to be a *group*. Groups of μ-clusters are analyzed recursively in order to find the clusters within. A cluster is created if every inside μ-cluster of the group is a $D\mu$C and every border μ-cluster is either a $D\mu$C or an $S\mu$C.

2.2 Local-density analysis

Density-based clustering algorithms as those from (Ester et al., 1996), proposing the DBSCAN algorithm, and (Barbosa et al., 2015), in which we proposed an initial version of ***DyClee***, group data samples according to density. Nevertheless, in these implementations, the concept of 'dense' is related to a global value ($MinPts$ in the case of (Ester et al., 1996) and α in the case of (Barbosa et al., 2015)). The problem of taking a global value to identify a point as dense appears when clusters with varied densities are present in the same data set as can be seen in Figure 2, where density is represented as μ-cluster opacity. In this case, density-based algorithms using a global approach may misclassify low density clusters as noise.

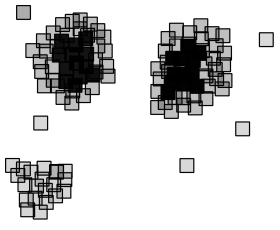


Fig. 2. μ-cluster groups of varied densities in 2D

Unlike the previously named approaches, ***DyClee*** analyses the dense character of each μ-cluster regarding the density of the other μ-clusters in the same group. This approach allows what is called multi-density clustering (Mitra et al., 2003).

As in the global approach the average and the median are chosen as thresholds, but they are applied recursively in the μ-clusters groups. In other words, for each group G_k, the μ-clusters having their density higher than or equal to the average density of the group ($avg(D_{G_k})$) and higher than or equal to the median density of the group ($median(D_{G_k})$) are considered as dense. μ-clusters having a density higher than or equal to only one of those measures (either average or median) are considered as $S\mu$Cs and those with density below both measures are considered as $O\mu$Cs. Summarizing:

$$D\mu C \Leftrightarrow D_z \geq median\left(D_{G_k}\right) \wedge D_z \geq avg\left(D_{G_k}\right), \quad (6)$$

$$S\mu C \Leftrightarrow D_z \geq median\left(D_{G_k}\right) \vee D_z \geq avg\left(D_{G_k}\right), \quad (7)$$

$$O\mu C \Leftrightarrow D_z < median\left(D_{G_k}\right) \wedge D_z < avg\left(D_{G_k}\right). \quad (8)$$

The μ-clusters group shown in Figure 2 is analyzed in this manner: First, the groups of μ-clusters are found and then, for each group, a cluster is formed with the denser connected μ-clusters as shown at the left of Figure 3. Once formed, the rest of the group is analyzed looking for the denser μ-clusters (with respect to the remaining elements of the group). If no dense region is found the next group in analyzed following the same method until all groups are analyzed. The final classification is shown at the right of Figure 3. The group or groups with the lower density are taken as outliers. It is worth noting that global-density approaches are unable to detect the green cluster since the densities of the μ-clusters in that group are low, with respect to the others clusters.

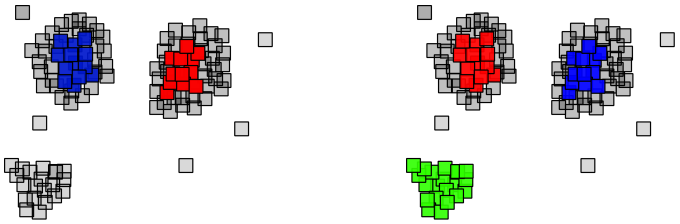


Fig. 3. Global-(left) and local-(right) density analysis results

3. TRACKING SYSTEM EVOLUTION

DyClee implements a forgetting process in order to cope with cluster evolution. Specifically, μ-clusters are weighted with a decay function dependent on the current time t and the last assignation time t_{lk}. This function $f\left(t, t_{lk}\right)$ emulates a forgetting process. When a new d-dimensional object $E_x = \left[x^1, \ldots, x^d\right]^T$ is assigned to a μ-cluster μC_k at t, the cluster last assignation time is updated to $t_{lk} = t$. The other attributes of the feature vector are updated as follows:

$$n_k^{(t)} = n_k^{(t-1)} f\left(t, t_{lk}\right) + 1 \quad \forall k \quad (9)$$

$$LS_{k,i}^{(t)} = LS_{k,i}^{(t-1)} f\left(t, t_{lk}\right) + x_i \quad \forall i, i = 1, \ldots, d. \quad (10)$$

$$SS_{k,i}^{(t)} = SS_{k,i}^{(t-1)} f\left(t, t_{lk}\right) + x_i^2 \quad \forall i, i = 1, \ldots, d. \quad (11)$$

Numerous machine learning methods have implemented some kind of forgetting function (also called decay function) to be able to detect or track concepts that drift or shift over time. As

introduced before in this work we propose several functions that may be used in the representation of the forgetting process.

The simplest forgetting function corresponds to a linear decay as given in equation (12). The function slope m could be inversely proportional to the time it takes to the function to go from one to zero, $m = 1/t_{w=0}$. This function with $t_{w=0} = 6000$ is plotted in blue in Figure 4. A linear decay has been used above all in biological and physical systems.

$$f(t, t_{lk}) = \begin{cases} 1 - m(t - t_{lk}) & t - t_{lk} \leq t_{w=0} \\ 0 & t - t_{lk} > t_{w=0} \end{cases} \quad (12)$$

If a non-forgetting time range is appended to a linear decay, we get a trapezoidal decay profile. This type of function is used as profile in electronic applications. The trapezoidal decay function is given in equation (13) where t_a represents the no forgetting time, and $t_{w=0}$ the time when the function reaches zero. This function is represented in red in Figure 4.

$$f(t, t_{lk}) = \begin{cases} 1 & t - t_{lk} \leq t_a \\ \dfrac{m - t}{m - t_a} & t_a \leq t - t_{lk} \leq t_{w=0} \\ 0 & t - t_{lk} > t_{w=0} \end{cases} \quad (13)$$

Statistical processes use overall an exponential decay function. This function is shown in equation (14), where λ_d is a positive rate known as exponential decay constant. The function is plotted in magenta in Figure 4. This type of functions is the most widely used to model decay since it has applications in all fields of science. A generalization of exponential decay is shown in equation (15). It is the decay function used by (Aggarwal et al., 2003) and (Kranen et al., 2011) and a graphical representation can be found in green in Figure 4. The change in the base from e to any value β gives interesting properties. For example, if β is chosen to be $\beta = 2\psi$, then the time at which half of the data is forgotten, is $\frac{1}{\psi\lambda_d}$. This function is known as the half life function and is widely used in biological processes.

$$f(t, t_{lk}) = e^{-\lambda_d(t - t_{lk})} \quad (14)$$

$$f(t, t_{lk}) = \beta^{-\lambda_d(t - t_{lk})} \quad (15)$$

Figure 4 shows the shape for the named functions in the case where $t_{w=0} = 6000$, $t_a = 2000$. For simplicity in the figure t_{lk} is set to zero. As previously mentioned, ***DyClee*** implements all the functions shown in equations (12) to (15), allowing proper adaptation to all kind of process evolutions. In ***DyClee*** the forgetting process impacts clusters density. As explained in the previous section, each t_{global} period the density of all μ-clusters is recalculated. Density change implies μ-cluster type change, its decrease makes $D\mu$-clusters become $S\mu$-clusters and $S\mu$-clusters become $O\mu$-clusters, and vice-versa.

4. OUTLIER DETECTION AND NOVELTY DETECTION

Outlier detection has been a widely investigated problem in several disciplines, including statistics, data mining and machine learning and several approaches have been proposed to deal with the outliers rejection problem ((Hawkins, 1980),(Markou and Singh, 2003),(Zimek et al., 2012)). Outlier detection methods can be classified according to several characteristics: the use of training (pre-labeled) data, the assumption of a standard statistical distribution, the type of data set, dimension of detected outliers, type of detected outliers (global/local), among

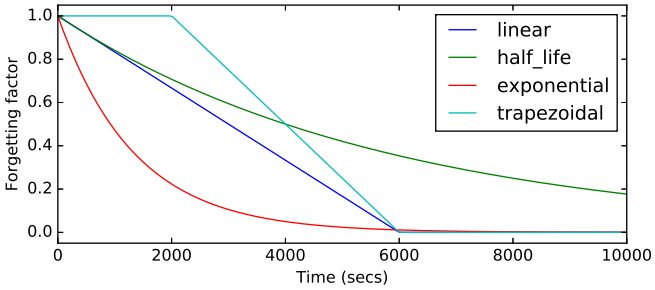


Fig. 4. Decay functions used to emulate data forgetting

others. Most of classification monitoring techniques focus on reject the abnormal observations derived from noised data.

In general, outliers indicate noise, damage or errors. Nevertheless, outliers can also be an indication of the occurrence of unknown events or unexpected patterns resulting from system evolution or reconfiguration.

Most commonly used outlier detection approaches are those classified as distribution-based, distance-based (Knorr et al., 2000), density-based (Breunig et al., 2000) or angle-based (Kriegel et al., 2008). If the probability density function (or functions) is (are) known and alleged to follow a normal distribution, the simplest approach to outlier detection is to compute the probability of a sample to belong to a class (or classes), and then use a threshold to establish its outlier character. This approach is equivalent to finding the distance of the sample to class means and threshold on the basis of how many standard deviations away the sample is (Markou and Singh, 2003).

Unfortunately, in most real world applications the data distribution is not known *a priori*, as a result, non-parametric methods are generally preferred. Non parametric approaches like distance-based, density-based or angle-based make no assumption about the statistic properties of the data which makes them more flexible than parametric methods. In this paper we use the clustering distance and density framework to detect abnormal data which may or may not represent the evolution of a known state or the emergence of a novel behavior.

Since no formal unique definition of outlier exists, the notion of outlier may greatly differ from one outlier detection technique to another. In this work, the outlierness or not of a sample is related to the outlierness of the μ-cluster that contains its information. Explicitly, for ***DyClee***, an *outlier sample* is defined as: *A sample contained in an $O\mu C$* and an $O\mu C$ is defined as *a representation of samples that, due to their position and density, do not correspond to any of the identified clusters nor represents relevant novel behavior*.

The distance and density analysis that ***DyClee*** performs allows it to detect both, global and local multivariate outliers, the former in all configurations and the latter when multi-density clustering is used. The outlier character of a sample is binary, that is, a sample can only be considered as outlier or not outlier. Nevertheless, the outlierness of a μ-cluster may be associated with its density which provides a score about how much outlier a μ-cluster is. In other words, the set of all $O\mu$-clusters can exhibit different densities although their densities remain close to each other, that is, in the same relative level with respect to the clusters densities ($S\mu$-clusters and $D\mu$-clusters).

If global density is used, ***DyClee*** detects only global outliers. As said in the previous section, in general, a μ-cluster is considered

as outlier if its density is lower than the median and the average of all clusters densities, as shown in equation 5. If local-density analysis is used, the equation 8 shows that an μ-cluster is considered as outlier iff its density is lower than the group median density and the group average density, being its group the set of all the μ-clusters connected (directly or indirectly) to it.

Using the local-density analysis allow low density populations (as faults) to be represented as well as high density populations (as is usually the case of normal behavior). In addition, the local-density analysis allows the detection of novelty behavior in its early stages when only a few objects giving evidence of this evolution are present. While it is deemed desirable to detect clusters of multiple densities, it is also important to maintain the ability to reject outliers. ***DyClee***'s solution to outlier rejection is based on the exclusion of the μ-clusters found to have low density in each analyzed group.

5. TESTING IN TOY EXAMPLES

This section will illustrate how ***DyClee*** can be used to achieve some important features in supervision. The first test aims to show the capability of tracking the system through different operation points, even if these points are relatively close to each other. To this end, we select the R15 dataset from (Veenman et al., 2002). This dataset of 600 points is generated by 15 similar 2D Gaussian distributions. ***DyClee*** deals with high overlapped data distributions from its conception. Figure 5 shows ***DyClee*** clustering results compared to those achieved in (Veenman et al., 2002). The 15 classes are correctly recognized.

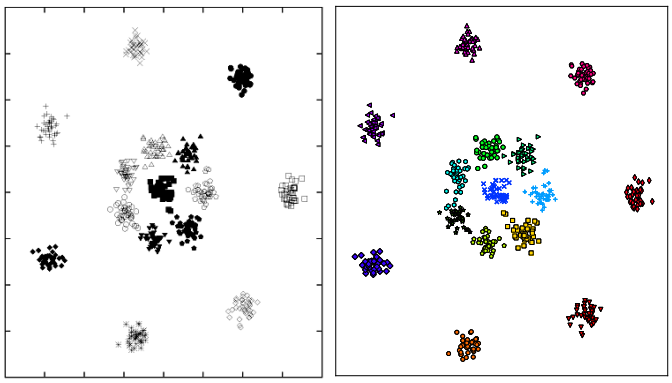


Fig. 5. Veenman Maximum Variance Cluster Algorithm (left) and ***DyClee*** clustering results (right)for R15 test set.

An academic industrial example suffering from this situation is the Tennessee Eastman Process. In this benchmark, two different products are produced from four reactants and the mass ratios between them vary from one operation mode to another.

As a second test we explore the ability of our algorithm to cluster multi-density distributions. As stated before, the ability to detect distributions evidencing different amount of samples or concentrations is a desirable feature in supervision since it can improves the detection and characterization of unknown behaviors. Multi-density situation are very common in industrial environments, where the measures coming from the process in normal operation mode(s) are wide more abundant that those coming from start-up or maintenance routines. Even more,

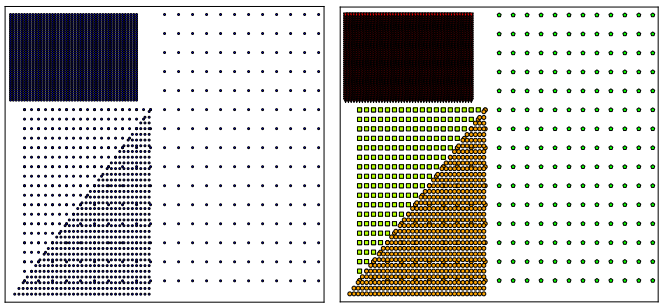


Fig. 6. At left, Clusters of varied density. At right ***DyClee*** clustering results

since faulty states are uncommon samples from those states will be also rare.

The toy example chose to test this feature is the multi-density set used in (Fahim et al., 2010). In this set four different clusters exhibit densities varying in a wide range. Two of these classes present also overlapping. The clustering results of our algorithm are shown in Figure 6. We can see that the four classes are correctly recognized.

Another highly desired feature comes from the fact that, in most of the real world applications, non-stationarity is typical and data is expected to evolve over time. In this example our algorithm is confronted with slowly time changing distributions, or as is usually called in the online learning scenario, concept drift (Gama et al., 2014).

These drifts can come, for example, from physical wear of some mechanical parts, by the sensitivity loss of some sensors or by the addition of a new source of noise. In order to shown ***DyClee*** performance over this issue, a synthetic set was created. Three clearly differentiated distributions are used to form three clusters. Two of this groups drift in time until shift positions as can be seen in Figure 7. Synthetic data are generated changing the center of the distribution each 100 samples.

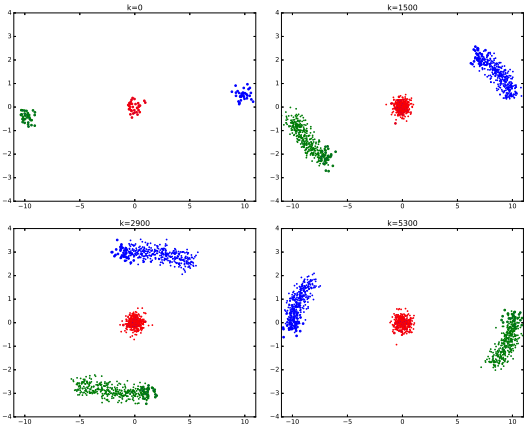


Fig. 7. Concept Drift toy example

Dynamic data cause several clusterers to fail in finding clusters distribution change since they cannot cope with evolution, losing hence, the tracking over the system state. Snapshots showing the distribution of μ-clusters in several time instants are depicted in Figure 8. It can be seen how clusters evolution is followed thanks to the drift of some of the existent μ-clusters and to the creation of new μ-clusters. Growth in the amount of clusters can be seen between snapshots one and two, and again

between snapshots two and three. $O\mu$-clusters are represented as gray boxes. This dynamic tracking is possible thanks to the implementation of the forgetting process that increase the clusterer reactivity to data changes.

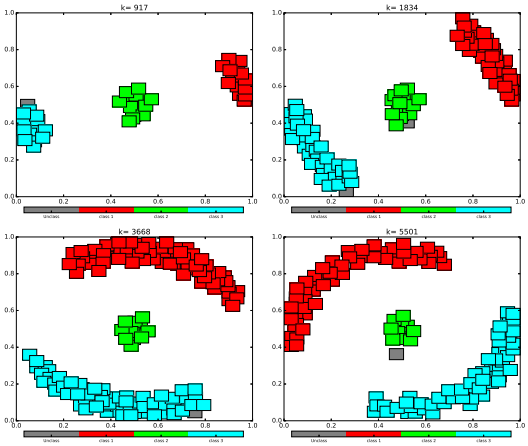


Fig. 8. μ-clusters following system evolution

In the next section these features help in the diagnosis of the CSTH benchmark introduced in (Thornhill et al., 2008).

6. DIAGNOSIS CASE STUDY: THE CSTH

The CSTH is a benchmark of a stirred tank in which hot (50°) and cold (24°) water are mixed and further heated using steam; the final mix is then drained using a long pipe. The configuration of this benchmark, developed by (Thornhill et al., 2008), is shown in Figure 9. It is assumed that the tank is well mixed so the temperature of the outflow is the same as that in the tank. Process inputs are set-points for the cold water, hot water and steam valves. Process outputs are hot and cold water flow, tank level and temperature. Process inputs and outputs represent electronic signals in the range $4-20mA$. The benchmark is tested in closed-loop. PID controllers are used to guide the plant as suggested in (Thornhill et al., 2008).

Thornhill *et al.* also suggests two operation points depending in whether or not the hot water flow is used. The suggested set-points for the operation points $OP1$ and $OP2$ are shown in table 1. Simulink models, with and without disturbances, are available at (Thornhill, web resource) website. The provided disturbances are real data sequences experimentally measured from the pilot plant at the University of Alberta.

We adapted this benchmark in order to implement dynamic events as evolving leaks or pipe clogging. The first simulated

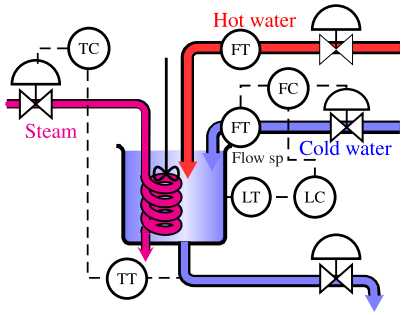


Fig. 9. Diagram of the continuous stirred tank heater

Variable	OP 1	OP 2
$Level$	12.00	12.00
$CWflow$	11.89	7.330
$CWvalve$	12.96	7.704
$Temperature$	10.50	10.50
$Steamvalve$	12.57	6.053
$HWvalve$	0	5.500

Table 1. Suggested operational points for the CSTH in mA

scenario is that of a pipe clogging. The second scenario implements several faults occurring alone or in pairs. Among the considered faults, tank leakage and valve stuck are found.

6.1 Scenario 1:Tracking state drift

One common problem in industrial applications is that states might drift when the physical parts of the system are exposed to wearing processes. In the case of the CSTH we simulate the evolution of $OP1$ when residues accumulate in the border of the output pipe causing a drop in the maximal output flow. The process measurements for this scenario can be seen as continuous lines in Figure 10. This figure also shows that our clustering algorithm is capable of following this evolution. The polynomial fit is also show in the figure, represented as dashed lines.

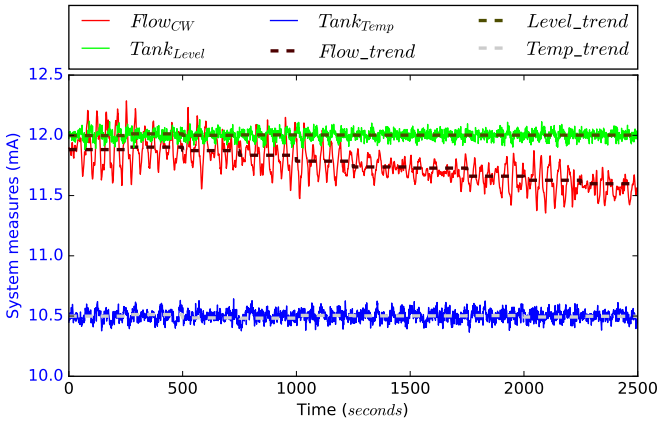


Fig. 10. ***DyClee*** TC and process measurements for scenario 1

As was said before following the system evolution at seen its measures can be a difficult task and even impossible when dozens of signals are analyzed at the same time. We propose to use a radar-like graphic representation to follow variables evolution. For the simulated scenario the graphics for $t = 250$ and $t = 2500$ are shown in Figure 11. The drift in the $Flow_{CW}$ variable are depicted as the filled area between the original characterized point and the current point.

6.2 Scenario 2: Tracking of multiple fault scenarios

Several faults between evolving leaks and stuck valves were simulated in this scenario. The total simulation time of this scenario is equivalent to a timespan of a month (2.419.200 $seconds$) in which the plant works the half of the time in $OP1$ and the other half in $OP2$ (operational points described in table 1). The faulty events included in this scenario are detailed in table 2. CSTH output signals are shown as background in Figure 12. At the beginning of the simulation the CSTH was working in $OP1$. Simple fault events and multiple faults events

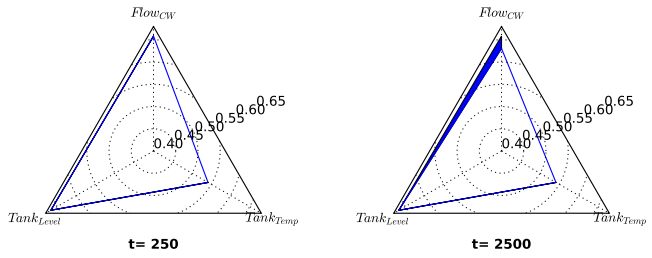


Fig. 11. ***DyClee*** clustering results for scenario 1.

t	Event	t	Event
0	Start at $OP1$	9.0	e-leak starts.
1.5	e-leak starts.	9.6	e-leak fixed
2.4	e-leak fixed	12.0	Changed to $OP2$
3.5	e-leak starts.	15.0	S_{valve} stuck 10%
3.8	2^{nd} e-leak starts.	15.6	Valve repaired
4.5	Leaks fixed	18.0	e-leak starts.
5.5	S_{valve} stuck 0%	18.4	Leak fixed
5.8	Valve repaired	20.0	HW_{valve} stuck 40%
6.5	HW_{valve} stuck 10%	20.6	Valve repaired
7.0	Valve repaired	24.2	End

Table 2. Scenario 2: Multiple fault simulation over a month timespan. (Simulation time in $sec \times 10^5$)

were simulated. ***DyClee*** Clustering results are shown in Figure 12.

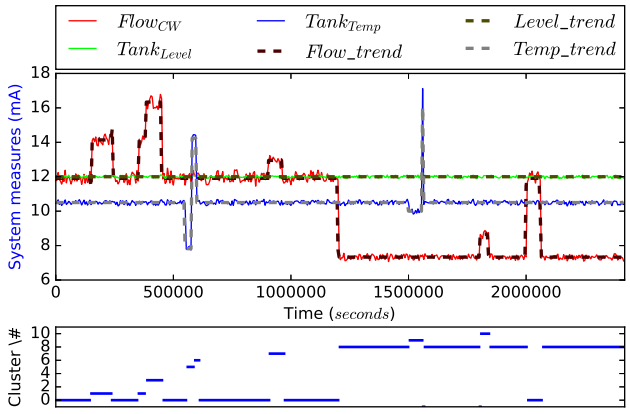


Fig. 12. ***DyClee*** clustering results and process measurements for scenario 2

Since ***DyClee*** works under a non-supervised learning paradigm, only the system measures are necessary as input for the algorithm. Unlike most of the tracking algorithms that can only track known behaviors, ***DyClee*** start of not behavior at all and built his knowledge when new behaviors are recognized. This can be seen in Figure 12 where ***DyClee*** can successfully track online the process and its evolution. Ten different behaviors are recognized. The label associated to these clusters is related to their order of apparition, and zero represents the non-representative behavior caused by extremely noised samples or by transition states.

In order to illustrate ***DyClee*** structural evolution a snapshot of the μ-clusters distribution in t between 405000 and 900000 seconds is presented in Figure 13. For this experiment the sampling time were $T_s = 0.5s$. The algorithm start window size

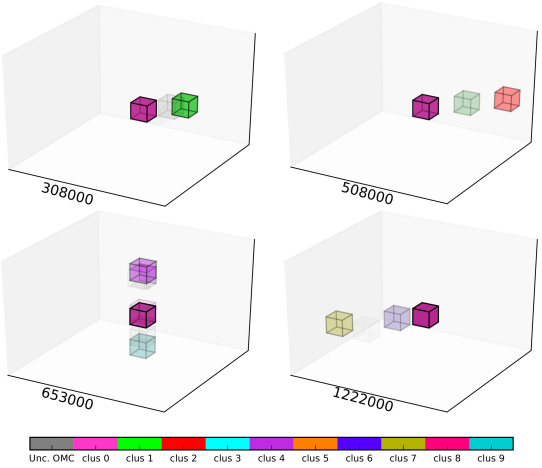


Fig. 13. Dynamic Clustering of the CSTH for scenario 2. t between 308000 and 1222000 seconds

was established as $win_{length} = 5000$ samples (2500 seconds), the minimum size as 1000 samples and the maximum as 10000 samples. The forgetting process were activated and the 'linear' forgetting function were selected with $\beta = 150000$, that is, thirty times the normal window size.

Clusters are represented in compact way in the radar plot shown in Figure 14. This figure shows the cold water flow as the key feature to detect the system evolution in most of states. Nevertheless, the tank temperature change is what characterize the cluster 5, corresponding to the stream valve fault.

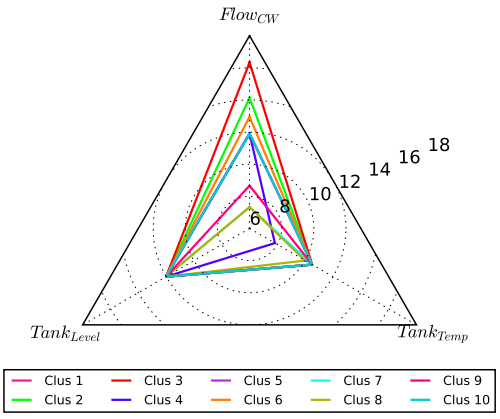


Fig. 14. Radar clusters' representation for scenario 2

7. CONCLUSIONS

In this paper, we propose a multi-density improvement to dynamic clustering that follows system evolution by adapting its forgetting process. This work illustrates with practical examples, how data-based clustering under non supervised learning paradigm can help in process supervision. The proposed algorithm is used to monitor industrial processes and has proved to be capable to detect different types of faults including those with time varying dynamics on the selected benchmark. The algorithm shows good performance in presence of disturbances and the results follow the evolution of the system. Tests have shown performance improvement when local-density analyses are used together with the fast distance-based clustering.

From a technical point of view, it would be interesting to compare the proposed algorithm with systems of different nature and include categorical features, which may consolidate the proposed methodology.

REFERENCES

Aggarwal, C.C., Han, J., Wang, J., and Yu, P.S. (2003). A framework for clustering evolving data streams. In *Proceedings of the 29th international conference on Very large data bases-Volume 29*, 81–92. VLDB Endowment.

Angelov, P. (2011). Fuzzily connected multimodel systems evolving autonomously from data streams. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 41(4), 898–910.

Barbosa, N., Massuyes, L.T., and Grisales, V.H. (2015). A data-based dynamic classification technique: A two-stage density approach. In *SAFEPROCESS 2015, Proceedings of the 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes*, 1224–1231. IFAC.

Bouchachia, A. and Vanaret, C. (2011). Incremental learning based on growing gaussian mixture models. In *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on*, volume 2, 47–52. IEEE, Elsevier.

Breunig, M.M., Kriegel, H.P., Ng, R.T., and Sander, J. (2000). Lof: Identifying density-based local outliers. *SIGMOD Rec.*, 29(2), 93–104.

Cao, F., Ester, M., Qian, W., and Zhou, A. (2006). Density-based clustering over an evolving data stream with noise. In *SDM*, 326–337.

Dash, S., Maurya, M.R., Venkatasubramanian, V., and Rengaswamy, R. (2004). A novel interval-halving framework for automated identification of process trends. *AIChE journal*, 50(1), 149–162.

Eduardo Mendel, F.M.V., Rauber, T.W., and Batista, R.J. (2011). Condition monitoring based on kernel classifier ensembles. In *9th IEEE International Conference on Industrial Informatics (INDIN)*, 81 –85. doi:10.1109/INDIN.2011.6034841.

Ester, M., Kriegel, H.P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databdata with noise. In *KDD*, 226–231.

Fahim, A., Salem, A.E., Torkey, F., Ramadan, M., Saake, G., et al. (2010). Scalable varied density clustering algorithm for large datasets. *Journal of Software Engineering and Applications*, 3(06), 593.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)*, 46(4), 44.

Gamero, F.I., Melndez, J., and Colomer, J. (2014). Process diagnosis based on qualitative trend similarities using a sequence matching algorithm. *Journal of Process Control*, 24(9), 1412 – 1424.

Hawkins, D.M. (1980). *Identification of outliers*, volume 11. Springer.

Hedjazi, L., Kempowsky-Hamon, T., Despnes, L., Le Lann, M.V., Elgue, S., and Aguilar-Martin, J. (2010). Sensor placement and fault detection using an efficient fuzzy feature selection approach. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, 6827 –6832.

Isaza, C.V., Orantes, A., Kempowsky-Hamon, T., and Lann, M.V.L. (2009). Contribution of fuzzy classification for the diagnosis of complex systems. In *Fault Detection, Supervision and Safety of Technical Processes*, 1132–1137.

Joentgen, A., Mikenina, L., Weber, R., and Zimmermann, H. (1999). Dynamic fuzzy data analysis based on similarity between functions. *Fuzzy Sets and Systems*, 105(1), 81–90.

Kempowsky, T., Subias, A., and Aguilar-Martin, J. (2006). Process situation assessment: From a fuzzy partition to a finite state machine. *Engineering Applications of Artificial Intelligence*, 19(5), 461–477.

Knorr, E.M., Ng, R.T., and Tucakov, V. (2000). Distance-based outliers: algorithms and applications. *The VLDB Journal – The International Journal on Very Large Data Bases*, 8(3-4), 237–253.

Kranen, P., Assent, I., Baldauf, C., and Seidl, T. (2011). The ClusTree: indexing micro-clusters for any stream mining. *Knoledge and information systems*, 29(2), 249–272.

Kriegel, H.P., Shubert, M., and Zimek, A. (2008). Angle-based outlier detection in high-dimensional data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, 444–452. ACM, New York, NY, USA.

Li, K., Yao, F., and Liu, R. (2011). An online clustering algorithm. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference on*, volume 2, 1104–1108. IEEE.

Markou, M. and Singh, S. (2003). Novelty detection: a review–part 1: statistical approaches and part 2: neural network based approaches. *Signal processing*, 83(12), 2481–2497.

Maurya, M.R., Paritosh, P.K., Rengaswamy, R., and Venkatasubramanian, V. (2010). A framework for on-line trend extraction and fault diagnosis. *Engineering Applications of Artificial Intelligence*, 23(6), 950 – 960.

Maurya, M.R., Rengaswamy, R., and Venkatasubramanian, V. (2007). Fault diagnosis using dynamic trend analysis: A review and recent developments. *Engineering Applications of Artificial Intelligence*, 20(2), 133 – 146.

Mitra, P., Pal, S.K., and Siddiqi, M.A. (2003). Non-convex clustering using expectation maximization algorithm with rough set initialization. *Pattern Recognition Letters*, 24(6), 863 – 873.

Thornhill, N. (web resource). The csth web site. Accessed: 2015-02-12. http://personal-pages.ps.ic.ac.uk/~nina/CSTHSimulation/index.htm.

Thornhill, N.F., Patwardhan, S.C., and Shah, S.L. (2008). A continuous stirred tank heater simulation model with applications. *Journal of Process Control*, 18(3), 347–360.

Vachtsevanos, G., Lewis, F., Roemer, M., Hess, A., and Wu, B. (2007). *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. John Wiley & Sons, Inc.

Veenman, C.J., Reinders, M.J., and Backer, E. (2002). A maximum variance cluster algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(9), 1273–1280.

Zimek, A., Schubert, E., and Kriegel, H.P. (2012). A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining*, 5(5), 363–387.

Educational Microgrid Testbed with Advanced Measurment Infrastructure and Demand Response

David Mejia, Diego Patiño * Eduardo Mojica-Nava **

* *Pontificia Universidad Javeriana, Bogota, Colombia (e-mail: david.mejia@javeriana.edu.co, patino-d@javeriana.edu.co).*
** *Universidad Nacional de Colombia, Bogota, Colombia (e-mail: eamojican@unal.edu.co)*

Abstract: The rapid transformation that the electric grid is facing towards the smart grid is a high trend in research nowadays. In particular, developed countries have being working in several projects that test and validate the new power system model given the advantages that it offers such as create a more efficient grid, increase reliability, robustness and promote a rational use of energy . In contrast, Colombia and similar countries recently began the attention to smart grid projects. Therefore, new tools that encourage the work on control systems oriented to the smart grids should be developed.
This paper presents a low cost microgrid small scale model designed and built on a laboratory environment. The project intents to serve as a tool in order to fill the gap between the knowledge and the interest associated with microgrid and smart grid at the E.E. undergraduate level. A clear description of the main functionalities and operation of the microgrid model such as distributed generation, intentional islanding, and sensitive loads is given. Also, two of the most important key aspects of the smart grid are explored in the prototype: advanced metering infrastructure and demand response. Finally, the project proves to be an example of motivation for the smart grid research in our academic sector.

Keywords: Smart Grid, Microgrid, Distributed Generation, Advanced Metering Infrastructure, Demand Response.

1. INTRODUCTION

Microgrid and smart grid power system models have emerged in the last decade as a response to several issues that the electricity industry is currently facing. For example, i) overload in the network infrastructure, ii) high costs and difficulty of the grid expansion under traditional model, iii) black outs that highly affect critical consumers and iv) low quality and lack of reliability of the service. To provide a power system that overcome those difficulties, the smart grid initiatives allow the integration of distributed generation, efficient management of electricity, self-healing of the medium voltage circuits and the possibility to design new electricity markets. In addition, the smart grid affords the possibility to give more services to the users in order to integrate them as active part of the system **?**Heydt (2010).

The research and development of the smart grid and microgrid have being growing specialy in the developed countries where it is easy to find important projects. Among these large scale projects, some examples are: the regional power grid with renewable energy resources in Japan and "More microgrids" in Europe Nikos and Hatziargyriou. The smart microgrid pilot project of the university of Genova that transforms a facility of the university into a microgrid system and the HyLab of the School of Engineering (Sevilla) Bonfiglio et al. (2012); Valverde et al. (2013) . Finally, the CERTS project in California integrates several scattered small scale generators and energy storage devices Lasseter et al. (2011).

In contrast, in Colombia and some countries of South America there are not many microgrid or smart grid pilot projects and it is shown a lack of interest to invest in the research and development of those topics. However, taking into account that 33.8 million people in the region lives in rural areas with high levels of poverty and without the electricity service, this situation is an opportunity to the development of islanded microgrids. A clear example of the microgrid potential is shown in the study of possibilities for implement smart microgrid projects over Chile Ubilla et al. (2014).

The efforts that have been carried out to test and validate the microgrid model are divided in two types. First, the microgrid test beds which objective is to prove the operation and control of the microgrid concept Jin and Tseng; Conference et al. (2014); Adinolfi et al. (2014). Second, the experimentation based on real time digital simulation (RTDS) and hardware in the loop systems (HIL). Those platforms explode the computational capabilities of the RTDS to prove control algorithms and test smart hardware

⋆ Sponsor and financial support acknowledgment goes here. Paper titles should be written in uppercase and lowercase letters, not all uppercase.

devices Jeon et al. (2010); Chen et al. (2014); Stanovich et al. (2013); Tatcho et al. (2010). The principal disadvantages of both test bed and HIL system are the high investment cost involved in the construction of even a simple microgrid demonstration system.

In Colombia and similar countries the economic situation makes it is difficult to get founds to create microgrid test bed and also it is not easy to have access to the technological resources needed for implement a RTDS or a HIL microgrid system. Hence, objective of the project is to develop a practical implementation of the microgrid concept into a low cost laboratory based test bed. In the last years, the small scale test bed has emerged as an economical option to explain the functions and possibilities of smart grids. One example of this type of project is the test bed in smart grids that was developed at the Georgia State University Song et al. (2012). However, it is difficult to find more projects within this approach.

The proposed low cost laboratory-based microgrid scale model shows the grid connected and islanded modes operation as well as the response to failures on the main grid and loads. Measurements through the advanced measurement infrastructure (AMI) of the different load and generators are taken with proprietary smart devices. The microgrid test bed is controlled by a supervisory control and data acquisition system (SCADA) developed in Labview with a graphical user interface (GUI) that emulates a power system control center. Also, the effect of the demand response (DR) implementation on the main grid and costumer side is shown.

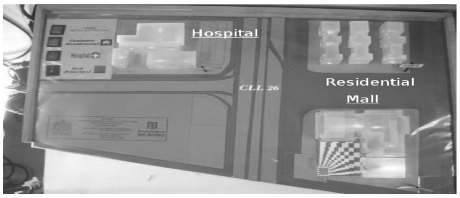
The article is organized as follows: Section II has a detailed description of the implementation of the microgrid model. Section III explains the microgrid operation modes. In section IV the description of the advanced measurement infrastructure and demand response is carried out. The main results of microgrid operation and the impact of the small scale test bed are illustrated in sections V and VI. Finally, conclusions and further directions of the project are presented.

2. MICROGRID TEST BED

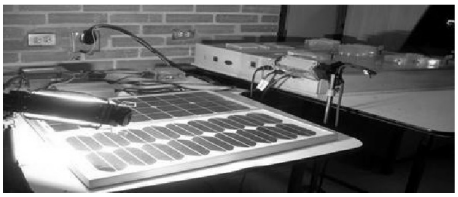
The small scale microgrid test bed is designed to explain by hardware demonstration the microgrid concept as well as its functions and operation (See Fig. 1).

2.1 Power Grid

The architecture of the small scale microgrid is shown in Fig. 2. The structure of the power grid is based on the microgrid model proposed by Robert Lasseter in Lasseter (2002). The circuit is a small AC single-phase radial electric grid that has three feeders for loads and two feeders for DG. Loads are supplied by the main grid and the distributed generation (DG) at 120V. At the single coupling point, the microgrid interconnects to the main grid through a power electronic interface associated to the slack node . The DG is composed by two distributed generators which are renewable energy sources (photovoltaic) and a storage unit (Lead acid). The



(a) Loads



(b) Distributed Generation

Fig. 1. Microgrid test bed

circuit also has three resistive load nodes that represent different type of costumers. The boxes in front of the different elements of the microgrid represent the different proprietary smart devices.

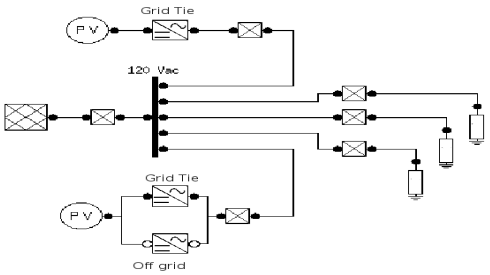


Fig. 2. Microgrid test bed

2.2 Communication Network

The scale model works under a wireless network sensor that operates with the 802.15.4 zigbee protocol. The wireless devices (nodes) send the measurement data to a master terminal unit of the control system for the real-time energy monitoring of the microgrid. The structure of the sensor network is shown in Fig. 3 where the central controller (n_{CC}) communicates with every node. The remaining nodes are: node n_1 which is the static power switch or main grid sensor/controller. Nodes n_2 to n_4 are the sensor/controllers in charge of the loads, n_5 and n_6 are the nodes corresponding to the master and slave DG. The node n_7 controls the demand response actions. Finally, node n_8 represents the switching node which selects between the grid-tie and off grid inverters for the distributed generator n_5.

In Fig. 3, it is shown a two way communication between n_1 and n_8. In this case, n_1 and n_8 interact to coordinate the moment of reconnection between the microgrid and the main grid. This communication process ensures a safe operation avoiding inverter hazard when the main grid interconnects with the microgrid.

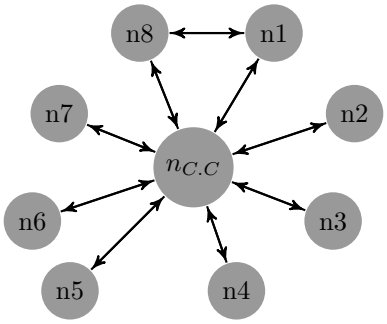


Fig. 3. Sensor network architecture

2.3 Supervisory control and data acquisition system

The SCADA system that operates the microgrid scale model is divided in two parts: i) the master terminal unit (MTU) is a computer with one node of communication (n_{CC}) that carries out supervisory tasks, alarm management and data transfer. The MTU also controls the different microgrid operation modes and its transitions. The global control tasks are executed by a state machine in charge of the desired microgrid operation. ii) Seven remote terminal units (RTU) are responsible for the collection of the energy data from the DG, and loads. It executes the local control of disconnection in case of failure in each node.

3. MICROGRID OPERATION

The microgrid has two principal operation modes: grid connected mode or normal operation. In the first scenario the microgrid is connected to the main grid and supply its demand of energy from both main grid and distributed energy resources (DER). The stand alone or island mode is the scenario when some failure or abnormal condition happens in the main grid and the microgrid must supply its loads (at least the sensitive loads) only with the DER.

When the microgrid has only renewable energy in the DG, additional transition modes are needed in order to synchronize DG with and without the main grid (see Fig. 4). In addition, when the demand response programs are active it is considered as another operation mode that is part of the grid connected mode.

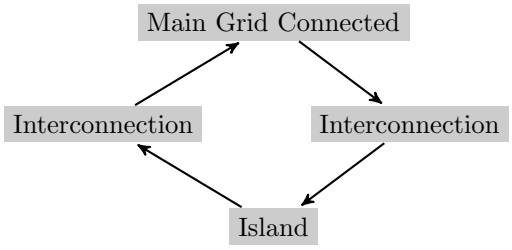


Fig. 4. Operation modes block diagram

4. SMART GRID FEATURES

The use of the test bed extends from the microgrid operation to include two of the main characteristics of the smart grid. The advanced measurement infrastructure is the technological basis for gathering the necessary information in real-time from the power system elements in order to create a more flexible and efficient grid. The demand response initiative includes the participation of the customer side turning the from a passive to an active player of the power system.

4.1 Advanced metering infrastructure

One of the main characteristics that the smart grids have, is the ability to measure energy in real-time through an advanced metering infrastructure . The AMI takes values of energy consumption of the whole grid and sends that information to the control system and customers National Energy Technology Laboratory (2008).

In order to create the AMI and integrate into the microgrid testbed, a low cost smart meter device is presented in Fig. 5. This device is in charge of collecting and transmitting the energy information from the load and generation nodes to the MTU. The device is capable to provide real time measurements of power, voltage, current and frequency. Also the RTU can take local control decisions in order to protect and island the different loads, DER of the microgrid. In addition, the measurements of the smart devices where validated comparing them with a energy analyzer Fluke 435 where an 2% error is obtained in the worst scenario.

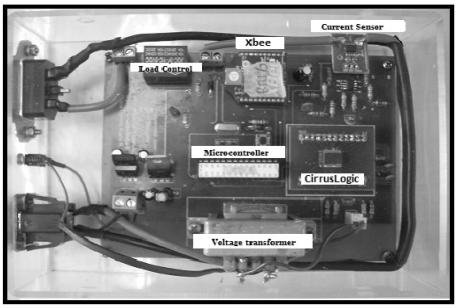


Fig. 5. Advance measurement infrastructure (RTU):

4.2 Demand response implementation

In a smart grid scenario, users of the electric service take decisions on their consumption based on the information obtained through the AMI. The active participation of the users in the network is known as the demand response Rahimi and Ipakchi (2010); Medina et al. (2010). The different programs implemented on the test bed are: price based plan (PBP) and incentive based plan (IBP).

In the PBP the residential and commercial users react to the price variations as shown on the eq. (1). When the price signal corresponds to high the users disconnect from the grid showing they are not willing to buy energy at that price.

$$PBP\_Participants = \begin{cases} \text{OFF: If, } Price = High; \\ \text{ON: If, } Price = Low. \end{cases} \quad (1)$$

The simulated customers that participate on the IBP react as a result of some incentive signal. The participants of

IBP sign a contract with the energy provider in which the company gives some incentive (rate discounts, money, etc.) in exchange for apply direct control and disconnect them. The direct control will happen only in some special situations defined in the same agreement.

For demonstration purposes the negotiation terms consist that the participants (some residential) will disconnect from the microgrid each time the highest peak in the energy demand profile occur. The participants on this DR program will act depending on the condition shown in eq. (2).

$$IBP\_Participants = \begin{cases} \text{OFF If: } 18h \leq t < 22h. \\ \text{ON Otherwise.} \end{cases} \quad (2)$$

5. EXPERIMENTAL RESULTS

In order to show the correct operation of the small scale microgrid test bed, a test protocol divided in two parts has been designed. First, six different scenarios working in real-time prove the different operation modes of the microgrid moving towards islanded mode and returning to the grid connected mode. Second, another scenario to test the implementation of the DR programs has a fixed simulation time. In the DR scenario one week simulation takes seven real-time minutes. Due to the space limitation only the performance of the microgrid in grid-connected, islanded and DR modes are presented.

Grid connected mode At the beginning of the simulation during the first 18 seconds all the nodes are active, notice that the loads has positive power (power consumed), and the DG has negative power (power produced) (see Fig. 6). In this scenario, it is clear that the main grid is delivering only 50W instead of 65W which is the total load consumption at that moment. Thus, the load takes the remaining 15W from the DG proving that DG reduces main grid consumption. From 18s to 185s, local failures occur on the different load nodes showing the reconnection dynamic.

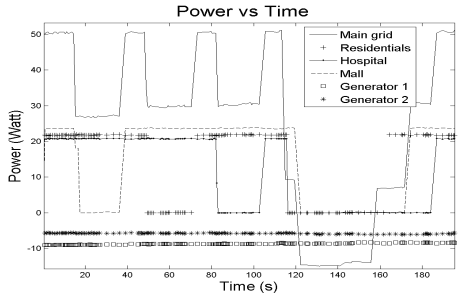


Fig. 6. Different Load Failure during microgrid grid connected operation

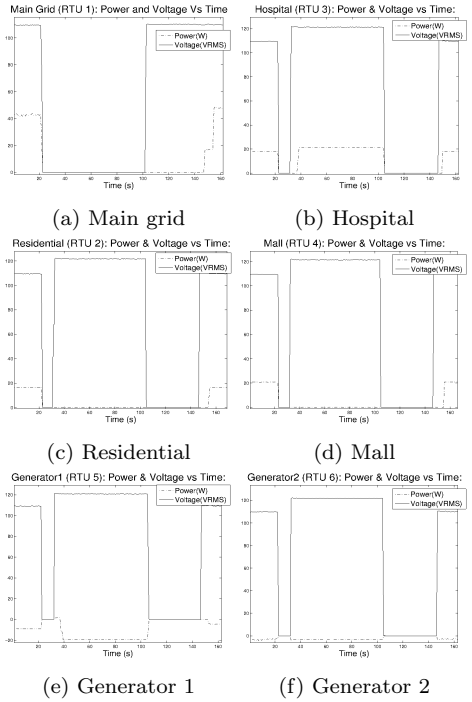
A special event is shown in Fig. 6 between 123s and 160s. When all the loads are off due to their failures, DG continues to deliver its power and notice that the power registered by the main grid sensor (in black straight line) is below 0W. This case shows how the microgrid delivers those 15W from the DG to the main grid.

For the majority of countries, the trade of energy between customers and the energy system is not regulated. For that reason, an optimized storage system that takes all the power delivered from the renewable energy sources and avoids injecting it into the main grid is needed. Another option is to implement a control strategy that ensures not to produce more energy than the energy consumed by the microgrid. In a real microgrid application this could happen not only when the microgrid loads fail but also when the DG supply is greater than the power consumed by the load.

Island mode The Fig. 7 (from 35s to 105s) shows an example of the islanded operation. Figs. 7b, 7c, 7d show that even though all load nodes have 120V only the hospital is consuming energy, which indicates that the residential and mall node are disabled by the central controller. The hospital (sensitive load) keeps operating normally and consuming all its power without being affected by the blackout.

The behavior of the microgrid test bed during the island mode illustrates the priority that the microgrid gives to sensitive loads and how helpful a microgrid system is when those type of loads are involved.

Between 35s and 105s in Fig. 7e it seems that the DG produces more energy than its maximum capacity which is 15W, but this is because the battery supports the DG to supply all the power necessary to the sensitive loads.



(a) Main grid (b) Hospital

(c) Residential (d) Mall

(e) Generator 1 (f) Generator 2

Fig. 7. Island mode

Demand response For the analysis of the different DR programs operation, data was collected by setting the system running in a period of one week simulated time for each program. The objective of this demonstration is to present the behavior of the system when a group of users decide to participate in a DR program. By changing the pattern of the energy the DR participants could receive economic benefits such as lower prices, subsidies or directly payment for the amount of energy they reduce. At the

same time the energy company reduces the risk of outages and damage on the electrical infrastructure without the high peaks specially at night.

In Fig. 8, the power consumption of Bogotá during one week is shown. It is clear that the consumption of Bogotá represented in this graph shows two peaks over 1950 MW. The first one is during the middle close to 12m, and the second one is the typical peak period of the night around 20h. Therefore, the price signal during both events will be high price.

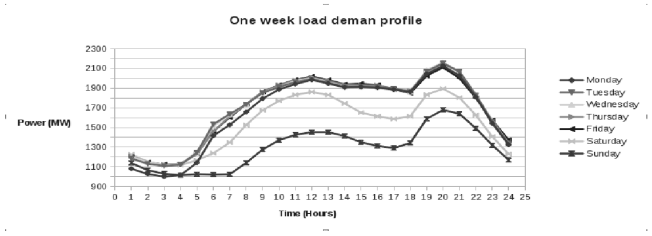


Fig. 8. One week Load demand profile in Bogotá, Colombia Xm Colombia (2012)

Price plan: Fig. 14 and 15 show how the residential and commercial power consumptions behaves when they decide to participate in the price plan. In the different simulations, the limit of power for the low price rate was 1950 MW. When the power demand in Bogotá data corresponds to a value over that limit the price signal will show high price. Then, at that moment some residential customers and circuits of the mall will be turned-off.

During the two peak periods of the day (See Fig. 10) the the residential node reduced the energy consumption from the average of 19W to 14W. The same situation happens with the mall (See Fig. 11), it reduces its consumption on the high price periods from 23W to 15W. Notice that in both results only a part of the consumers reacts to the price signal not the whole node (residential or mall). The changes on the consumers use of energy also has an direct impact on the energy consumption of the main grid reducing the energy deliver from 65W to 50W as it is shown in Fig. 9. Finally, during the weekend no one is disconnected from the microgrid because the consumption of Bogotá never cross 1950 MW limit, then the price signal is always low price (See Fig. 8).

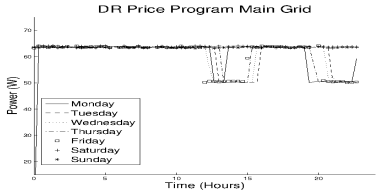


Fig. 9. Energy delivered from the main grid in DR price program

Incentives plan: Fig. 8 shows that during all the week after the 19h and before 22h the high peak period of the day occurs in Bogotá. During this period the energy consumption of the residential node varies as Fig. 13 shows. The change in the power consumption of the

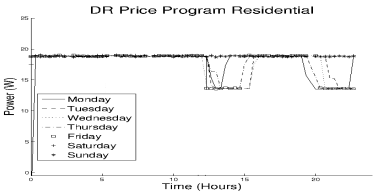


Fig. 10. Consumption of energy of residential in DR price program

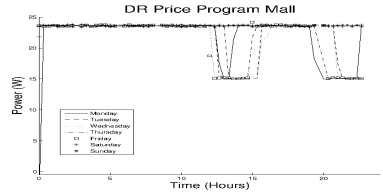


Fig. 11. Consumption of energy of mall in DR price program

Residential load represents a power reduction of 14W approximated in the main grid node (see Fig. 12).

Fig. 14 demonstrates how the commercial node does not have any reduction on its energy consumption during the incentive base program. This illustrate that the commercial customers are not interested in participate in a program where the energy company has the decision to connect/disconnect. In general, the mall does not want to allow direct control because it can lead not only to comfort degradation but also to economic losses.

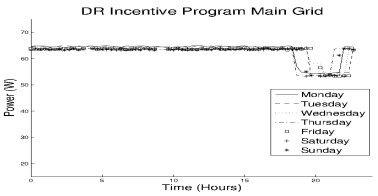


Fig. 12. Energy delivered from the main grid

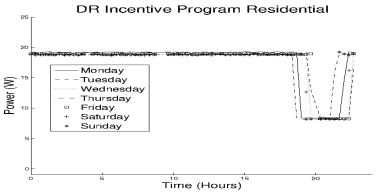


Fig. 13. Residential response to DR incentives program.

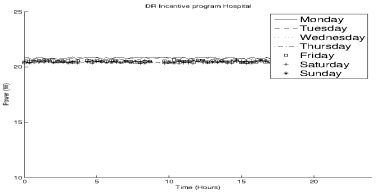


Fig. 14. Mall response to DR incentives program.

6. IMPACT OF THE PROJECT

The small scale microgrid test bed is a typical example of an smart grid project in which a multidisciplinary work

is needed. For instance, the test bed includes: power system integration, control systems, communication network development and electronic design.

The microgrid small scale test bed was presented to E.E students from last semester and motivate some of them to develop their senior projects in topics related with microgrid and smart grids. The different projects where: an automation of a low power educational fuel cell system, a communication protocol oriented to smart grids, a robotic platform of electric vehicles and a ring main unit (electrical measurement device) prototype.

Finally, the project was shown to CODENSA the energy distribution company of Bogotá. Given the positive opinion that the test bed received, the university get the funds to create a new version of the test bed which will be a micro smart grid testbed prepared for the company.

7. CONCLUSION

In this work a hardware based small scale microgrid test bed that integrates control systems, communications, sensing technology and power systems in order to illustrates the concept of the microgrid model as well as advanced measurement infrastructure and demand response is presented. The energy monitoring of the system affords the possibility to obtain in real-time the state of the microgrid and operate it secure and efficiently. In addition, the use of a friendly human user interface which emulates a real control system allow a good interaction between the user and the test bed in order to enhance the comprehension of how a microgrid works. Also, the real time demonstration of the demand response behavior allows to enhance the comprehension of this concept and understand how the demand side participation can benefit the customer as well as the energy company.

With these features, the test bed can be used as a tool for teaching and training in introductory smart grids subjects as well as be a platform for research. Finally, the project is an example for the region and similar countries to develop small scale smart grid test beds which can encourage students and energy companies to get involved in smart grid control research.

REFERENCES

Adinolfi, F., Burt, G., Crolla, P., D'Agostino, F., Saviozzi, M., and Silvestro, F. (2014). Distributed Energy Resources Management in a Low Voltage Test Facility. *IEEE Transactions on Industrial Electronics*, 0046(c), 1. doi:10.1109/TIE.2014.2377133.

Bonfiglio, a., Delfino, F., Pampararo, F., Procopio, R., Rossi, M., and Barillari, L. (2012). The Smart Polygeneration Microgrid test-bed facility of Genoa University. *Proceedings of the Universities Power Engineering Conference*. doi:10.1109/UPEC.2012.6398656.

Chen, B., Butler-purry, K.L., Goulart, A., and Kundur, D. (2014). Implementing a Real-Time Cyber-Physical System Test Bed in RTDS and OPNET. 1–6.

Conference, M.C., Kabalci, E., Hossain, E., and Bayindir, R. (2014). Microgrid Test-bed Design with Renewable Energy Sources. 907–911.

Heydt, G.T. (2010). The next generation of power distribution systems. *Smart Grid, IEEE Transactions on*, 1(3), 225–235.

Jeon, J.h., Kim, J.y., Kim, H.m., Kim, S.k., Cho, C., Kim, J.m., Ahn, J.b., and Nam, K.y. (2010). Development of Hardware In-the-Loop Simulation System for Testing Operation and Control Functions of Microgrid. *IEEE Transactions on Power Electronics*, 25(12), 2919–2929. doi:10.1109/TPEL.2010.2078518.

Jin, G.Y. and Tseng, K.J. (????). System Design for the Building-integrated Microgrid Test-bed - a Case Study.

Lasseter, R.H. (2002). MicroGrids. In *2002 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No.02CH37309)*, volume 1, 305–308. Ieee. doi:10.1109/PESW.2002.985003.

Lasseter, R.H., Eto, J.H., Schenkman, B., Stevens, J., Vollkommer, H., Klapp, D., Linton, E., Hurtado, H., and Roy, J. (2011). CERTS Microgrid Laboratory Test Bed. *IEEE Transactions on Power Delivery*, 26(1), 325–332. doi:10.1109/TPWRD.2010.2051819.

Medina, J., Muller, N., and Roytelman, I. (2010). Demand response and distribution grid operations: Opportunities and challenges. *IEEE Transactions on Smart Grid*, 1(2), 193–198.

National Energy Technology Laboratory (2008). ADVANCED METERING INFRASTRUCTURE.

Nikos, P. and Hatziargyriou (????). More microgrids EU pilot projects. URL `http://www.microgrids.eu/index.php?page=kythnos`.

Rahimi, F. and Ipakchi, A. (2010). Demand response as a market resource under the smart grid paradigm. *Smart Grid, IEEE Transactions on*, 1(1), 82–88.

Song, W.Z., De, D., and Tan, S. (2012). A wireless smart grid testbed in lab. *Wireless Communications, IEEE*, 19(June), 58–64.

Stanovich, M.J., Leonard, I., Sanjeev, K., Steurer, M., Roth, T.P., Jackson, S., and Bruce, M. (2013). Development of a smart-grid cyber-physical systems testbed. *2013 IEEE PES Innovative Smart Grid Technologies Conference, ISGT 2013*, 1–6. doi:10.1109/ISGT.2013.6497874.

Tatcho, P., Zhou, Y., Li, H., and Liu, L. (2010). A real time digital test bed for a smart grid using RTDS. *Power Electronics for Distributed Generation Systems (PEDG), 2010 2nd IEEE International Symposium on*, 658–661. doi:10.1109/PEDG.2010.5545920.

Ubilla, K., Jimenez-Estevez, G.a., Hernadez, R., Reyes-Chamorro, L., Hernandez Irigoyen, C., Severino, B., and Palma-Behnke, R. (2014). Smart Microgrids as a Solution for Rural Electrification: Ensuring Long-Term Sustainability Through Cadastre and Business Models. *IEEE Transactions on Sustainable Energy*, 5(4), 1310–1318. doi:10.1109/TSTE.2014.2315651.

Valverde, L., Rosa, F., and Bordons, C. (2013). Design, planning and management of a hydrogen-based microgrid. *Industrial Informatics, IEEE Transactions on*, 9(3), 1398–1404.

Xm Colombia (2012). Xm: Colombia Power System Operator.

Energy Price and Load Estimation by Moving Horizon Estimator with Holt-Winters Model

J. Rendón * **P. Deossa** ** **A. Márquez** *** **J. Espinosa** ****

* *Universidad Nacional de Colombia, Facultad de Minas, Cr. 80 No. 65 -233, Medelln, Antioquia (e-mail: jerendonr@ unal.edu.co)*
** *Universidad Nacional de Colombia, Facultad de Minas, Cr. 80 No. 65 -233, Medelln, Antioquia (e-mail: padeossa@ unal.edu.co)*
*** *Universidad Nacional de Colombia, Facultad de Minas, Cr. 80 No. 65 -233, Medelln, Antioquia (e-mail: amarque@ unal.edu.co)*
**** *Universidad Nacional de Colombia, Facultad de Minas, Cr. 80 No. 65 -233, Medelln, Antioquia (e-mail: jespinov@ unal.edu.co)*

Abstract: With the increasing penetration of smart grid technologies, prediction and forecasting task of the power system variables such as load and spot price became a difficult task. Traditional methodologies lack performance to capture the new dynamics of the power system. The Holt Winters model is widely used to estimate seasonal series. This model requires an heuristic setting of its non-dynamic parameters. This paper proposes a comparative study between Kalman Filter and Moving Horizon Estimator as optimal dynamic identification methods for the Holt-Winters model parameters. In order to evaluate the proposed optimal setting performance, the energy price and load time series of the Colombian electrical power system were used as case study, furthermore the classical estimation methods are chosen as reference frames.

Keywords: Moving Horizon Estimator, Kalman filter, Holt-Winters, energy markets.

1. INTRODUCTION

Operation of electric power systems is focused on ensuring the safety, continuity and reliability in the generation and delivery of electricity throughout the system. A required task to meet the mentioned objectives is to plan and predict the system behavior. Predictions and forecasting of the electric power systems variable had been made since several years ago, however recently, the penetration of smart grid technologies such as: green power plants, demand response strategies, climate change and markets liberalization have increased the dynamics and uncertainties of the variables. Now prediction and forecast for dynamic problems is an increasing relevant research topic. Load and energy spot price are two important driving variables in power system planning and operation, these series have direct influence in the generation-expansion plans (long term) and the economic dispatch of the system (short term), as mentioned, the influence of the smart grid technologies are changing the dynamics of the power systems included the mentioned variables, in this sense, the seasonality properties of the mentioned time series is changing and the traditional tools used to estimate and forecast the time series are failing compromising efficiency of the power system.

The literature reports several studies about modeling. Recent publications mentions the following research topics: estimation and prediction of price and load. In (Niimura, 2006) two different kind of models are presented: statistics models and autoregressive models. Statistic models are focused in the ARMA,ARIMA and GARCH structures. Computational methods presents fuzzy models, neural networks and chaotic methods. This work concludes that model performance highly depends on the prediction horizon and the model selection depends on the expected precision and the data variance. In (Weron, 2014a) different prediction techniques are applied to energy price time series, the methodologies review includes: multiagent methods, methods of fundamental decomposition by components, regressive methods, and evolutionary computation methods are reviewed. This paper also discusses the future of predicting price and concluded that: the techniques that predict better price are combinations of the methods discussed and prediction techniques intervals, due to the high volatility presents this series. In (Suganthi and Samuel, 2012) and (Singh et al., 2012) the authors described different load prediction models such as autorregressive models, fuzzy logic, neural networks, smoothing exponential model, expert systems and ant colony algorithms. All these models and techniques are currently used for planning and forecasting power system load. In (Haghi and Tafreshi, 2007) a categorization of the methods used to predict price in all time horizons (long, medium and short term) is shown. Methods reviewed in this work are: game theory based models, fuzzy and neural networks models, statistics methods like time series and autorregressive models. The study compares the mentioned models emphasizing their advantages. Model validation is performed in short time prediction horizon with low volatility time series. In (Ramírez, 2013) time series, neural networks

and regression based methodologies are compared with exponential smoothing models including the Holt Winters in short, medium and long term horizons. Here the lowest error is achieved with the Holt-Winters model at different small time horizons compared with the other techniques. But concluded that due to the non-inclusion of external variables in the different techniques, none of them will have a goo performance in long term predictions.

Now, presented models are applied to estimation and prediction of price and load, however they have some failings as is the case autorregressive models where prediction is poor in the short term as they do not consider the effect of external variables (Ramírez, 2013), or neural networks where a high computational cost to determine the number of neurons and is considered an unstable technique (Snchez and Velsquez, 2010); similarly in fuzzy models which have a bias in the results for the same model topology (Krafft and Mantrala, 2009). Furthermore in (Goodwin, 2010), (Chatfield and Yar, 1988) and (Gardner Jr., 2006) one of the most widely used for prediction and estimation series models, is the Holt-Winter model, whose trend and seasonality are set, as prices and load time series. These papers reports a reduction in error estimation and low computational burden when the Holt Winters models is applied to short term load and price estimation but in long and medium term time windows the results do not exhibit the mentioned benefits. This lack of efficiency is attributed to the model static structure which causes a cumulative error related with the presence of exogenous variables in the data series. Facing these facts, this work evaluates the technique Moving Horizon Estimator (MHE) discussed in (Valencia et al., 2011) and (Boulkroune et al., 2010) as a method of adjustment for the Holt-Winters model. This model is rewritten in a novel way in terms of state space in order to facilitate the development and solving of proposed estimator. At the same time it also seeks to verify what are the conditions of model structure and the estimator that ensure optimal estimation error reduction. The performance is compared with other classical estimation techniques like Kalman Filter and autorregressive methods.

This paper reports these results as follows: in Section 2 the price 2013 and demand 2014 series are described, these time series will be used to compare the Holt-Winters adjustment model with optimal estimation techniques. In Section 3 the Holt-Winters model is adjusted, finally, in Section 4 the optimal estimation will be made through the proposed techniques, here the quantitative analysis of the estimation errors is made for each model. Finally the conclusions are presented.

2. TIME SERIES DESCRIPTION

This work uses the 2013 Spot Price time series (hourly resolution), shown in Fig. 1 and the 2014 first semester load for regulated users for the EPM retailer (diary resolution), shown in Fig. 2. The load and price time series are available in XM web page (XM, 2016). Series data found on this page are preprocessed and are used to formulate the Figures and the autocorrelation functions shown. These series are chosen because they have a trend and seasonality characteristics, this fact can be used to properly formulate the model and also to improve estimation techniques presented.

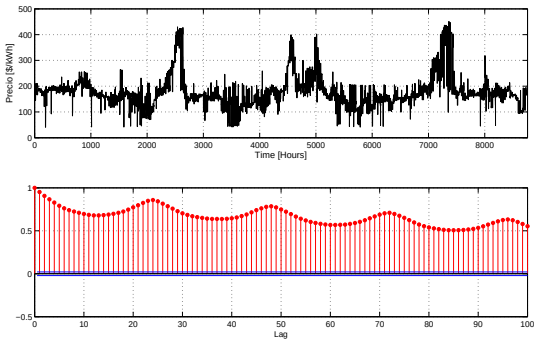


Fig. 1. Top: 2013 Price series. Bottom: 2013 Price series ACF

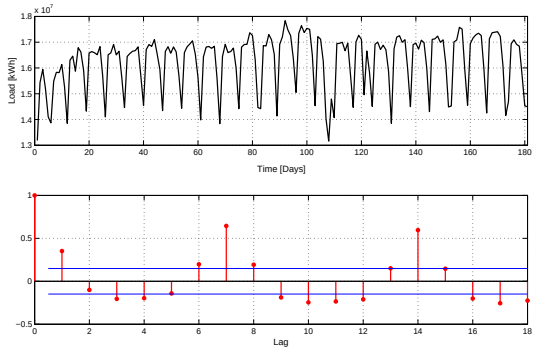


Fig. 2. Top: 2014 Load series. Bottom: 2014 Load series ACF

Figure 1 shows evidence that the time series has high random component influenced by external variables, by inspection it was thought that the series is not seasonal, but the auto correlation function (ACF) shows a seasonality component every 24 hours. Furthermore in Fig. 2 the weekly seasonality of the load series is clearly and is checked by ACF inspection. Validation of the seasonality of the series, is made through the Dickey-Fuller test (ADF Test) (Ng and Perron, 1995) and when this is not conclusive, a logarithmic transformation and adjustment of trends in the series is made to stabilize the variance and obtain a the correct conclusion, as explained in (Weron, 2014b), (Montgomery et al., 2015). Table 1 shows a summary of the results obtained for the series with and without logarithmic transformation.

Table 1. ADF test for time series

Time series	**Inspection ACF**	**ADF Test**	**Conclusion**
Price 2013	Not seasonal	0	Not seasonal
Load 2014	Seasonal	0	Not seasonal
Transformation Price 2013	Seasonal	1	Seasonal
Transformation Load 2014	Seasonal	1	Seasonal

3. THE HOLT-WINTERS MODEL

Holt-Winters (HW) models are often used in seasonal time series with disagreeable trending, seasonal or random variation, and they are grouped in additive or multiplicative. This model is the result of Holt (1957) and Peter Winston (1960) investigations, which proposed prediction methods whose principles are founded on the second-order exponential smoothing (Goodwin, 2010), (Hyndman et al., 2008). In (Gardner Jr., 2006) a complete state of the art about exponential smoothing models and the different forms of the Holt-Winters models are presented. Because the profiles of the proposed series, an additive seasonality and a linear trend will be assumed (seven days for load and 24 hours for price), based on this, an additive trend model will be used as follows:

$$\mu_k = \alpha y_k - \alpha S_{k-p} + (1-\alpha)\mu_{k-1} + (1-\alpha)T_{k-1} \quad (1)$$

$$T_k = \gamma\mu_k - \gamma\mu_{k-1} + (1-\gamma)T_{k-1} \quad (2)$$

$$S_k = \delta y_k - \delta\mu_k + (1-\delta)S_{k-p} \quad (3)$$

where μ_k is the level component, ${}_kT$ is the trend component, S_k is the seasonal component and p is the period in a seasonal cycle and α, γ and δ are parameters of the model.

Based on the presented model this work represents the Holt Winters model in a state space representation. This representation is widely used in estimation and optimization problems and control theory in general. The state space model is given as follows:

$$x_{k+1} = Ax_k + Bu_k \quad (4)$$

$$y_k = Cx_k + Du_k \quad (5)$$

$$x_k = [\,\mu_k \;\; T_k \;\; S_{k-p+1} \;\; S_{k-p+2} \;\; \cdots \;\; S_k\,]^T \quad (6)$$

$$A = \begin{bmatrix} 1-\alpha & 1-\alpha & -\alpha & 0 & \cdots & 0 \\ -\alpha\gamma & 1+\alpha\gamma & -\alpha\gamma & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \cdots & 1 \\ -\delta(1-\alpha) & -\delta(1-\alpha) & (\delta\alpha+(1-\delta)) & 0 & \cdots & 0 \end{bmatrix} \quad (7)$$

$$B = [\,\alpha \;\; \alpha\gamma \;\; 0 \;\cdots\; 0 \;\; \delta - \delta\alpha\,]^T \quad (8)$$

$$y_k = \mu_k + T_k + S_{k-p+1} \quad (9)$$

$$C = [\,1 \;\; 1 \;\; 1 \;\; 0 \;\cdots\; 0\,] \quad (10)$$

$$D = [\,0\,] \quad (11)$$

This model does not have an input (exogenous) vector. When the model is used for prediction, it is assumed as an autonomous system. Regarding the initial values of the states there are several heuristics to estimate these values as described in (Segura and Vercher, 2001), due to the ease in calculations compared to other methods, the initial state of this model are calculated with the methodology proposed by (Makridakis et al., 1998):

- $\mu_0 = \bar{D}_1$, where $\bar{D}_1$ is the mean of all measured data for the first seasonal cycle of the series.
- $T_0 = \frac{\bar{D}_1 - \bar{D}_2}{p}$, where $\bar{D}_2$ is the mean of all measured data for the second seasonal cycle of the series.
- $So_{k-p} = \frac{D_k}{\bar{D}_1}$, for $k = 1, 2, ..., p$; where D_k is the given measure in the instant k from the first seasonal cycle.

In order to improve the identified model for the series of price and load, the optimal parameters were calculated as: $\alpha = 0.00038$, $\gamma = 0.00033$, $\delta = 0.15888$ and $\alpha = 0.0057971$, $\gamma = 0.0006222$, $\delta = 0.0046903$, respectively. The simulation results with these parameters are shown in Fig. 3 and 4.

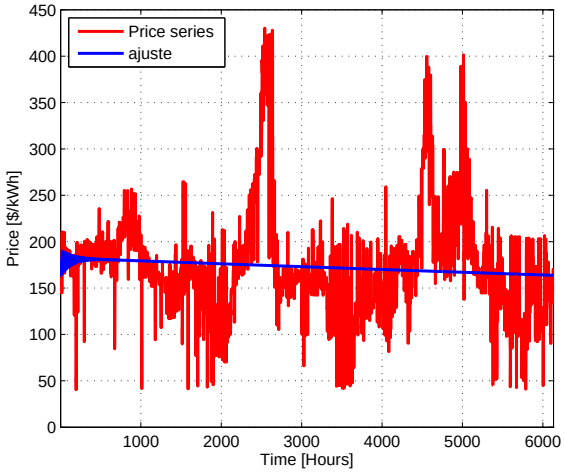


Fig. 3. Optimal adjustment HW model for price series

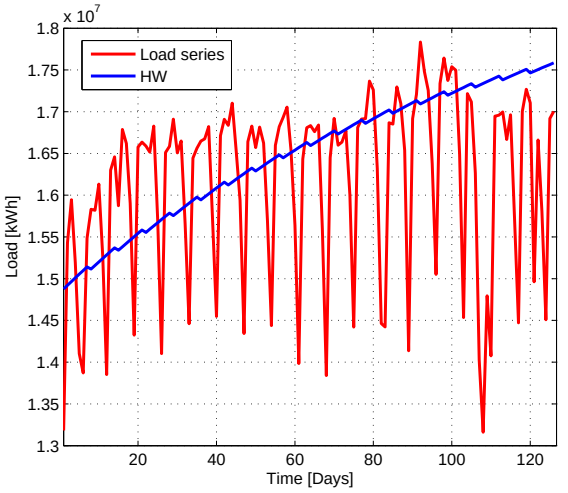


Fig. 4. Optimal adjustment HW model for load series

As can be seen in both figures, the model with $y_d = 0$, does not adjust properly to the series, and since the adjustment was not satisfactory, it was decided to make an adjustment based on parameter optimization to achieve a better result and capture the trend of the series. This model will be used for implementing the proposed estimation methods.

4. MOVING HORIZON ESTIMATOR

The Moving Horizon Estimator (MHE) is an estimation technique based on optimization. In MHE, the problem of state estimation of a given system is solved, considering all previous measurements available in a specific time window. Similarly to the Kalman filter, MHE takes into account the stochastic variables in the model, like the measurement and model noises. Therefore it acts as a complete and optimal estimator (Muske et al., 1993), (Rao

et al., 2001), (Pawlowski et al., 2010). In the literature, several applications of the MHE for dynamic systems are found. This paper explores its use with the HW model.

4.1 Mathematical structure of Moving Horizon Estimator

In order to solve the optimization problem of the MHE, it is necessary to know the matrices associating the current state x_k with the current output y_k, a previous state x_{k-N}, determined by a measurement horizon N. Those relationships are given by the expression (12) and (13) respectively, taken from (Valencia et al., 2011).

$$\tilde{x} = \Phi x_{k-N} + \gamma\tilde{u} + \alpha\tilde{w} \tag{12}$$

$$\tilde{y} = \Gamma x_{k-N} + \Lambda\tilde{u} + \Lambda_\omega\tilde{\omega} \tag{13}$$

where $\tilde{x} \in \mathbb{R}^{Nn_x}$, $\tilde{y} \in \mathbb{R}^{Nn_y}$, $\tilde{u} \in \mathbb{R}^{(N+1)n_x}$, $\tilde{\omega} \in \mathbb{R}^{(N+1)n_w}$ and they are defined in the expressions (14) to (22).

$$\tilde{x} = \left[\, x_{k-N+1} \;\; x_{k-N+2} \;\; x_{k-N+3} \;\; \cdots \;\; x_k \,\right]^T \tag{14}$$

$$\Phi = \left[\, A^T \;\; (A^2)^T \;\; (A^3)^T \;\; \cdots 6(A^N)^T \,\right]^T \tag{15}$$

$$\gamma = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ A^2B & AB & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix} \tag{16}$$

$$\tilde{u} = \left[\, (u_{k-N})^T \;\; (u_{k-N+1})^T \;\; \cdots \;\; (u_{k-1})^T \,\right]^T \tag{17}$$

$$\tilde{\omega} = \left[\, (\omega_{k-N})^T \;\; (\omega_{k-N+1})^T \;\; \cdots \;\; (\omega_{k-1})^T \,\right]^T \tag{18}$$

$$\tilde{y} = \left[\, (y_{k-N+1})^T \;\; (y_{k-N+2})^T \;\; \cdots \;\; (y_k)^T \,\right]^T \tag{19}$$

$$\overline{C} = \begin{bmatrix} C & 0 & 0 & \cdots & 0 \\ 0 & C & \ddots & \cdots & \vdots \\ \vdots & \ddots & C & \ddots & \vdots \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & C \end{bmatrix} \tag{20}$$

$$\Gamma = \overline{C}\Phi \tag{21}$$

$$\Lambda = \overline{C}\gamma \tag{22}$$

where $B_\omega \in \mathbb{R}^{n_w \times n_w}$ is the relationship matrix between the uncertainty modeling with states; likewise if change B with B_ω, can be found $\alpha = \gamma$, $\Lambda = \Lambda_\omega$.

In the state estimation problem, the objective is to minimize errors between the measurement and estimation at every instant of measurement, and likewise minimize modeling errors, as shown in (23).

$$\begin{aligned} \min_{\hat{x}_{k-N},\omega_{k+j}} \; & \sum_{j=-N+1}^{0} \|(y_{k+j} - \hat{y}_{k+j})\|_Q^2 + \|\omega_{k+j}\|_R^2 \\ \text{subject to:} \; & \\ & \hat{x}_{k+j+1} = A\hat{x}_{k+j} + Bu_{k+j} + B_\omega\omega_{k+j} \\ & \hat{y}_{k+j} = C\hat{x}_{k+j} + v_{k+j} \end{aligned} \tag{23}$$

where $y_{k+j} \in \mathbb{R}^{n_y}$ is the measurement data in the instant $k+j$, $\hat{y}_{k+j} \in \mathbb{R}^{n_y}$ is the data estimate in the instant $k+j$, $\hat{x} \in \mathbb{R}^{n_x}$ is the data estimate at a given time, $v_{k+j} \in \mathbb{R}^{n_v}$ are the measurement noises in the instant $k+j$.

In order to simplify the solution of the optimization problem, based on (13), equation (23) can be rewritten as follows:

$$\begin{aligned} \min_{\hat{x}_{k-N},\tilde{\omega}} \; & (\tilde{y} - \hat{\tilde{y}})^T\overline{Q}(\tilde{y} - \hat{\tilde{y}}) + \tilde{\omega}^T\overline{R}\tilde{\omega} \\ \text{subject to:} \; & \hat{\tilde{y}} = \Gamma\hat{x}_{k-N} + \Lambda_\omega\tilde{\omega} \end{aligned} \tag{24}$$

where:

$$\overline{R} = \begin{bmatrix} R & 0 & 0 & \cdots \\ 0 & R & 0 & \cdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & R \end{bmatrix} \in \mathbb{R}^{Nn_v} \tag{25}$$

$$\overline{Q} = \begin{bmatrix} Q & 0 & 0 & \cdots \\ 0 & Q & 0 & \cdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & Q \end{bmatrix} \in \mathbb{R}^{Nn_w} \tag{26}$$

once it is replaced (13) with $\tilde{u} = 0$, in (24), the expression (27) is obtained .

$$\min_{\hat{x}_{k-N},\tilde{\omega}} \quad (\Gamma x_{k-N} + \Lambda_\omega\tilde{\omega} - \hat{\tilde{y}})^T\overline{Q}(\Gamma x_{k-N} + \Lambda_\omega\tilde{\omega} - \hat{\tilde{y}}) + \tilde{\omega}^T\overline{R}\tilde{\omega} \tag{27}$$

The expression (27) can be brought to (28).

$$\min_x \quad \frac{1}{2}x^THx + F^Tx \tag{28}$$

where H, F y x are given by:

$$H = \begin{bmatrix} \Gamma^T\overline{Q}\Gamma & \Gamma^T\overline{Q}\Lambda_\omega \\ \Lambda_\omega^T\overline{Q}\Gamma & \Lambda_\omega^T\overline{Q}\Lambda_\omega + \overline{R} \end{bmatrix} \tag{29}$$

$$F = \left[\, \tilde{y}^T\overline{Q}\Gamma \;\; \tilde{y}^T\overline{Q}\Lambda_\omega \,\right] \tag{30}$$

$$x = \begin{bmatrix} \hat{x}_{k-N} \\ \tilde{\omega} \end{bmatrix} \tag{31}$$

The solution of the optimization problem can be replaced in (12) and (13) to calculate the estimation value of x_k and y_k. In summary, the steps for performing the estimation with MHE are the follows:

(1) Define the estimation window N with N measurement data known.
(2) Define the covariance matrix R and Q.
(3) Define the matrix B_ω, this is usually choose as B Boulkroune et al. (2010).
(4) Compute Γ, Λ, $\overline{Q}$ and $\overline{R}$.
(5) Compute H and F.
(6) Compute $\hat{x}_{k-N}$ and $\tilde{\omega}$ with the optimization solve.
(7) Compute x_k y y_k.

(8) Return to fifth step, to recalculate F, with the next measurement data.

Solving the problem with the described steps generates the $\hat{y}_k$. This value should be compared with the original data series.

4.2 Simulation results

In this section a comparative study between MHE and Kalman filter (KF) with the Holt-Winters model, and the autoregressive methods is presented. The measured variable is the load and price time series described in Section 2, based on this, the objective is to estimate the states of the Holt-Winters model. The noise matrices described in equation (32) are used to stress the methodology performance.

$$\begin{aligned} w &\sim \mathrm{N}(0, Q) \\ Q &= diag(1 \times 10^5) \\ w_d &\sim \mathrm{N}(0, Q_d) \\ Q_d &= 1 \times 10^2 \\ v &\sim \mathrm{N}(0, R) \\ R &= 1 \times 10^{-3} \end{aligned} \tag{32}$$

Given that the original data is noiseless the measurement noise is considered a small value. This implies the assumption where the model noise is bigger that the measurement $R << Q$. The simulation results are shown in Fig.5 to 7.

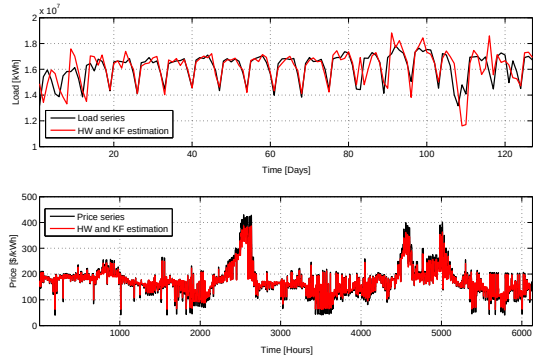


Fig. 5. Holt Winters model with Kalman Filter estimation

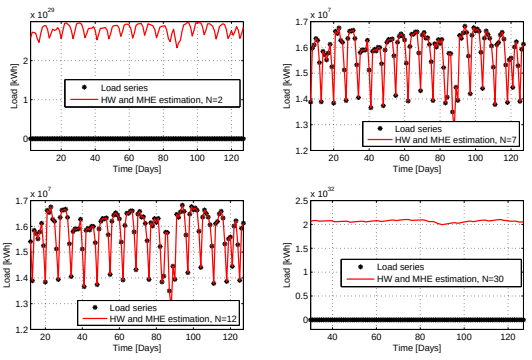


Fig. 6. Holt Winters model with MHE estimation for Load

The obtained results are compared using the mean squared error (MSE) between the estimated series and the original

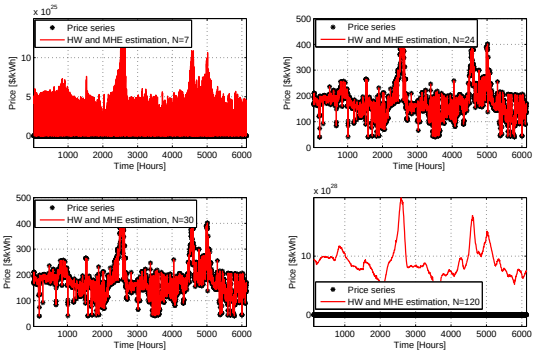


Fig. 7. Holt Winters model with MHE estimation for price

data. The results for all the estimations are shown in Table 2 and 3, in order to validate with traditional methodologies the Kalman Filter and autorregressive methods (ARMA, ARIMA, AR and MA) had been included.

Table 2. MSE for price estimation, with several methods

Estimation series error	
Estimator	**MSE for price**
MHE with $N = 7$	$4,89 \times 10^{25}$
MHE with $N = 24$	$9,15 \times 10^{-14}$
MHE with $N = 50$	$1,04 \times 10^{28}$
MHE with $N = 120$	$9,33 \times 10^{28}$
FK	$30,74$
FKI	$21,14$
MA	$10,82$
AR	$0,9543$
$ARMA$	$3,49 \times 10^{4}$
$ARIMA$	$0,03$

Table 3. MSE for load estimation, with several methods

Estimation series error	
Estimator	**MSE for load**
MHE with $N = 2$	$2,85 \times 10^{29}$
MHE with $N = 7$	$3,76 \times 10^{-8}$
MHE with $N = 12$	539
MHE with $N = 30$	$2,13 \times 10^{32}$
FK	9.06
FKI	5.87
$MA(q)$	$5,46 \times 10^{10}$
$AR(p)$	$31,33 \times 10^{7}$
$ARMA$	$2,27 \times 10^{14}$
$ARIMA$	$0,01$

Among the presented methodologies the Holt Winters with MHE presents the smallest estimation error. This happens when the lenght of the estimation window had similar value to the series seasonality. The lenght of the estimation window had a direct impact on the optimization result. This performance is explained by the model structure and the optimization method used. With the proper seasonal values the model captures all the information of the seasonal series. The MHE can replicate the identified cycle in the estimation of the remaining data series cycles. Changes in the seasonal cycles are adjusted in the next estimation, this clearly depends on the estimation window size getting the minimum error if the window size has exactly the seasonality value, if the windows size is different the correction is not optimal. When the window size is smaller than the

seasonal cycle of the series, the lack of complete seasonal data increases the error. Likewise, as the measurement window is larger than the period of the seasonality the setting error cumulates and the performance decreases. Hence it is best to select for this class of systems a horizon of measurement equal to the seasonality of the series, such that the design of the estimator is consistent with the series properties.

5. CONCLUSIONS

This work successfully presented a comparative study between an optimal estimation methodology and traditional tools in order to estimate energy load and price time series in the Colombian market. Based on a consolidated estimation model as Holt-Winters a state space model was proposed allowing to use optimal estimation methodologies to identify the Holt-Winters model states. The methodology was validated comparing the results with Kalman filter, ARMA, AR, MA and ARIMA models. The MHE methodology achieved the minimal estimation error. It is remarkable that the proposed estimation methodology highly depends of the relationship between the time series seasonality and the measurement window size. The proper selection on the estimation window brings to the estimation methodology the ability to compensated the exogenous variables effects in time series.

ACKNOWLEDGEMENTS

This project was supported by Colciencias with the scholarship given to Julián Rendón as a part of the project "Jovenes investigadores" call number 645 the doctoral scholarship of Pablo Deossa call 528.

REFERENCES

Boulkroune, B., Darouach, M., and Zasadzinski, M. (2010). Moving horizon state estimation for linear discrete-time singular systems. *IET Control Theory and Applications*, 4(3), 339–350. doi:10.1049/iet-cta.2008.0280.

Chatfield, C. and Yar, M. (1988). Holt-Winters Forecasting: Some Practical Issues. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 37(2), 129–140. doi:10.2307/2348687.

Gardner Jr., E.S. (2006). Exponential smoothing: The state of the artPart II. *International Journal of Forecasting*, 22(4), 637–666. doi: 10.1016/j.ijforecast.2006.03.005.

Goodwin, P. (2010). The Holt-Winters Approach to Exponential Smoothing: 50 Years Old and Going Strong. *Foresight: The International Journal of Applied Forecasting*, (19), 30–33.

Haghi, H.V. and Tafreshi, S.M. (2007). An overview and verification of electricity price forecasting models. In *Power Engineering Conference, 2007. IPEC 2007. International*, 724–729. IEEE.

Hyndman, R., Koehler, A.B., Ord, J.K., and Snyder, R.D. (2008). *Forecasting with Exponential Smoothing: The State Space Approach*. Springer Science & Business Media.

Krafft, M. and Mantrala, M.K. (2009). *Retailing in the 21st Century: Current and Future Trends*. Springer Science & Business Media.

Montgomery, D.C., Jennings, C.L., and Kulahci, M. (2015). *Introduction to Time Series Analysis and Forecasting*. Wiley, 2nd edition.

Muske, K., Rawlings, J., and Lee, J.H. (1993). Receding Horizon Recursive State Estimation. In *American Control Conference, 1993*, 900–904.

Ng, S. and Perron, P. (1995). Unit Root Tests in ARMA Models with Data-Dependent Methods for the Selection of the Truncation Lag. *Journal of the American Statistical Association*, 90(429), 268–281. doi: 10.1080/01621459.1995.10476510.

Niimura, T. (2006). Forecasting Techniques for Deregulated Electricity Market Prices - Extended Survey. In *Power Systems Conference and Exposition, 2006. PSCE '06. 2006 IEEE PES*, 51–56. doi: 10.1109/PSCE.2006.296248.

Pawlowski, A., Guzman, J., Rodrguez, F., Berenguel, M., and Snchez, J. (2010). Application of time-series methods to disturbance estimation in predictive control problems. In *2010 IEEE International Symposium on Industrial Electronics (ISIE)*, 409–414. doi: 10.1109/ISIE.2010.5637867.

Ramírez, A. (2013). *Métodos utilizados para el pronóstico de demanda de energía eléctrica en sistemas de distribución*. Thesis, Univerdida Tecnolgica de Pereira.

Rao, C.V., Rawlings, J.B., and Lee, J.H. (2001). Constrained linear state estimationa moving horizon approach. *Automatica*, 37(10), 1619–1628. doi: 10.1016/S0005-1098(01)00115-7.

Segura, J.V. and Vercher, E. (2001). A spreadsheet modeling approach to the HoltWinters optimal forecasting. *European Journal of Operational Research*, 131(2), 375–388. doi:10.1016/S0377-2217(00)00062-X.

Singh, A., Ibraheem, I., Khatoon, S., Muazzam, M., and Chaturvedi, D. (2012). Load forecasting techniques and methodologies: A review. In *2012 2nd International Conference on Power, Control and Embedded Systems (ICPCES)*, 1–10. doi:10.1109/ICPCES.2012.6508132.

Snchez, P. and Velsquez, J.D. (2010). Problemas de investigacin en la prediccin de series de tiempo con redes neuronales artificiales.

Suganthi, L. and Samuel, A.A. (2012). Energy models for demand forecastingA review. *Renewable and Sustainable Energy Reviews*, 16(2), 1223–1240. doi: 10.1016/j.rser.2011.08.014.

Valencia, F., Lopez, J., Marquez, A., and Espinosa, J. (2011). Moving horizon estimator for measurement delay compensation in model predictive control schemes. In *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, 6678–6683. doi:10.1109/CDC.2011.6161346.

Weron, R. (2014a). Electricity price forecasting: A review of the state-of-the-art with a look into the future. *International Journal of Forecasting*, 30(4), 1030–1081. doi:10.1016/j.ijforecast.2014.08.008.

Weron, R. (2014b). A review of electricity price forecasting: The past, the present and the future. HSC Research Reports HSC/14/02, Hugo Steinhaus Center, Wroclaw University of Technology.

XM (2016). Intelligent Information Management. URL `http://informacioninteligente10.xm.com.co`.

Evolutive Extension: A biological approach to heuristic algorithms ⋆

A. Pérez-González, O. Begovich, J. Ruiz-León

CINVESTAV-IPN Unidad Guadalajara Zapopan, Jalisco, Mexico.
Email: [aperez, obegovi, jruiz]@gdl.cinvestav.mx

Abstract: In this paper, we propose an extended form of bio-inspired heuristic algorithm, called Evolutive Extension (EvE) in this work, based on the evolutive approach. EvE is an optimization algorithm which considers several nature examples to construct a list of norms and procedures in search to obtain a better performance than other heuristic algorithms. This method, as many similar ones, is focused on the task of finding an optimum solution to a minimization problem. Using as a base method the Differential Evolution algorithm (DE), EvE includes the abstraction of natural phenomena like elitism, extinction, symbiosis, parasitism, and different forms of reproduction. The study of performance is carried out by a statistical analysis of the results obtained from several executions over a list of typical test functions. The performance of the algorithm and its comparison against the performance of the original form of Differential Evolution is shown. This investigation includes the analysis of the calibration parameters for EvE, also seen from a statistical approach. The presented results show a better performance for EvE algorithm, due to its low number of iterations required to achieve the minimum for each of the test functions proved in this paper. Also, for EvE algorithm it is possible to define a region in which one lies the calibration parameters that allows to the algorithm to solve all the presented tests functions in the less number of iterations, which gives an easy reference to tune up the algorithm.

Keywords: Evolutive Extension, Differential Evolution, Optimization, Heuristic Method, Bio-inspired Algorithm.

1. INTRODUCTION

Heuristic methods take significant relevance, due to their ability to handle a great number of engineering problems that do not have an analytic solution, or the solution of the problem is too complex to obtain (Cooper (1964)), (Pearl (1984)). The heuristic methods offer a good solution if we consider that not always the optimum solution is reachable by the algorithm, but the solution found is acceptable, and the time required to compute it can be considerably low.

A common approach for heuristic methods is the bio-inspired trend: Genetic Algorithms: Houck et al. (1995), Evolutive Computing: Eiben and Schoenauer (2002), social behavior-based algorithms (like Ant Colony: Bilchev and Parmee (1995), or PSO: Pérez-González et al. (2014)), Evolutive Methods: Mühlenbein et al. (1988), etc. This investigation selects the evolutive methods as base of construction, due to the easy and universal understanding of the behavior of life as a constant mechanism of adaptation and optimization. Evolution represents the nature's way to solve a problem. This perspective motivates the attempts to replicate the behavior of evolution in a computational algorithm, where a problem represents the natural environment, and each possible solution is a member of a population in constant fight to survive.

Evolutive methods are based on populations of solutions. Thus, for each iteration of an evolutive algorithm, there is not a unique solution, but a set of them. These methods are based on generating, selecting, combining and replacing a set of solutions. Due to the maintenance of a group instead of a unique solution throughout the entire search process, evolutive methods usually present higher computing times than another heuristic methods (Cormen et al. (2001)).

In an evolutive method, the evolution process operates over the chromosomes, and not directly over the individuals. This allows a "natural selection" procedure, in which the chromosomes with better properties are most often reproduced. In this reproduction process, evolution takes place by means of the combination of the chromosomes of the parents. This process is called recombination. Besides the recombination process, mutations must be considered as a way to alter the chromosomes of any member of the population. Also, an evolutive algorithm is considered as a "no-memory" method, because in the process of formation of chromosomes, only information from the previous period is considered (Bäck (1996)).

An evolutive algorithm integrates and implements efficiently two fundamental ideas: simple representations for the solutions of the problem and simple

⋆ The research of J. Ruiz-León was performed during his sabbatical stay at the Institute of Information Theory and Automation, Czech Academy of Sciences, supported by CONACyT, Mexico, project 265475.

transformations to modify and improve these representations (Martı (2003)). To put in practice this scheme in a computational algorithm, it is necessary: to specify a chromosomic representation, where the number of chromosomes is proportional to the size of the problem; an initial population built in this representation; and a stop criterion based on a finite number of iterations or in an error bound. Also, an evaluation measure is necessary to establish a selection-elimination criterion for chromosomes, and finally it is needed to define one or more recombination rules, and one or more mutation rules (Fleetwood (2004)).

As we mentioned before, there are many optimization algorithms in the bio-inspired approach, and many of them use the evolutive structure. Evolutive algorithms commonly use a binary codification to represent the chromosomic form for each candidate solution in the search space of a problem. This binary representation simplifies the execution of tasks as recombination and mutation, because it is only necessary to replace some bits into the complete chain to perform those instructions, but it is also necessary to translate the original form of the search space to a binary representation, and backwards to the original form after all the evolutive operations (Schmitt (2001)). Only the translation cost can be expensive for some applications, where it is preferable to work directly over the original search space.

Differential Evolution (DE) is a bio-inspired algorithm that deals with that situation, when the search space is a subset of $\Re^n$. Using a set of simple arithmetic rules, DE deals with the mutation and recombination tasks, showing a great capacity to solve problems of high complexity (Storn and Price (1997)).

The aim of this work, is to propose an extension of the DE algorithm, the Evolutive Extension (EvE) algorithm. EvE is based on the inclusion of operations than can represent some natural phenomena trying to improve the searching capacity of the original algorithm. These new rules are introduced in order to modulate the capacity of greedy search, multiple initialization, premature convergence and regional search, as characteristics that define the performance of a heuristic algorithm, according to Storn and Price (1995).

The main contribution of this work is the proposal of the EvE algorithm itself. Of course, it is necessary to include not only the structure of the algorithm, but also its performance, comparing it with the original Differential Evolution algorithm, to show the superiority of EvE. Also, this work shows a statistical approximation to the problem of calibration for EvE. The structure of this paper is the following: Section 2 presents the original DE algorithm; Section 3 presents the EvE algorithm; Section 4 presents the test functions used in this paper; and Section 5 presents and discusses the results of DE and EvE solving the minimization problem of each test function. Finally, we offer some conclusions about all the implementation.

2. DIFFERENTIAL EVOLUTION ALGORITHM

Differential Evolution is a heuristic algorithm based on the survival of a population into an environment that test the superiority of each member, forcing them to compete between all of them (Price et al. (2006)). Let PRN be the number of individuals PR_p^g that conforms a population, and let VPR_p^g be the p-th mutant vector with $p = 1, ..., PRN$, from the g-th generation of vectors. Then, the recombination process is defined by:

$$U_{p,m}^g = \begin{cases} VPR_{p,m}^g & if\ rand(0,1) < C_r \\ & or\ m = I_{rand}, \\ PR_{p,m}^g & otherwise, \end{cases} \quad (1)$$

where $U_{p,m}^g$ is the m-th element of the p-th trial vector U from the g-th generation, C_r is a recombination crossover, which directly modulates how the combination between the original parent PR and the mutant VPR happens, and I_{rand} is an aleatory positive integer in the range $[1, M]$ with $m = 1, ..., M$ (where M is the dimension of the search space), and it allows to guarantee that $U_p^g \neq PR_p^g$ while there exist enough genetic diversity within the population.

In the current literature, several variations of DE algorithms are presented. Such variations are defined by the expression that allows the calculation of the mutant vectors VPR_p^g (Storn and Price (1995)). The most common and generalized form according to Das and Suganthan (2011), is the random/1 form, calculated as follows:

$$VPR_p^g = PR_{r1}^g + F(PR_{r2}^g - PR_{r3}^g), \quad (2)$$

where $r1 \neq r2 \neq r3$ are aleatory positive integers uniformly distributed in the range $[1, PRN]$. F, called mutation factor, is a constant to fix in the interval $[0, 2]$ (Fleetwood (2004)), and it regulates the capability of exploration (near to 2) and exploitation (near to 0) for the DE algorithm.

A selection process is carried out based on the performance of each recombined individual, by the expression:

$$PR_p^{g+1} = \begin{cases} U_p^g & if\ f(U_p^g) < f(PR_p^g), \\ PR_p^g & otherwise, \end{cases} \quad (3)$$

where $f(\ )$ represents an evaluation function, or cost function that maps from the search space $\Re^n$ to $\Re$. Also, for each iteration, it is convenient to include a global superiority memory with the form:

$$Gbest = \begin{cases} U_p^g & if\ f(U_p^g) < f(Gbest), \\ Gbest & otherwise, \end{cases} \quad (4)$$

such that, at the end of the execution of DE, *Gbest* is presented as the best solution found.

3. EVOLUTIVE EXTENSION ALGORITHM

DE algorithm takes directly a set of candidate solutions from the search space, and operates them to improve the cost of each one in search of solving an optimization problem. As we mentioned before, this evolutive process uses mutation and recombination in a simple set of arithmetic rules. But is it possible to improve the searching capacity of the algorithm? There exist another natural behavior than can be expressed as a simple arithmetic rule in search of obtaining better and/or faster results? EvE algorithm tries to implement an answer to those questions, considering basic evolutive behaviors inspired in biology, such as elitism, extinction, symbiosis, parasitism and different forms of reproduction. Elitism and extinction

have been considered in another bio-inspired algorithms, but never in a scheme as complete as EvE.

Elitism: Our first approach is the elitism. In nature, elitism represents the recognition of the best members of a population, propitiating their survival and reproduction. Computationally speaking, elitism represents the addition of certain memory to an algorithm. A memory driven by attributes has an effective and subtle effect in the search, allowing the identification of an attribute than represents a set of possible solutions. Thus, an already identified attribute can prevent that a non-revised candidate solution be accessed, due that it contains the already exploited attribute, or, on the other hand, the identified attribute can redirect the search to non-revised candidate solutions which have the desired attribute. EvE takes into consideration two kinds of memory:

- A short term memory, where the attributes of recently visited solutions are stored. This memory allows to perform intense exploration over specific region of the search space.
- A long term memory, where the frequencies of the occurrence of attributes are stored, seeking to identify the regions in the search space defined by the attributes. Long term memory also allows to diversify the search orientation.

Let X be the search space and $N(x)$ the selection of each one of the PRN individuals that forms the EvE's population. Then:

$$E_n(x) = N_n^*(x) \setminus PR_i, \tag{5}$$

represents the association of each PR with a previously declared attribute, after checking its compliance. That is, the equivalence class where each PR is associated with the compliance of an attribute. Then, EvE uses two attributes to perform its elitism memory:

$$\begin{aligned} \exists PR_i^g, \varepsilon, \gamma : \|PR_i^g - PR_i^{g+1}\| < \varepsilon, \\ |f(PR_i^g) - f(PR_i^{g+1})| > \gamma, \end{aligned} \tag{6}$$

and

$$\begin{aligned} \exists PR_i^g, \delta, \sigma : \|PR_i^g - PR_i^{g+1}\| > \delta, \\ |f(PR_i^g) - f(PR_i^{g+1})| < \sigma. \end{aligned} \tag{7}$$

Expression 6 defines a significant cost change region, in which a small modification in an individual causes a high change in its evaluation cost, so, the search can be diversified allowing a great variety of evaluation costs; whereas expression 7 defines a significant step region, in which it is necessary to perform great modifications to the individuals in order to obtain a small change in its evaluation cost: in this region, the search can be intensified by the steps required by each individual to perform an improvement. For each individual, 6 and 7 represent mutually exclusive attributes.

Extinction: Our second proposed rule is extinction. In nature, extinction is the total elimination of the members of a population, due to their incapacity to adapt to the environment. For EvE algorithm, it is necessary to compensate the greedy characteristic that produces premature convergence in the population. Let $E_x = 0$ represent a counter, then:

$$E_x = \begin{cases} E_x + 1 \; if \; \|Gbest - PR_i^g\| = 0, \\ E_x \qquad\qquad\qquad otherwise. \end{cases} \tag{8}$$

E_x increases every time a member of the population replicates the best individual found at the moment. When E_x surpass an extinction bound, all the members of the population are reinitialized, and the chromosomes of the best solution found are reinserted in the population, in a random index:

$$PR_{randi(PRN)}^{0_n} = Gbest^{g_{n-1}}, \tag{9}$$

where the super index 0_n represents the generation in which the population has been reinitialized.

Reproduction: For the standard DE algorithm, reproduction only happens by the recombination of the chromosomes of a parent and a mutant generated under the same index. EvE algorithm recombines diversified individuals in search of generating a more aggressive attitude, covering faster the search space. The first reproduction model is the multipoint crossing. Let PR_i^g and PR_j^g be two members of the population, which belong to the same region (expression 6 or 7), then, they can generate two new members by:

$$\begin{aligned} PR_{(i|j),m_{mod2=0}}^{g+1_a} &= PR_{i,m}^g, \\ PR_{(i|j),m_{mod2\neq 0}}^{g+1_a} &= PR_{j,m}^g, \end{aligned} \tag{10}$$

$$\begin{aligned} PR_{(i|j),m_{mod2=0}}^{g+1_b} &= PR_{j,m}^g, \\ PR_{(i|j),m_{mod2\neq 0}}^{g+1_b} &= PR_{i,m}^g. \end{aligned} \tag{11}$$

Expression 10 represents the formation of one of two children, the (a) child, in which the odd chromosomes are taken from the parent with index j, and the even chromosomes are taken from the parent with the index i. The expression 11 represents the formation of the second child (b), whose chromosomes are taken in the contrary way. We also use the index $(i|j)$ to represent that each one of the children can take the index of any of the two parents, only if they improve their evaluation cost, that is, if child (a) or (b) surpass both parents, it takes the index of the parent with the worst performance; if one of the children surpass the performance of only one parent, it takes its index; if both children surpass both parents, child (a) takes the i index, and child (b) takes the j index.

Using the elitist memory of EvE algorithm, it is possible to establish an attraction between two members of the population. Let PR_i^g and PR_j^g be two individuals with a common attribute, but with $\|PR_i^g - PR_j^g\| > \varepsilon$, where ε is a fixed constant that represents at least $\|UB - LB\|/2$ (UB and LB are the upper and lower bounds of each dimension of the search space, respectively). Then:

$$PR_{(i|j),m}^{g+1} = \frac{1}{2} PR_{i,m}^g + \frac{1}{2} PR_{j,m}^g, \tag{12}$$

where, the individual with the worst evaluation cost is eliminated from the population, and, as in the previous reproduction model, the index $(i|j)$ represents that the child takes the index of the parent with the worst performance, if the mentioned child is not the individual with the worst performance.

Parasitism: Another natural phenomena considered in the EvE algorithm is parasitism. In nature, parasitism is the survival of an individual at the expense of a second one.

In the EvE algorithm, we represent that idea giving to a member of the population the capacity to mimic another one of better performance, in search of guaranteeing its survival. Let PR_i^g and PR_j^g be two individuals of EvE's population, and there exists recognition of PR_i^g's good performance, and PR_j^g has the parasite property. Then:

$$PR_{j,m}^{g+1} = (1-\rho)PR_{i,m}^g + (\rho)PR_{j,m}^g, \tag{13}$$

where ρ is a constant in the range $[0,1]$, and defines an "infection ratio". Parasitism helps the EvE algorithm to assign good properties to individuals with bad performance, which allows to perform a more aggressive search.

Symbiosis: The final nature example used to create EvE algorithm is the symbiosis phenomena. In nature, symbiosis establishes a relationship of mutual benefit. Let PR_i^g be an individual with symbiotic attitude, then, its mutation model takes the form:

$$\begin{aligned} VPR_i^g = PR_i^g + F(Gbest - PR_i^g) \\ + F(PR_{r1}^g - PR_{r2}^g), \end{aligned} \tag{14}$$

which simulates a mutual benefit relation between PR_i^g and *Gbest*. The first one guarantees that its children contain information of the individual of best performance, and the second one ensures its reproduction. In Equation 14, $r1 \neq r2$ are random integers uniformly distributed in the range $[1, PRN]$ (Qin and Suganthan (2005)).

4. TEST FUNCTIONS

The study of the performance of a heuristic algorithm is commonly realized through the data mining from several executions over a set of test functions. According to Price et al. (2006), it is possible to classify those test functions in three kinds:

- Unimodal functions: Only contain a global minimum, with no local minimums.
- Unconstrained multimodal functions: Present at least two local minimums, but their global minimum does not depend of the domain of the function.
- Constrained multimodal functions: There is no global minimum defined for these functions. The minimum evaluation value depends of the domain of the function.

For this paper, we use the test functions presented in Tables 1, 2 and 3.

Table 1. Unimodal Test Functions

Sphere:
$f(x) = \sum_{j=1}^{N} x_j^2,$
$-10 \leq x_j \leq 10,\ j = 1, ..., N,\ f(x^*) = 0, x^* = 0.$
Generalized Rosenbrock:
$f(x) = \sum_{j=1}^{N-1} (100(x_{j+1} - x_j^2)^2 + (x_j - 1)^2),$
$-20 \leq x_j \leq 20,\ j = 1, ..., N,\ f(x^*) = 0, x^* = 1.$
Schwefels Ridge:
$f(x) = \sum_{k=1}^{N} ((\sum_{j=1}^{k} x_j)^2),$
$-50 \leq x_j \leq 50,\ j = 1, ..., N,\ f(x^*) = 0, x^* = 0.$

For Tables 1, 2 and 3, x^* represents the value in which the test function has its minimum evaluation cost, $f(x^*)$.

5. EXECUTION AND RESULTS

In order to solve each of the test functions presented in the last section, it is necessary to specify all the execution conditions. First, for each test function, $N = 10$, that is, $x \in \Re^{10}$. Also, for all x^* values, we tolerate an error of $e = \pm 10^{-6}$. Finally, for EvE algorithm it is necessary to establish some programing factors. The extinction bound is set as $3PRN$ trying to guarantee that the extinction phenomena will not be triggered by a population with a low genomic variety, and the infection ratio is 0.25, as a fixed small step for the parasitism phenomena that allows that the parasite does not duplicate the parasited individual. In practice, those parameters show good results, avoiding excessive reinitialization or premature convergence. In order to compare the original DE algorithm against EvE, we use the same number of individuals in both of them, $PRN = 50$.

The first approach is to let the calibration parameters for DE and EvE be selected in a random way, thus $C_r \in [0,1]$ and $F \in [0,2]$, in a uniform distribution. That is, we want to know for which calibration parameters each algorithm requires the less number of iterations to solve each of the minimization problems. Then, we perform 1000 executions for each algorithm, trying to cover most of the $C_r \times F$ plane, and every time, C_r and F are randomly chosen.

Then, for DE and EvE, we can identify a "comfort zone", CZ, as the $C_r \times F$ region where a test function is minimized with the less number of iterations. Figure 1 and Figure 2 show an example of the comfort zone.

Table 2. Unconstrained Multimodal Test Functions

Ackley:
$f(x) = -20e^{-0.2\sqrt{\frac{1}{N}\sum_{j=1}^{N} x_j^2}} - e^{\frac{1}{N}\sum_{j=1}^{N} \cos(2\pi x_j)} + 20 + e,$
$-30 \leq x_j \leq 30,\ j = 1, ..., N,\ f(x^*) = 0, x^* = 0.$
Rastrigin:
$f(x) = \sum_{j=1}^{N} (x_j^2 - 10\cos(2\pi x_j) + 10),$
$-6 \leq x_j \leq 6,\ j = 1, ..., N,\ f(x^*) = 0, x^* = 0.$
Salomon:
$f(x) = -\cos(2\pi\|x\|) + 0.1\|x\| + 1,\ \|x\| = \sqrt{\sum_{j=1}^{N} x_j^2},$
$-50 \leq x_j \leq 50,\ j = 1, ..., N,\ f(x^*) = 0, x^* = 0.$

Table 3. Constrained Multimodal Test Functions

Schwefel:
$f(x) = -\frac{1}{N}\sum_{j=1}^{N} x_j \sin(\sqrt{\lvert x_j \rvert}),$
$-500 \leq x_j \leq 500,\ j = 1, ..., N,\ f(x^*) = -418.983, x^* = 420.9687.$
Rana:
$f(x) = \sum_{j=1}^{N-1} x_j \sin(\alpha)\cos(\beta) + x_{(j+1) mod N} \cos(\alpha)\sin(\beta),$
$\alpha = \sqrt{\lvert x_{j+1} + 1 - x_j \rvert},\ \beta = \sqrt{\lvert x_{j+1} + 1 + x_j \rvert},$
$-512 \leq x_j \leq 512,\ j = 1, ..., N,\ f(x^*) = -511.708, x^* = -512.$

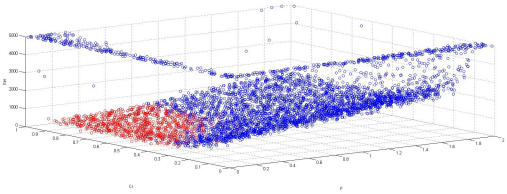


Figure 1. Comfort zone, in red, for EvE in the sphere function

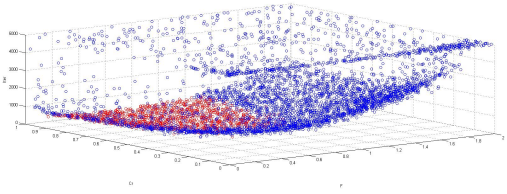


Figure 2. Comfort zone, in red, for DE in the sphere function

The z axis in Figures 1 and 2 represent the number of iterations required to solve the problem. As we can see, both algorithms can solve the sphere function of $\Re^{10}$. The red region represents the comfort zone, in which the problem is solved in less than 500 iterations. It is easy to see that, for EvE algorithm, the comfort zone is more clearly defined that for DE algorithm.

The process is as follows: after 1000 executions for each algorithm in each function, a vector that storages the number of iterations required to achieve the minimum evaluation value is read, and if the number of iterations is at most 500, the index of that stored value is used to access to the C_r and F values that allow that performance. Then, the minimum and maximum values of C_r and F that have the "at most 500 iterations" property defines the CZ of DE and EvE. To avoid to take in consideration some minimum or maximum at the extremes of the distribution, it is necessary to discard the calculated value if it is isolated, that is, if this value has no at least another point with the property of "at most 500 iterations" in a neighborhood of 0.01, in order to guarantee that there is at least two (C_r, F) coordinated pairs in the CZ bounds. Using the CZ of each algorithm, 1000 new executions are performed. The success ratio is computed by the number of success with less than 500 iterations over the number of executions.

Table 4 summarizes the results of DE and EvE algorithm for the Unimodal test functions.

Table 4. Results for Unimodal Test Functions

Function	DE CZ	EvE CZ	Success ratio
Sphere	$0.23 \leq C_r \leq 0.97$	$0.31 \leq C_r \leq 1$	$P(DE) = 0.841$
	$0.11 \leq F \leq 0.61$	$0.13 \leq F \leq 0.42$	$P(EvE) = 0.997$
Generalized	$0.55 \leq C_r \leq 0.95$	$0.53 C_r 0.95$	$P(DE) = 0.712$
Rosenbrock	$0.10 \leq F \leq 0.81$	$0.13 \leq F \leq 0.51$	$P(EvE) = 0.991$
Schwefels	$0.32 \leq C_r \leq 0.97$	$0.28 \leq C_r \leq 0.99$	$P(DE) = 0.883$
Ridge	$0.13 \leq F \leq 0.78$	$0.13 \leq F \leq 0.87$	$P(EvE) = 0.998$

As we can see in Table 4, Unimodal test functions do not represent a high challenge, neither for DE nor for EvE. Table 5 presents the results for the Unconstrained Multimodal functions, and Figure 3 and Figure 4 show an example of the CZ of each algorithm in this kind of functions.

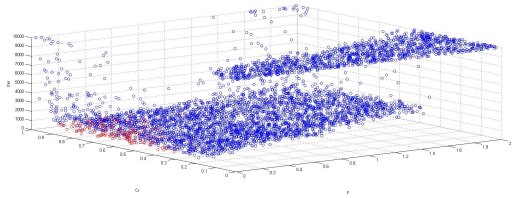


Figure 3. Comfort zone, in red, for EvE in the Rastrigin function

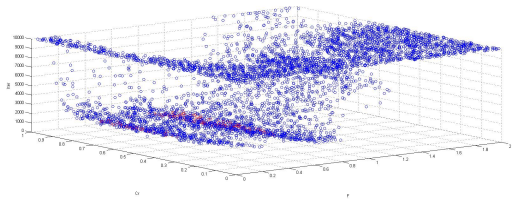


Figure 4. Comfort zone, in red, for DE in the Rastrigin function

Table 5. Results for Unconstrained Multimodal Test Functions

Function	DE CZ	EvE CZ	Success ratio
Ackley	$0.66 \leq C_r \leq 1$	$0.55 \leq C_r \leq 1$	$P(DE) = 0.641$
	$0.21 \leq F \leq 0.48$	$0.13 \leq F \leq 0.26$	$P(EvE) = 0.887$
Rastrigin	$0.36 \leq C_r \leq 0.86$	$0.50 \leq C_r \leq 0.96$	$P(DE) = 0.744$
	$0.78 \leq F \leq 0.96$	$0.11 \leq F \leq 0.28$	$P(EvE) = 0.903$
Salomon	$0.44 \leq C_r \leq 1$	$0.54 \leq C_r \leq 0.98$	$P(DE) = 0.573$
	$0.63 \leq F \leq 1.12$	$0.18 \leq F \leq 0.76$	$P(EvE) = 0.781$

Table 5 still exhibits the superiority of EvE over DE. Although the success ratio is considerably low for Unconstrained Multimodal functions in comparison with the success ratio achieved in Unimodal functions, it is important to remember that we only consider as a success the achievement of the minimum in less than 500 iterations.

Finally, Table 6 summarizes the results for the most difficult kind of test functions used in this paper, the Constrained Multimodal functions. Also, Figures 5 and 6 present an example of the CZ for these functions.

Table 6. Results for Constrained Multimodal Test Functions

Function	DE CZ	EvE CZ	Success ratio
Schwefel	$0.40 \leq C_r \leq 0.99$	$0.35 \leq C_r \leq 1$	$P(DE) = 0.693$
	$0.36 \leq F \leq 0.81$	$0.12 \leq F \leq 0.32$	$P(EvE) = 0.910$
Rana	$0.32 \leq C_r \leq 0.99$	$0.33 \leq C_r \leq 1$	$P(DE) = 0.469$
	$0.38 \leq F \leq 1.72$	$0.13 \leq F \leq 1.05$	$P(EvE) = 0.687$

For all the three kinds of test functions considered in this paper, EvE algorithm shows a superiority in the capacity to achieve the minimum evaluation value in less iterations. Also, it is important to note that no execution has success ratio of 1, and that is intrinsically a heuristic behavior, there exist random components in both algorithms that cause to mismatch some parts of the execution, like the initialization step, or the selection of the components of the mutation equations. Nevertheless, the superiority of EvE algorithm surpass with 0.218 points the performance of DE in the worst EvE case. For the better DE case, EvE has a superior performance with 0.115 points of success ratio over the DE's performance.

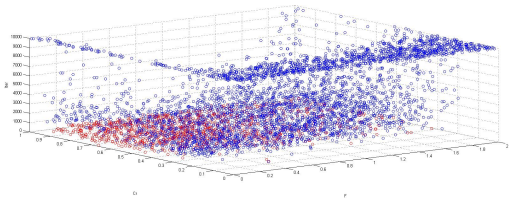


Figure 5. Comfort zone, in red, for EvE in the Rana function

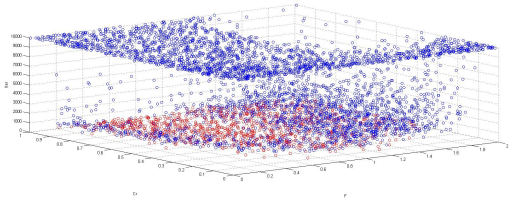


Figure 6. Comfort zone, in red, for DE in the Rana function

6. CONCLUSIONS

In this paper, a new bioinspired heuristic algorithm is presented, and its performance is discussed. Using the evolutionary approach, and the Differential Evolution algorithm as a base of construction, we take some natural processes as inspiration to construct a set of operations that improve the performance of the EvE algorithm, from the extinction process to avoid the premature convergence behavior, to the parasitism phenomena, that allows to explore the search space using a convex combination. As a competition scheme, we take several classic test functions and run over them the EvE algorithm and the original DE algorithm to make a point of comparison. In all the cases, EvE has a better performance than DE, but it is important to mention that the execution cost is higher for EvE, than for DE, this is obviously due to the set of rules adding to extend the algorithm. Nevertheless, this cost is not far above than DEs cost, thanks to the use of a programing scheme of one rule at a time, that is, if an individual is recombined, it is not candidate anymore for a multipoint crossing procedure, and so on, until the end of each generation.

Another important characteristic that arises from our results is the definition of the CZ. It is easy to see that for DE algorithm, CZ is not clearly defined for all the test function, whereas for EvE, we can conclude that a CZ of $0.55 \leq C_r \leq 0.95$ and $0.18 \leq F \leq 0.26$ covers all our test functions, providing EvE a better generalization property.

As future work, EvE will be compared against another heuristic algorithms. Also, it will be necessary to formally define its execution cost. And, at the same time, it is proposed as future work the formal explanation of every EvE rule and process (remembering that this is an introductory paper); as well as the addition of new rules that may improve its performance.

ACKNOWLEDGEMENTS

We sincerely thank to CONACyT for the economic incentives provided during this investigation and to CINVESTAV for the provided resources.

REFERENCES

Bäck, T. (1996). *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press.

Bilchev, G. and Parmee, I.C. (1995). The ant colony metaphor for searching continuous design spaces. In *Evolutionary Computing*, 25–39. Springer.

Cooper, L. (1964). Heuristic methods for location-allocation problems. *Siam Review*, 6(1), 37–53.

Cormen, T.H., Leiserson, C.E., Rivest, R.L., and Stein, C. (2001). *Introduction to algorithms*, volume 6. MIT press Cambridge.

Das, S. and Suganthan, P.N. (2011). Differential evolution: a survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 15(1), 4–31.

Eiben, A.E. and Schoenauer, M. (2002). Evolutionary computing. *Information Processing Letters*, 82(1), 1–6.

Fleetwood, K. (2004). An introduction to differential evolution. In *Proceedings of Mathematics and Statistics of Complex Systems (MASCOS) One Day Symposium, 26th November, Brisbane, Australia*.

Houck, C.R., Joines, J., and Kay, M.G. (1995). A genetic algorithm for function optimization: a matlab implementation. *NCSU-IE TR*, 95(09).

Martı, R. (2003). Procedimientos metaheurısticos en optimización combinatoria. *Matemátiques, Universidad de Valencia*, 1(1), 3–62.

Mühlenbein, H., Gorges-Schleuter, M., and Krämer, O. (1988). Evolution algorithms in combinatorial optimization. *Parallel Computing*, 7(1), 65–85.

Pearl, J. (1984). *Heuristics: intelligent search strategies for computer problem solving*. Addison-Wesley Pub. Co., Inc., Reading, MA.

Pérez-González, A., Begovich, O., and Ruiz-Leon, J. (2014). Modeling of a greenhouse prototype using PSO algorithm based on a LabView application. In *11th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, 1–6. IEEE.

Price, K., Storn, R.M., and Lampinen, J.A. (2006). *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media.

Qin, A.K. and Suganthan, P.N. (2005). Self-adaptive differential evolution algorithm for numerical optimization. In *IEEE Congress on Evolutionary Computation*, volume 2, 1785–1791. IEEE.

Schmitt, L.M. (2001). Theory of genetic algorithms. *Theoretical Computer Science*, 259(1), 1–61.

Storn, R. and Price, K. (1995). *Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces*, volume 3. ICSI Berkeley.

Storn, R. and Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4), 341–359.

Fixed-Time Convergent Unknown Input Observer for LTI Systems

Juan G. Rueda-Escobedo * **Jaime A. Moreno** *
Pablo Oliva-Fonseca *

* *Eléctrica y Computación, Instituto de Ingeniería, Universidad Nacional Autónoma de México, 04510 México D.F., México (e-mails: JRuedaE@iingen.unam.mx; JMorenoP@ii.unam.mx; POlivaF@iingen.unam.mx).*

Abstract: In this work, a family of non linear observers for linear time invariant systems is presented. The algorithms have the capability of providing an estimate of the system internal state in finite time. The observer converges in fixed-time since it exists an upper bound for the convergence time which is independent of the initial error. This note also provides a methodology to apply the proposed algorithms when unknown inputs are present, and how to retain its properties. Simulation examples are provided to contrast the behavior of the nonlinear observer with a classic Luenberger one.

Keywords: Observers, Estimation Algorithms, Finite-Time, Fixed-Time

1. INTRODUCTION

In the field of automatic control, the problem of reconstructing unmeasured variables is almost always present. This task is handled by observers. The classical methods to design this class of algorithms only provide strategies that asymptotically reconstruct the desired variables. However, in recent years, algorithms capable of estimating the internal state of linear time invariant (LTI) systems, and some nonlinear ones, have appeared (Raff and Allgöwer, 2008; Menard et al., 2010; Pin et al., 2013). Nevertheless, these methods required to put the analyzed system in specific coordinates to handle the observer design. An upgrade with respect to finite-time convergent observers are the fixed-time convergent ones (Cruz-Zavala et al., 2011, 2012; Polyakov, 2012). Fix-time means that the convergence time can be upper bounded with independence of the initial error. These algorithms do not only give a theoretical improvement with respect to asymptotic convergence, but also help to separate the effect of the observer from the controller when the last is nonlinear (Cruz-Zavala et al., 2011; Atassi and Khalil, 2000).

Recently, the authors presented in (Rueda-Escobedo et al., 2016) a discontinuous finite-time convergent observer for LTI systems. In this note, the previous result is extended to include continuous versions of the algorithm. This new version not only keeps the finite-time convergence, but provides fixed-time convergence. In contrast to other available methods, the design of the proposed algorithms can be done in the original system's coordinates, and the gains can be chosen using the same techniques required for the design of linear observers.

In real systems it is not uncommon to have uncertainties and perturbations, which can be modeled as unknown inputs. In this situation it is desirable to have an observer capable of reconstructing the internal state of the system despite the presence of the unknown inputs. Commonly in the literature, this kind of observers are called *unknown input observers* (UIO). Sufficient and necessary conditions for the existence of an UIO for a given LTI system are already known, and can be found in (Hautus, 1983). This kind of observers have found application especially in the field of fault detection (Chen and Saif, 2006; Hwang et al., 2010; Yin et al., 2014). Given the importance of this topic, this work also provides a methodology to apply the proposed algorithm to the observation problem in the presence of unknown inputs. It has to be said that under some stronger assumptions than those needed for the existence of an UIO, the proposed observer retains the fixed-time convergence.

The paper organization is as follows; in Section 2 the observation algorithm is presented together with its convergence analysis and properties. In Section 3, a methodology for design UIO for linear systems is explained; this methodology can be used with any observer, but is intended to be used with the one proposed in this work. Finally, in Section 4, the design of the proposed algorithm is illustrated, step by step, for a perturbed linear system.

2. ALGORITHM DESCRIPTION

A general LTI system is considered. The system is described by

$$\begin{aligned} \dot{x} &= A\,x + B\,u(t), \\ y(t) &= C\,x, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^r$, and $y(t) \in \mathbb{R}^q$. Matrices A, B, and C have appropriate dimensions an are assumed to be known. The only available signals are the input $u(t)$ and the output $y(t)$. Additionally, it is required that the pair (A, C) is observable in order to be able of design an observer with arbitrary speed of convergence.

Let $\hat{x}(t)$ be the estimate of $x(t)$. Under the previous assumptions, the following is an observer for system (1):

$$\begin{aligned} \dot{\hat{x}} =& A\,\hat{x} + B\,u(t) - L(C\hat{x} - y(t)) \\ & - P^{-1}N(t)\sum_{i=1}^{2} k_i \lceil N(t)\hat{x} - \psi(t) \rfloor^{p_i}. \end{aligned} \tag{2}$$

The symbol $\lceil\cdot\rfloor^p$ takes the signed p power of its argument in the scalar case and for vectors it is understood element-wise, that is, $\lceil\cdot\rfloor^p = |\cdot|^p \mathrm{sign}(\cdot)$. The exponents p_i have to be chosen such that $p_1 \in [0, 1)$ and $p_2 > 1$, and the gains k_i need only to be positive. Matrix L is designed to make $A - LC$ Hurwitz, and matrix P is computed from the algebraic Lyapunov equation

$$\big(A - LC\big)^\top P + P\big(A - LC\big) = -R,\ R = R^\top > 0.$$

Matrix $N(t)$ and signal $\psi(t)$ are related by the equality $\psi(t) = N(t)x(t)$, and are computed as

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}N =& -\big(A + KC\big)^\top N - N\big(A + KC\big) + C^\top C, \\ N(t_0) =& 0, \end{aligned}$$

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\psi =& -(A + KC)^\top \psi + C^\top y(t) \\ & - N(t)\big(Ky(t) - Bu(t)\big), \\ \psi(t_0) =& 0. \end{aligned}$$

Here, the matrix K has to be proposed such that the matrix $A + KC$ is anti-Hurwitz. Taking the dynamics of $N(t)$, and the derivative of $N(t)x(t)$, the dynamics for $\psi(t)$ is gotten.

Notice that $N(t)$ is the constructibility gramian (Hespanha, 2009, Chap. 15) of system $\dot{\chi} = (A + KC)\chi$ with output $C\chi$. If the pair (A, C) is observable, so is the pair $(A + KC, C)$, since the term $KC\chi$ represents an output injection. Then $N(t)$ acquires full rank, and becomes bounded from below, i.e. $N(t) \geq \alpha(t^\star)\mathbb{I}$ for $t \geq t^\star > t_0$. Since K is chosen to make the matrix $A + KC$ anti-Hurwitz, $N(t)$ will remain bounded (Abou-Kandil et al., 2003, Chap. 1).

The property $\psi(t) = N(t)x(t)$, and the full rank of $N(t)$, makes the nonlinear terms in (2) a direct injection of the estimation error:

$$\lceil N(t)\hat{x} - \psi(t) \rfloor^{p_i} = \lceil N(t)\big(\hat{x} - x\big) \rfloor^{p_i}.$$

Also notice that this terms are only zero if the estimation error is zero, while the linear term in the algorithm becomes zero whenever the estimation error belongs to $\ker(C)$.

This algorithm has the following properties

Theorem 1. Consider system (1) and the algorithm (2). If the pair (A, C) is observable, then for any $t^\star > t_0$ $N(t) \geq \alpha(t^\star)\mathbb{I}$ for $t \geq t^\star$. Fix $t^\star$ to make α constant. Then $\hat{x}(t) \equiv x(t)$ for

$$t \geq t^\star + \frac{\lambda_M^{\frac{p_1+1}{2}}(P) n^{p_1}}{2^{\frac{p_1-1}{2}}(1-p_1)k_1\alpha^{p_1+1}} + \frac{\lambda_M^{\frac{p_2+1}{2}}(P) n^{p_2}}{2^{\frac{p_2-1}{2}}(p_2-1)k_2\alpha^{p_2+1}},$$

where $\lambda_M(\cdot)$ denotes the greatest eigenvalue of its argument. This means that (2) converges in finite time. Since the time needed for the estimate to reach the actual state does not depend on the initial error, it is said that the algorithm converges in fixed-time. △

The larger the value of α, the smaller the time needed to converge. Taking $t^\star$ bigger makes $\alpha(t^\star)$ larger; however, the function $\alpha(t^\star)$ is upper bounded, then there exists a value for $t^\star$ from which α does not increase further. The value of α can be controlled through the design of K. About this, it is recommendable to make the real part of the eigenvalues of $A + KC$ close to zero.

2.1 Convergence Analysis and Proof of Theorem 1

The convergence analysis of (2) is going to be handled through the study of the error dynamics. For this purpose, let us define the estimation error as $e(t) = \hat{x}(t) - x(t)$. Taking the time derivative of $e(t)$ yields

$$\dot{e} = \big(A - LC\big)e - P^{-1}N(t)\sum_{i=1}^{2} k_i \lceil N(t)e \rfloor^{p_i}. \tag{3}$$

If the solution $e(t) = 0$ is attractive and stable then the algorithm converges to $x(t)$. To investigate that matter, take as Lyapunov function candidate $V(e) = \frac{1}{2}e^\top Pe$, and its derivative along the trajectories of (3) satisfies

$$\dot{V}(t) \leq -\frac{1}{2}e^\top(t)R\,e(t) - \sum_{i=1}^{2}\frac{k_i}{n^{p_i}}\|N(t)e(t)\|_1^{p_i+1}, \tag{4}$$

because

$$e^\top(t)N(t)\lceil N(t)e(t) \rfloor^{p_i} \geq \frac{1}{n^{p_i}}\|N(t)e(t)\|_1^{p_i+1}.$$

From (4) it is clear that $e(t) = 0$ is asymptotically stable. Furthermore, due to the observability of system (1), matrix $N(t)$ is bounded from below by $\alpha\mathbb{I}$ for $t \geq t^\star$ for some $t^\star$, then

$$\begin{aligned} \frac{k_i}{n^{p_i}}\|N(t)e(t)\|_1^{p_i+1} \geq \\ \frac{k_i\alpha^{p_i+1}}{n^{p_i}}\left(\frac{2}{\lambda_M(P)}\right)^{\frac{p_i+1}{2}} V^{\frac{p_i+1}{2}}(t). \end{aligned}$$

The last inequality allows us to change (4) into two simultaneous differential inequalities

$$\dot{V}(t) \leq -\frac{k_i \alpha^{p_i+1}}{n^{p_i}} \left(\frac{2}{\lambda_M(P)}\right)^{\frac{p_i+1}{2}} V^{\frac{p_i+1}{2}}(t),$$

with solution

$$V(t) \leq \left(V^{\frac{1-p_i}{2}}(t^\star) - \frac{2^{\frac{p_i-1}{2}}(1-p_i)k_i\alpha^{p_i+1}}{\lambda_M^{\frac{p_i+1}{2}}(P)n^{p_i}}(t-t^\star) \right)^{\frac{2}{1-p_i}}. \tag{5}$$

Notice that $V(t)$ satisfies both inequalities for all $t \geq t^\star$. Between t_0 and $t^\star$, the Lyapunov function does not increase, this is prevented by the quadratic term in its derivative. For the convergence time calculation, it is assumed that $V(t)$ remains constant in the interval $[t_0, t^\star)$.

For $i = 2$, the appropriate inequality is going to be used to estimate the time needed to guarantee $V(t) \leq 1$, whereas the inequality for $i = 1$ will be used to calculate the necessary time for $V(t)$ to decrease from a unitary value to zero.

According to the described process, (5) is rewritten for $i = 2$:

$$t - t^\star \geq \frac{\lambda_M^{\frac{p_2+1}{2}}(P)n^{p_2}}{2^{\frac{p_2-1}{2}}(p_2-1)k_2\alpha^{p_2+1}} \left(1 - \frac{1}{V^{\frac{p_2-1}{2}}(t^\star)}\right).$$

Taking the limit when $V(t^\star) \to \infty$, it is clear that for

$$t \geq t^\star + \frac{\lambda_M^{\frac{p_2+1}{2}}(P)n^{p_2}}{2^{\frac{p_2-1}{2}}(p_2-1)k_2\alpha^{p_2+1}} = t_1$$

$V(t) \leq 1$. Now assume that $V(t_1) = 1$, using inequality (5) with $i = 1$, we proceed to estimate the time needed to reach zero. This happens when the *rhs* of (5) equals zero, then $V(t) \equiv 0$ for

$$t \geq t_1 + \frac{\lambda_M^{\frac{p_1+1}{2}}(P)n^{p_1}}{2^{\frac{p_1-1}{2}}(1-p_1)k_1\alpha^{p_1+1}}.$$

Then the time needed for $V(t)$ to decrease from its initial value to zero is at most

$$t^\star + \frac{\lambda_M^{\frac{p_1+1}{2}}(P)n^{p_1}}{2^{\frac{p_1-1}{2}}(1-p_1)k_1\alpha^{p_1+1}} + \frac{\lambda_M^{\frac{p_2+1}{2}}(P)n^{p_2}}{2^{\frac{p_2-1}{2}}(p_2-1)k_2\alpha^{p_2+1}},$$

which is independent from the initial error. This concludes the proof of Theorem 1.

3. UNKNOWN INPUT OBSERVER DESIGN

In this section, a perturbed linear system will be considered. Take again into consideration system (1), but now with an unknown input $\nu(t) \in \mathbb{R}^q$

$$\begin{aligned} \dot{x} &= A\,x + B\,u(t) + D\,\nu(t), \\ y(t) &= C\,x. \end{aligned} \tag{6}$$

It is assumed that matrix D is also known. Without loss of generality, it is assumed that $\text{rank}(C) = m$, and $\text{rank}(D) = q$.

The existence of an observer capable of estimating x without the knowledge of $\nu(t)$ is conditioned to the system to be strong* detectable (Hautus, 1983). If the system is strong* detectable, then $\lim_{t\to\infty} y(t) = 0$ implies $\lim_{t\to\infty} x(t) = 0$ with independence of the unknown input and the initial state. This property can be tested through the following criteria:

Theorem 2. (Hautus, 1983)(Moreno, 2001) The system (6) is strong* detectable if and only if

$$\text{rank}\begin{bmatrix} s\,\mathbb{I} & -D \\ C & 0 \end{bmatrix} = n + q, \quad \forall\, s \in \mathbb{C}_0^+$$

and

$$\text{rank}(CD) = q,$$

where $\mathbb{C}_0^+$ is the closed right half plane of the complex plane. △

The first condition means that all the transmission zeros of the system have negative real part, and that the system is detectable. The second statement requires that $m \geq q$, in other words, it is required at least the same number of outputs as the unknown inputs.

Suppose the rank condition for CD is met. Define the matrix

$$S = \begin{bmatrix} S_1 \\ S_2 \end{bmatrix},\ S \in \mathbb{R}^{m\times m} \tag{7}$$

with

$S_1 = (C\,D)^+ = (D^\top C^\top C\,D)^{-1} D^\top C^\top,\ S_1 \in \mathbb{R}^{q\times m}$,

$S_2 = Q\big(\mathbb{I}_m - C\,D\,S_1\big),\ S_2 \in \mathbb{R}^{(m-q)\times m}$,

and $Q \in \mathbb{R}^{(m-q)\times m}$ such that S_2 has full row rank. Take the $n \times n$ matrix

$$T = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} S_1\,C \\ \begin{bmatrix} S_2\,C \\ M \end{bmatrix} \end{bmatrix}, \tag{8}$$

with $M \in \mathbb{R}^{(n-m)\times n}$ in such a way that makes T invertible and $MD = 0$.

Taking the linear transformation $z = T\,x$, the system is transformed into

$$\begin{aligned} \dot{z}_1 &= \bar{A}_{11}z_1 + \bar{A}_{12}z_2 + \bar{B}_1 u(t) + \nu(t), \\ \dot{z}_2 &= \bar{A}_{21}z_1 + \bar{A}_{22}z_2 + \bar{B}_2 u(t), \end{aligned} \tag{9}$$

with $\bar{A} = TAT^{-1}$, $\bar{B} = TB$, and $z_i = T_i x$, $i = \{1, 2\}$. Define as output $\bar{y}(t) = S\,y(t)$, this yields

$$\bar{y}(t) = \begin{bmatrix} \bar{y}_1(t) \\ \bar{y}_2(t) \end{bmatrix} = \begin{bmatrix} z_1 \\ \bar{C}_{22}z_2 \end{bmatrix}.$$

With this transformation, the state is partitioned into the corrupted part by the unknown input z_1, and the part free from its effect z_2. Notice that the corrupt part can be recovered directly from the output, whereas the other part has to be reconstructed from $\bar{y}_2$. This transformation is classic and can be found in many texts, see for example (Kudva et al., 1980) and (Moreno, 2001).

To be able to reconstruct the full state, at least asymptotically, it is required that the pair (A, C) is detectable. If $m = q$ only $\bar{y}_1$ would be available, and from the detectability it can be concluded that $z_2 \to 0$ as $t \to \infty$, then any estimate of z_2 that goes to zero can be used. If $m > q$, i.e. there are more

outputs than unknown inputs, a better estimation of z_2 can be attempted. The detectability of pair (A, C) implies the detectability of pair $(\bar{A}_{22}, \bar{C}_{22})$, and an observer for z_2 can be designed using $\bar{y}_2$ as output, and z_1 and $u(t)$ as input. The interesting case is when $(\bar{A}_{22}, \bar{C}_{22})$ is observable. In such case, the velocity of convergence for a z_2 observer can be assigned. If one uses the observer proposed in Section 2, it is possible to reconstruct z_2 in fixed-time, and also the original state using the relation $x = T^{-1}z$.

Let us summarize the previous discussion. Suppose that the pair (A, C) is observable, that $\text{rank}(CD) = q$, and that $m > q$. Under these conditions the observer proposed in Section 2 can be used to estimate, in fixed-time, the internal state of a linear system with unknown inputs. This is done through the state and output transformation $z = Tx$, $\bar{y}(t) = S\,y(t)$, with T as in (8) and S as in (7). This puts the system into the form shown in (9). Then the methodology presented in the previous section can be applied to design an observer for the subsystem

$$\begin{aligned} \dot{z}_2 &= \bar{A}_{22}z_2 + \bar{A}_{21}z_1 + \bar{B}_2u(t), \\ \bar{y}_2(t) &= \bar{C}_{22}z_2. \end{aligned}$$

This allows the fixed-time reconstruction of the original state through $\hat{x} = T^{-1}\hat{z}$ with $\hat{z} = [\bar{y}_1^\top \ \hat{z}_2^\top]^\top$.

4. NUMERICAL EXAMPLE

To exemplify the observer design proposed in previous sections, let us consider the following system:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 1 & 2 & -1 \\ -7 & -6 & 2 \\ -1 & -2 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t) + \begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix} \nu(t), \\ y(t) &= \begin{bmatrix} 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix} x. \end{aligned} \tag{10}$$

Now, the system has to be transformed in order to decouple the corrupted and uncorrupted dynamics. For this matter, define the transformation matrix as

$$T = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \left[\begin{array}{ccc} 1 & 0 & 1 \\ \hline 2 & 1 & 0 \\ 0 & 1 & -1 \end{array}\right] \tag{11}$$

In this case, matrix S is $\mathbb{I}_2$, which is clear after comparing T and C. Applying the transformation to the system, it becomes

$$\begin{aligned} \dot{z}_1 &= -2\,z_1 + [1 \ -1]z_2 + u(t) + \nu(t), \\ \dot{z}_2 &= \begin{bmatrix} -3 & 1 \\ -4 & 0 \end{bmatrix} z_2 + \begin{bmatrix} 1 \\ 2 \end{bmatrix} z_1 + \begin{bmatrix} 0 \\ -1 \end{bmatrix} u(t), \\ \bar{y}_1 &= z_1, \\ \bar{y}_2 &= [1 \ 0]z_2. \end{aligned} \tag{12}$$

Since the pair $(\bar{A}_{22}, \bar{C}_{22})$ is observable, we can proceed to design an observer for z_2.

To estimate z_2 with the proposed observer, we set $L = 0$, $p_1 = 1/3$, $p_2 = 2$, $k_1 = k_2 = 5$, and $K = [3.7 \ \ 3.9]^\top$. For $R = \mathbb{I}_2$, this configuration yields

$$P^{-1} = \frac{1}{17}\begin{bmatrix} 42 & 36 \\ 36 & 60 \end{bmatrix}.$$

For the simulation, the input u was set in $\cos(t)$; for the unknown input, a triangle wave of amplitude and period 2 was chosen. The system was initialized in $x_0 = [10 \ \ -5 \ \ 3]^\top$. To compare the response of the proposed algorithm, a linear one was also simulated. For the linear observer, the gain $\bar{L} = [4 \ 6]^\top$ was chosen. For both observers, the initial conditions were set in zero, $\hat{z}_2(t_0) = 0$.

From Figure 1 to 3, the estimate of x provided by the proposed observer and the linear observer can be compared with the true state. In the plots, it can be observed that the estimate produced with the nonlinear algorithm got closer to the true state much faster than the linear one. This can be corroborated in Figure 4, in which the norm of the estimation error was plot. From this last graph, the finite-time convergence can be established in less than two seconds.

To exhibit the fixed-time convergence, the simulation was repeated with different initial conditions. The initial value of the estimation error was increased in one order of magnitude in each successive simulation. As can be seen in Fig 5, the convergence time stay under $2.6[s]$ although the initial error was increased above 10^6 in magnitude. Between the trajectories for $O(10^5)$ and $O(10^6)$ there is no appreciable difference, and this characteristic is maintained for orders of magnitude beyond 10^6.

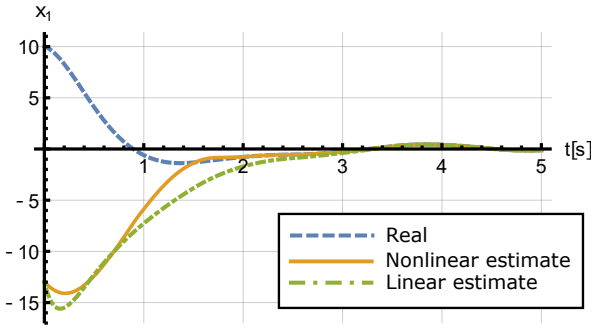


Fig. 1. Estimate of $x_1(t)$.

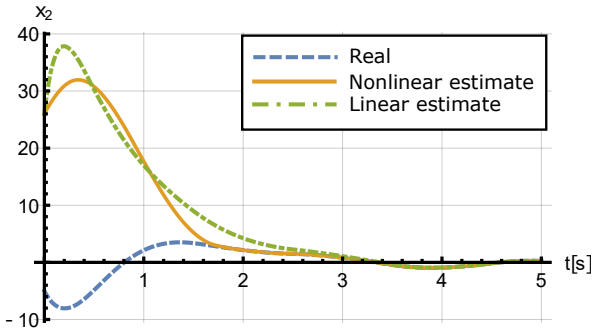


Fig. 2. Estimate of $x_2(t)$.

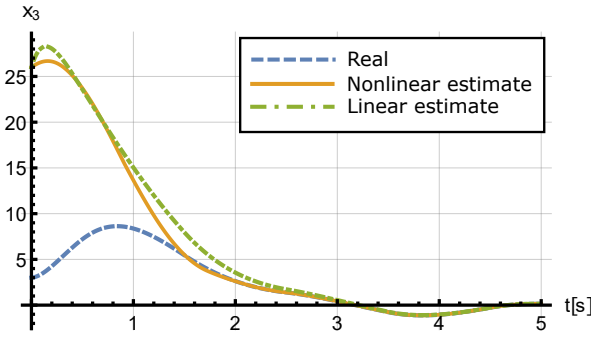


Fig. 3. Estimate of $x_3(t)$.

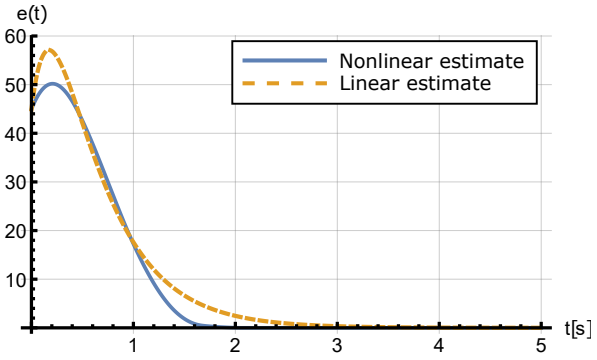


Fig. 4. Error norm.

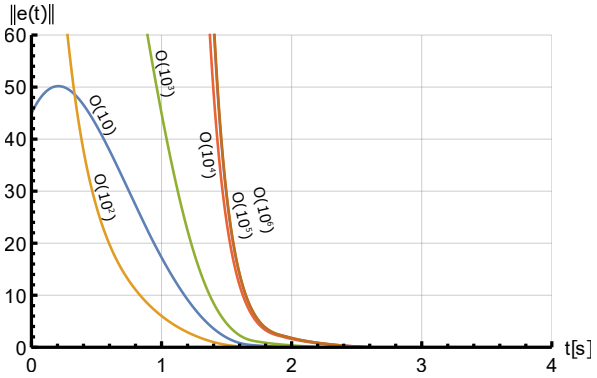


Fig. 5. Error norm for different magnitude of the initial error.

5. CONCLUSION

In this work, a novel fixed-time convergent observer for any observable linear time-invariant systems is presented. The observer is globally convergent, and its design can be done in the system original coordinates, in contrast with other finite-time convergent algorithms. It is also provided a methodology to apply this observer in the case of perturbed systems, and can be used as an unknown input observer. In such situation, if there is enough information available, the algorithm keeps its properties, that is, the algorithm can provide an estimate of the perturbed system state in fixed time.

ACKNOWLEDGEMENT

The authors thank the financial support from PAPIIT-UNAM (Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica), project IN113614; Fondo de Colaboración II-FI UNAM, Project IISGBAS-100-2015; CONACyT (Consejo Nacional de Ciencia y Tecnología), project 241171; and CONACyT CVU: 491701, and 620585.

REFERENCES

Abou-Kandil, H., Freiling, G., Ionescu, V., and Jank, G. (2003). *Matrix Riccati Equations in Control and Systems Theory*. Birkhäuser Basel.

Atassi, A. and Khalil, H. (2000). Separation results for the stabilization of nonlinear systems using different high-gain observer designs. *Systems & Control Letters*, 39(3), 183 – 191.

Chen, W. and Saif, M. (2006). Fault detection and isolation based on novel unknown input observer design. In *2006 American Control Conference*.

Cruz-Zavala, E., Moreno, J.A., and Fridman, L. (2012). Asymptotic stabilization in fixed time via sliding mode control. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 6460–6465.

Cruz-Zavala, E., Moreno, J.A., and Fridman, L.M. (2011). Uniform robust exact differentiator. *IEEE Transactions on Automatic Control*, 56(11), 2727–2733.

Hautus, M. (1983). Strong detectability and observers. *Linear Algebra and its Applications*, 50, 353 – 368.

Hespanha, J.P. (2009). *Linear Systems Theory*. Princeton Press, Princeton, New Jersey.

Hwang, I., Kim, S., Kim, Y., and Seah, C.E. (2010). A survey of fault detection, isolation, and reconfiguration methods. *IEEE Transactions on Control Systems Technology*, 18(3), 636–653.

Kudva, P., Viswanadham, N., and Ramakrishna, A. (1980). Observers for linear systems with unknown inputs. *IEEE Transactions on Automatic Control*, 25(1), 113–115.

Menard, T., Moulay, E., and Perruquetti, W. (2010). A global high-gain finite-time observer. *IEEE Transactions on Automatic Control*, 55(6), 1500–1506.

Moreno, J. (2001). Existence of unknown input observers and feedback passivity for linear systems. In *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, volume 4, 3366–3371 vol.4.

Pin, G., Lovera, M., Assalone, A., and Parisini, T. (2013). Kernel-based non-asymptotic state estimation for linear continuous-time systems. In *American Control Conference (ACC), 2013*, 3123–3128.

Polyakov, A. (2012). Fixed-time stabilization of linear systems via sliding mode control. In *Variable Structure Systems (VSS), 2012 12th International Workshop on*, 1–6.

Raff, T. and Allgöwer, F. (2008). An observer that converges in finite time due to measurement-

based state updates. In *17th IFAC World Congress*, 1051–1067.

Rueda-Escobedo, J., Moreno, J.A., and Oliva-Fonseca, P. (2016). Finite-time state estimation for lti systems with a first-order sliding mode. In *14th International Workshop on Variable Structure Systems*, to appear.

Yin, S., Wang, G., and Karimi, H.R. (2014). Data-driven design of robust fault detection system for wind turbines. *Mechatronics*, 24(4), 298 – 306.

Generalized Predictive Traffic Control for Isolated Intersections

Anna Sarrazola, **Laura Noreña**, **Alejandro Marquez**, **Jairo Espinosa**

Departamento de Energía Eléctrica y Automática, Universidad Nacional de Colombia, Colombia, (e-mail: amsarrazolaa@unal.edu.co, lmnorenam@unal.edu.co, amarque@unal.edu.co, jespinov@unal.edu.co)

Abstract: This paper contains the results of simulation in the implementation of a Generalized Predcitive Control (GPC) Model for a traffic light system of an isolated intersection of network traffic, using the SUMO microsimulator as simulated plant and the linear prediction model called Modified Multi-class Queueing Networks (MMQM) for the GPC.

Keywords: GPC controller,MMQM, traffic light control, microsimulation, SUMO.

1. INTRODUCTION

The steady growth of big cities as focus of economic activity has generated an increased demand of transportation and traffic. Therefore, the mobility is negatively affected by phenomena such as congestion delaying the total journey time of the users. Furthermore the pollution and toxic emissions increase due to these phenomena affecting the quality of living of the citizens (Papageorgiou et al., 2003).

Due to the inherent dynamic nature of the traffic network, its evolution in time can be represented by mathematical dynamic models, which are used in simulation environments for analyzing the behavior of variables of interest that describe the network. These models are key elements for the design of traffic control strategies for achieving a sustainable mobility for the users. Dynamic modeling of vehicular traffic is an important factor in the monitoring and control of traffic (Yang et al., 2010) (Lighthill and Whitham, 1955) (Daganzo, 1995).

In order to simplify the implementation of controllers in large traffic networks, some linear traffic models have been developed. Multi-Class Queueing Networks (MQN)(Le et al., 2013) are a simple and important linear representation due to different class relations allowing to describe any traffic. Howevwer, this model assumes a certain quantity of vehicles leaving each class at every time step (regardless of whether there are available vehicles or not). Therefore, a modification of this model was proposed in a previous work (Norena et al., 2015). We have called Modified Multiclass Queueing Networks (MMQN). The MMQN model assumes that what goes out of a class is a proportion of the current number of vehicles in that case. It is worth to notice that the MMQN model is still linear, which is suitable for Generalized Predictive Controller (GPC) implementation.

The whole proposal, apart from the MMQN, consist on estimate the parameters of the proposed model assuming a SUMO as the real system. SUMO is a high performance microscopic traffic simulator featuring tools to generate traffic demand and to import road networks from several sources (Krajzewicz et al., 2012). In addition, a Kalman filter-based (Kalman, 1960) GPC scheme using the MMQN as the prediction model is implemented through simulation.

This paper is organized as follows: Section 2 presents the mathematical formulation of the Modified Multi-class Queueing Network Based Model; Section 3 describe the case study, including its modeling via MMQN, design of experiment, parameter identification, state estimation structure and GPC scheme; Section 4 presents a numerical simulation of the proposed case study.

2. MODIFIED MULTI-CLASS QUEUEING NETWORK BASED MODEL

The Multi-class Queueing Networks Based Model (MQN) is a representation of urban traffic networks proposed in (Le et al., 2013). The modified Multi-class Queueing Network Based Model (MMQN) modeling methodology is motivated by the MQN methodology in an attempt to address some issues.

2.1 Definition of the network elements

It is assumed that time evolves in discrete steps (n = 0,1,2...) corresponding to traffic light cycles (Le et al., 2013). The network state maintains counts of the quantity of vehicles at different abstractions of locations based on the following 3 types of classes:

- Delay (D): this class represents a portion of a (multilane) street where vehicles are in free flow regime. This class can only evolve to another delay (D) class or a route (R) class (Le et al., 2013).
- Roure (R): this class represents a portion of a (multiline) street where drivers decide where to go among all options of turning. This class typically evolves into queue (Q) classes (Le et al., 2013).

⋆ Research supported by project: Modelamiento y control de tráfico urbano en la ciudad de Medellín etapa 2, código 1118-669-45309

- Queue (Q): this class represents a turning direction before an intersection (Le et al., 2013).

Let K_D, K_R and K_Q be the number of classes of each type. Classes are indexed by $k = 1, ..., k$ with $K = K_D + K_R + K_Q$. Classes of type $j \in D, R, Q$ are denoted by H_j and they are numbered as follows: $H_D = 1, ..., K_D$, $H_R = K_D + 1, ..., K_D + K_R$ and $H_Q = K_D + K_R + 1, ..., K_D + K_R + K_Q$ (i.e the delay (D) classes are the first to be numbered, followed by the route (R) classes an finally the queue (Q) classes) (Le et al., 2013).

The network state for a class k, denoted by $x_k(n)$, is an instantaneous count of vehicles in the class $k \in H_{D,R,Q}$. The state vector $(X(n) \in \mathbb{R}^{\urcorner}$ is the vector formed by all state related to each class (i.e $X(n) = [x_1(n)...x_k(n)]^T$) (Le et al., 2013).

Vehicles leaving any class, except classes of type Q, move to other classes through predefined links. A link l is a connection between its source class, denoted by $s_l \in H_{D,R}$, and its destination class, denoted by $d_l \in H_{D,R,Q}$. Links are indexed by $l = 1, ..., L$. Each link l connecting two classes, say $s_l = x_k$; $k \in H_{D,R}$ and $d_l = x_k^{'}$; $k^{'} \in H_{D,R,Q}$, has an associated number $\alpha_{k,k'} \in [0, 1]$, which is a dimensionless number that defines the proportion of the current number of vehicles that will go from the states k to the state $k^{'}$ over a whole traffic light cycle. Furthermore, if it is not a link, for example from class i to the class j, define $\alpha_{i,j} = 0$. Additionally, Q-type classes (recall Q-type classes are just before intersections) have an associated maximum flow rate of vehicles leaving these kind of classes in an unitary time step (a traffic light cycle). These flow rates are labeled as $f_{i,max}$; $i \in F$, where i indexes the phases and $F = \{1, ..., f\}$ is the set of all phases of the intersection. A phase is an arrangement of the different turning options in an intersection given a configuration of the traffic light. Consequently, every phase has a duration $g_{ui}(n)$, which corresponds to the amount of time the phase i is active in a whole traffic light cycle.

The last element considered by this model is the arrival of vehicles to an intersection. Vehicles arrive exogenously according to the demand that arrives to a class k, denoted by $a_k(n)$; $k \in H_{D,R,Q}$. Each class k has associated a demand $a_k(n)$, which is positive if exogenous vehicles can arrive to class k and zero if they cannot.

2.2 Network Equations

According to the previous subsection, the proposed MMQN equations are:

Si $k \in H_{D,R}$:

$$x_k(n+1) = x_k(n) + a_k(n) + \sum_{j \in H_{D,R}} \alpha_{j,k} x_j(n) - \sum_{j \in H_{D,R}} \alpha_{k,j} x_k(n)$$

Si $k \in H_Q$:

$$x_k(n+1) = x_k(n) + a_k(n) + \sum_{j \in H_{D,R}} \alpha_{j,k} x_j(n) - \sum_{i \in F} g_{ui}(n) f_{i,max}$$

3. CASE STUDY

An isolated urban intersection, is taken as example for applying the GPC. The prediction model used in the GPC strategy will be the MMQN.

3.1 Case study definition

It is a scenario of an isolated intersection composed by four links $O1a, O2a, O3a$ and $Oa4$ that intersect at the point as shown in Figure 1. In each link, the vehicles have two intentions of rotation: right turn and circulation in a straight line trajectory.

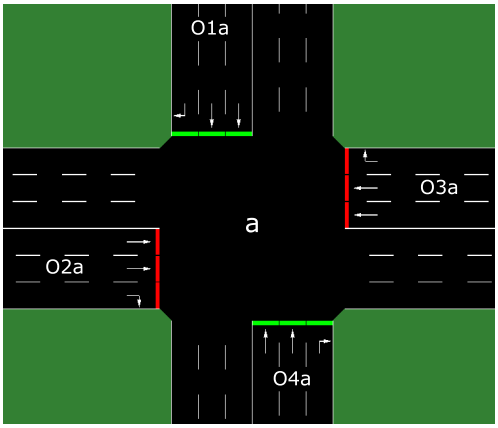


Fig. 1. Isolated intersection view from SUMO

In this intersection, vehicles can arrive from any street and they can also turn in the direction according to the phases configuration show on Figure 2.

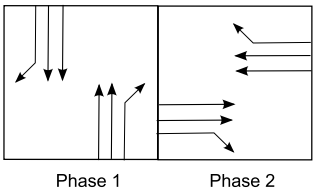


Fig. 2. Phases for an isolated intersection

As shown in equation 1, the duration of phase 2, $g_{u2}(n)$ is the complement of the duration of phase 1, $g_{u1}(n)$ to complete the total cycle time cd.

$$g_{u2}(n) = cd - g_{u1}(n) \tag{1}$$

The equations representing the model of the intersection using MMQM are presented below:

$$x_1(n+1) = x_1(n) + a_1(n) - \alpha_{1,5} x_1(n) \tag{2}$$

$$x_5(n+1) = x_5(n) + \alpha_{1,5} x_1(n) - \alpha_{5,9} x_5(n) - ... - \alpha_{5,10} x_5(n) - \alpha_{5,11} x_5(n) \tag{3}$$

$$x_9(n+1) = x_9(n) + \alpha_{5,9} x_5(n) - g_{u_1}(n) f_1$$

$$x_{10}(n+1) = x_{10}(n) + \alpha_{5,10} x_5(n) - g_{u_1}(n) f_1$$

$$x_{11}(n+1) = x_{11}(n) + \alpha_{5,11} x_5(n) - g_{u_1}(n) f_1$$

$$x_2(n+1) = x_2(n) + a_2(n) - \alpha_{2,6}x_2(n) \quad (4)$$

$$x_6(n+1) = x_6(n) + \alpha_{2,6}x_2(n) - \alpha_{6,12}x_6(n) - ... - \alpha_{6,13}x_6(n) - \alpha_{6,14}x_6(n) \quad (5)$$

$$x_{12}(n+1) = x_{12}(n) + \alpha_{6,12}x_6(n) - g_{u_2}(n)f_2$$
$$x_{13}(n+1) = x_{13}(n) + \alpha_{6,13}x_6(n) - g_{u_2}(n)f_2$$
$$x_{14}(n+1) = x_{14}(n) + \alpha_{6,14}x_6(n) - g_{u_2}(n)f_2$$

$$x_3(n+1) = x_3(n) + a_3(n) - \alpha_{3,7}x_3(n) \quad (6)$$

$$x_7(n+1) = x_7(n) + \alpha_{3,7}x_3(n) - \alpha_{7,15}x_7(n) - ... - \alpha_{7,16}x_7(n) - \alpha_{7,17}x_7(n) \quad (7)$$

$$x_{15}(n+1) = x_{15}(n) + \alpha_{7,15}x_7(n) - g_{u_2}(n)f_3$$
$$x_{16}(n+1) = x_{16}(n) + \alpha_{7,16}x_7(n) - g_{u_2}(n)f_3$$
$$x_{17}(n+1) = x_{17}(n) + \alpha_{7,17}x_7(n) - g_{u_2}(n)f_3$$

$$x_4(n+1) = x_4(n) + a_4(n) - \alpha_{4,8}x_4(n) \quad (8)$$

$$x_8(n+1) = x_8(n) + \alpha_{4,8}x_4(n) - \alpha_{8,18}x_8(n) - ... - \alpha_{8,19}x_8(n) - \alpha_{8,20}x_8(n) \quad (9)$$

$$x_{18}(n+1) = x_{18}(n) + \alpha_{8,18}x_8(n) - g_{u_1}(n)f_4$$
$$x_{19}(n+1) = x_{19}(n) + \alpha_{8,19}x_8(n) - g_{u_1}(n)f_4$$
$$x_{20}(n+1) = x_{20}(n) + \alpha_{8,20}x_8(n) - g_{u_1}(n)f_4$$

3.2 Experimental set up

The controlled signal for the MMQM model and SUMO was $g_1(n)$ and the total light cycle was set to a constant value $cd = 60[s]$. The input vehicle demand was considered as a disturbance (in this case known) for estimating the parameters. The inputs were chosen in order to avoid saturating the intersection. Additionally, the cumulative sum of the vehicles entering each link, for one hour (3600 seconds) is as follows.

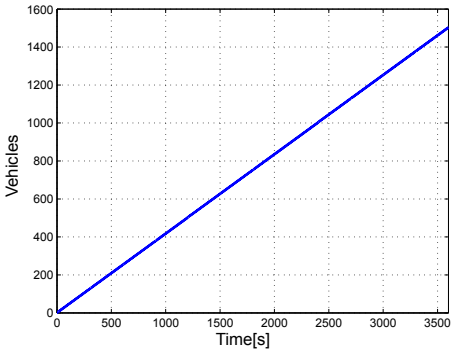


Fig. 3. Input demand

Table 1 shows a constant flow of vehicles entering each lane of each link.

Table 1. Input demand vehicles

Link	Input flow [veh/h]
o1a	500
o2a	300
o3a	400
o4a	400

3.3 Parameter estimation

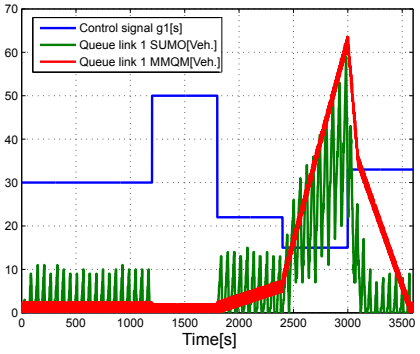
The parameter identification was done using the Parameter Estimation Toolbox from Simulink, which performs the estimation through optimization procedures based on input/output data. The identified parameters were $f_{i,max}$ in this case $f1, f2, f3, f4$ related with the maximum output flow of vehicles that can cross from one link to another through the intersection. They depend directly on the input demand of vehicles $a_i(n)$ of each link and must take nonnegative values. The parameters $\alpha_{1,5}, \alpha_{2,6}, \alpha_{3,7}, \alpha_{4,8}, \alpha_{5,9}, \alpha_{5,10}, \alpha_{5,11}, \alpha_{6,12}, \alpha_{6,13}, \alpha_{6,14}$, $\alpha_{7,15}, \alpha_{7,16}$ $\alpha_{7,17}, \alpha_{8,18}, \alpha_{8,19}, \alpha_{8,20}$ are not estimated, they are assumed to be a constant and balanced for a distribution of vehicles on each turning direction.

An input signal was designed in such a way that the green light time changes in an interval between 10 and 50 seconds to ensure that a cycle time (60 seconds) contains both phases. The identified parameters are shown in Table 2

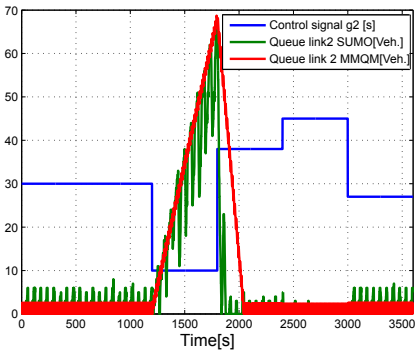
Table 2. Parameter estimation

Parameter	Value
f_1	0.0072
f_2	0.0046
f_3	0.0053
f_4	0.0067

Figure 4 shows the comparison between the MMCM model and the SUMO data when the input signal is applied. Both respond to the control signal, decreasing the vehicles accumulation when the green light time increases and vice-versa. It can also be noticed that the model trend fits the SUMO data.



(a) Phase 1 comparison



(b) Phase 2 comparison

Fig. 4. Comparison: estimated parameters - SUMO - control action

3.4 State estimator

A Kalman filter based estimator was designed using the extended model described in equations (Kalman, 1960). This model includes an integral action for representing the uncertainties in the system.

$$\begin{aligned} x(n+1) &= Ax(n) + Bu(n) + Gw(n) + B_d d(n) \\ y(n) &= Cx(n) + v(n) \\ d(n+1) &= d(n) + B_{w_d} w_d(n) \end{aligned} \tag{10}$$

The terms G_w and V_n corresponds to the process and measurement noise, respectively, d_n is included for tracking purposes. For the filter design the system is written as follows:

$$\begin{bmatrix} x(n+1) \\ d(n+1) \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x(n) \\ d(n) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(n) + \begin{bmatrix} G & 0 \\ 0 & B_{w_d} \end{bmatrix} \begin{bmatrix} w(n) \\ w_d(n) \end{bmatrix} \tag{11}$$

$$y(n) = [C \;\; 0] \begin{bmatrix} x(n) \\ d(n) \end{bmatrix} + v(n)$$

Covariance matrices tuned to design the filter were $Qn = 10^5 I$ y $Rn = 10I$, $Bd = B$. Finally, the state observer is:

$$\begin{bmatrix} \hat{x}(n+1) \\ \hat{d}(n+1) \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}(n) \\ \hat{d}(n) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(n) + L\left(y - [C \;\; 0] \begin{bmatrix} \hat{x}(n) \\ \hat{d}(n) \end{bmatrix} \right) \tag{12}$$

3.5 Generalized Predictive Control (GPC)

This section provides a Generalized Predictive Control with prediction horizon $N_p = 10$ for tracking references. The objective function is described as follows (explained in *section 3.3*):

$$\min_U \sum_{j=1}^{N} ||y(n+j) - y_{ref}||_Q^2 + \sum_{j=0}^{N} ||u(n+j)||_R^2 \tag{13}$$

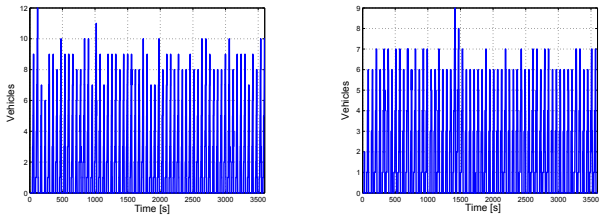
Subject to:

$$\begin{aligned} & x(n+j+1) = Ax(n+j) + Bu(n+j) \\ & y(n+j) = Cx(n+j) \\ & 10 \leq u_i(n+j) \leq 50 \;\; \forall j = 1, ..., N \\ & 0 \leq y_i(n+j) \leq 44 \;\; \forall j = 1, ..., N \\ & u_1(n+j) + u_2(n+j) = 60 \;\; \forall j = 1, ..., N \end{aligned}$$

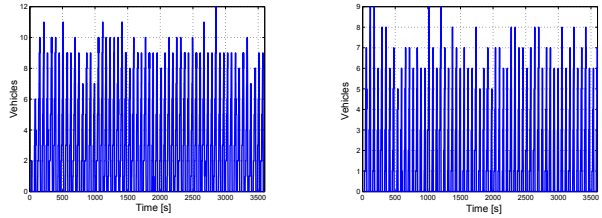
The aim is to have the minimum queue length, thus the reference signal is set to zero (no queue). The constraints denote minimum and maximum values for the phase duration u_i. The sum of phase durations is the total cycle time (60s), the maximum and minimum values of queue length y_i based on link the capacity and the variables are subject to the MMQN model.

4. SIMULATION RESULTS

The obtained queue length of each link with $y_{ref} = 0$ and constant input demands are shown in following figures:



(a) vihicles queue link 1 (b) vihicles queue link 2



(c) vihicles queue link 3 (d) vihicles queue link 4

Fig. 5. Queue length with constant demand

It is noted how vehicles leaves the links every traffic light cycle, and they are not accumulated to form congestion. Also, the controller gives priority to higher demand phase, in this case Phase 1, and varies the control signal (green timing)to maintain the network balance, as shows 6.

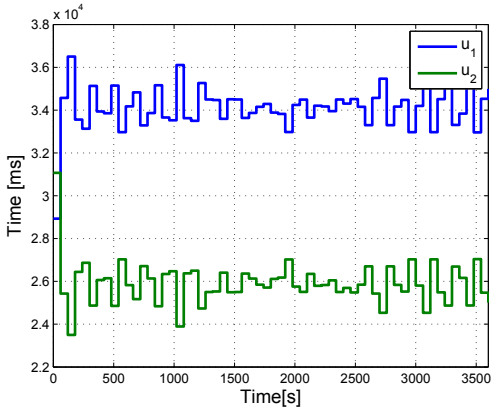


Fig. 6. GPC control signal with constant demand

In order to test the response of the GPC controller in another situation, a change on the input demand was tested. The constant input changes to variable demand in the middle of the simulation time, $t = 1800s$. Thus, figure 7 shows how the controller is looking for keeping the network balance by reducing the green time of Phase 1.

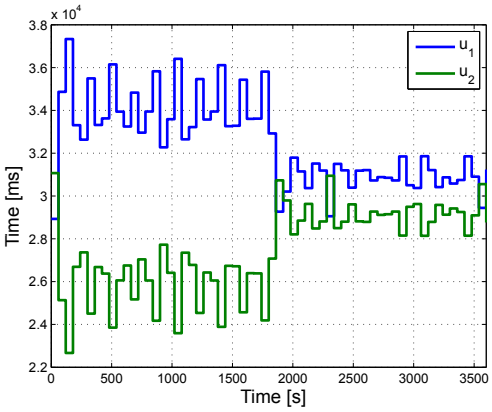


Fig. 7. GPS control signal with variable demand

4.1 Reference change

In this subsection a change of variable reference for the link 2 of phase 2 was applied, the rest of links have zero reference throughout simulation time.

According to Figure 8, it is observed how the length of the queues follows the reference, changing the value from 0 to 18 vehicles in the link 2. However, the queues do not reach the reference value due to the nature of the system, since the simulated demand is not sufficient to accumulate the desired value.

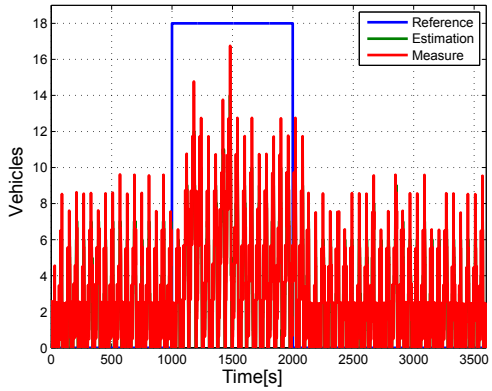


Fig. 8. Vehicles in the queue link 2 reference exchange

In Figure 9, it is observed that the time interval in Green of Phase 2 decreases in order to achieve the reference.

5. CONCLUSION

Optimal controllers are subject to covariance matrices that define the weights to give importance to some control objectives over others. Therefore the tuning of these matrices is a critical part in the design of these controllers. If the value of the R matrix is very big the controller does not respond to significant changes in demand, because the control variable is strongly penalized.

The total number of vehicles waiting at the interaction during simulation, compared between strategies fixed time and GPC, as shown in the Table 3

The GPC controller has cumulated 695 vehicles less at the intersection compared with to the fix time control strategy.

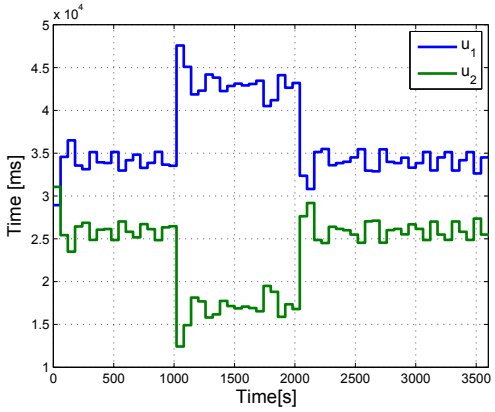


Fig. 9. GPC control signal reference exchange

Table 3. Comparison of strategies

Controller	Value
Fixed Time	29103
GPC	28408

The level of service at the intersection improved by about 3%, also the GPC controller responded satisfactorily to changes in the demand.

ACKNOWLEDGEMENTS

Research supported by COLCIENCIAS project: Modelamiento y control de tráfico urbano en la ciudad de Medellín etapa 2, código 1118-669-45309, CT 202-2015. Universidad Nacional de Colombia Proyecto HERMES 25374

REFERENCES

Daganzo, C.F. (1995). Requiem for second-order fluid approximations of traffic flow. *Transportation Research Part B*, 29(4), 277–286.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1), 35–45.

Krajzewicz, D., Erdmann, J., Behrisch, M., and Bieker, L. (2012). Recent Development and Applications of {SUMO - Simulation of Urban MObility}. *International Journal On Advances in Systems and Measurements*, 5(3), 128–138.

Le, T., Vu, H.L., Nazarathy, Y., Vo, Q.B., and Hoogendoorn, S. (2013). Linear-quadratic model predictive control for urban traffic networks. *Transportation Research Part C: Emerging Technologies*, 36, 498–512.

Lighthill, M.J. and Whitham, G.B. (1955). On Kinematic Waves. II. A Theory of Traffic Flow on Long Crowded Roads. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 229(1178), 317–345.

Norena, L., Sarrazola, A., Jim, E., Jaramillo, O., Portilla, C., Alejandro, M., and Espinosa, J. (2015). Model Predictive Control of an Urban Traffic Intersection Based on the Modified Multi-class Queuing Networks Models. 1–4.

Papageorgiou, M., Kiakaki, C., Dinopoulou, V., and Kotsialos, a. (2003). Review of road traffic control strategies. *Proceedings of the IEEE*, 91(12), 2043–2067.

Yang, J., Zhang, L., and Chen, Y. (2010). Modeling and control of signaling split in urban traffic network based on hybrid systems. *2010 Sixth International Conference on Natural Computation*, (Icnc), 3497–3502.

Generalized Proportional Integral Control for Aperiodic Gait Stabilization of a Bipedal Robot with Seven Degrees of Freedom ⋆

Arcos-Legarda Jaime * **Cortes-Romero John** ** **Tovar A.** ***

* *Department of Mechanic and Mechatronic Engineering, Universidad Nacional de Colombia, Bogotá, Colombia, (wjarcosl@unal.edu.co).*
** *Department of Electric and Electronics Engineering, Universidad Nacional de Colombia, Bogotá, Colombia, (jacortesr@unal.edu.co)*
*** *Mechanical Engineering, Indiana University-Purdue University Indianapolis, Indianapolis, Indiana, USA, (tovara@iupui.edu)*

Abstract: The main goal of this paper is to achieve asymptotically stable walking for a bipedal robot over an Aperiodic Gait pattern. The five-link planar bipedal robot has one degree of under-actuation, four actuators and point feet. Trajectory tracking and multi-orbital stability were performed using Generalized Proportional Integral (GPI) controller. The stability of the multi-periodical walking was tested through the computation of Poincaré return maps. This analysis was performed by numerical simulation with a gait pattern reconfiguration. The results of the simulation show that the GPI control achieved a performance robust enough to overcome the gait pattern reconfiguration without changes in the control law with an event-based action. The robot analyzed corresponds to a prototype under development at the Control Laboratory of the *Universidad Nacional de Colombia, Bogotá*.

Keywords: Bipedal Robots, Aperiodic Gait, GPI Control, Hybrid Systems.

1. INTRODUCTION

Although the study of bipedal walking robots has been a highly active research area, bringing outstanding achievements in the last three decades, important problems are still unsolved regarding the design and control of bipedal robots. One of the open issues in which research needs to focus is in ensuring the stability of aperiodic gaits, Grizzle et al. (2014). Even if we can assume that the nature of the walking is a periodic repetition of movements, there are some factors like uneven ground or external disturbances, that may produce the needs to interrupt the periodicity of the gait. This paper proposes a control system and a trajectory generation strategy that ensure the stability of the walking in the changes of periodic gait patterns.

The control strategies to achieve a stable gait in the presence of uneven ground or external disturbances have been based on the use of event-based feedback controllers, as it was proposed by Grizzle et al. (2003) and Hamed and Grizzle (2013). In the present work a GPI robust controller was developed, which can be used with different walking patterns. A GPI robust control had been previously used in the simulation of the control of a gait exoskeleton in Arcos-Legarda et al. (2013), but without the analysis of underactuted walking phase and with a fix periodic trajectory. Whereas, the research presented here takes into a count the underactuated phase and the dynamic walking though the use of a bipedal robot with point feet.

The approach used by the GPI control is based on the assumption that the unmodeled dynamics and the external disturbances can be lumped in a total disturbance which can be approximately reconstructed by a *m*-order polynomial. Similar methods have been used by the authors in Arcos-Legarda et al. (2016), where a periodic gait stability in bipedal robots was achieved. Other works for active disturbance rejection for bipedal robots were developed with time-based trajectories, as Martínez-Fonseca et al. (2016) or based in Zero Moment Point (ZMP) control, like in Hill and Fahimi (2015).

This paper is organized as follows: In the section 2 are described the features of the robot and its mathematical model is developed. Section 3 presents the GPI control strategy proposed. Then a trajectory generation strategy based on the state of the robot is depicted in section 4. The simulation of the strategies proposed in the previous sections is evaluated in the section 5, and additionally a stability test is done. Finally in section 6 conclusions are drawn.

2. ROBOT MODEL

The robot's model was obtained from the study of the bipedal robot prototype shown in Fig. 1. This robot is under development at the Control Laboratory of the *Universidad Nacional de Colombia, Bogotá*. The model is described as a multibody system formed by the five-link mechanism shown in the Fig.2. The bipedal robot has a torso connected to one leg at each sides of the hip. The legs are composed of a thigh serially with a shin connected by a revolute joint, working as a knee. The leg ending has a point-foot, without ankle. The robot's movements are restricted to the saggital plane by a radial bar with central pivot located 1m away from the robot. The description of the robot shows seven degrees of freedom: the angle of the torso respect to a fix frame, two angles at the hip, two in the knees and the cartesian plane position of the hip.

⋆ This paper is a result of the research project number 28340, supported by the Engineering Research Division of the National University of Colombia.

The bipedal walking is a sequence of changes between single and double support. Here, it is assumed that the double support phase is instantaneous and that the end of the support leg has unilateral constraints. In this way it does not have rebound nor slip and its vertical reaction force is repulsive and the lateral force is inside of the friction cone. This constraint converts the end of the support leg into a pivot, which transforms the system into a five degrees of freedom mechanism. The robot has four actuators: two between the torso and the thighs and two more in the knees. Although the actuators have flexible joints to the links, here the stiffness of the union is configured to its maximum value, it in order to avoid the additional dynamic of the strings. As in Tzafestas et al. (1996), Westervelt et al. (2007) and Arcos-Legarda et al. (2016) the Lagrange's differential equation was used to find the model of the robot in the single support phase to get the Euler-Lagrange expressed as:

$$D_s(q_s)\ddot{q}_s + C(q_s,\dot{q}_s)\dot{q}_s + G_s(q_s) = B_s(q_s)u + \delta(q_s,\dot{q}_s) + \zeta, \quad (1)$$

where $q_s := [q_1 \quad q_2 \quad \dots \quad q_5]^T$ is the generalized coordinates vector, $D_s(q_s)$ is the inertial matrix, $C_s(q_s,\dot{q}_s)\dot{q}_s$ is a vector of Centripetal and Coriolis forces, $G_s(q_s)$ is a vector of forces associated to the gravity, $B_s(q_s)u$ is a vector of generalized forces, $\delta(q_s,\dot{q}_s)$ takes into account the uncertainties of the model and ζ is an unknown vector of the external disturbances.

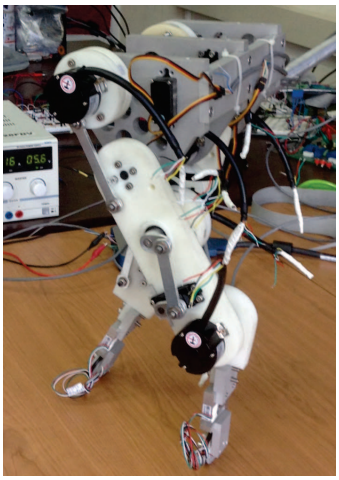


Fig. 1. Robot Prototype, under development at the Control Laboratory of the *Universidad Nacional de Colombia, Bogotá*

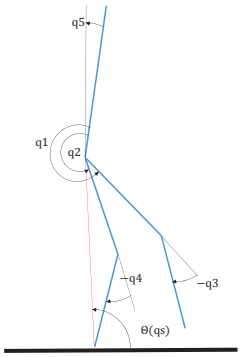


Fig. 2. Multibody Model of the Bipedal Robot

As was exposed by Hurmuzlu and Marghitu (1994), the event of exchange of the support leg produces an impact that is often studied as the collision of rigid bodies. This impact phase together with the continuous dynamic of the single support phase form a hybrid model. As proposed by Grizzle et al. (2001) and Westervelt et al. (2003), it is mandatory to take in consideration external forces affecting the system at the impact time. For this reason the exchange of support leg phase of the walking was studied with the robot as an unpinned system with an augmented model that takes into account a fix point over the robot's body, corresponding to $p_e := [p_e^h \quad p_e^v]^T$, where p_e is the position of the end of the support leg, p_e^h refers to the horizontal coordinate of that point and p_e^v to the vertical one. Hence, the generalized coordinates vector for impact phase is $q_e := [q_s^T \quad p_e^T]^T$. With the new generalized coordinate vector the dynamic model is described as:

$$D_e(q_e)\ddot{q}_e + C_e(q_e,\dot{q}_e)\dot{q}_e + G_e(q_e) = B_e(q_e)u + \delta F_{ext}, \quad (2)$$

where δF_{ext} represents impulsive forces caused by the ground reaction in the swing leg at the impact time. Based on the concept of momentum conservation, the angular momentum before and after the impact must be the same and even when the angular position remains constant in that instantaneous period of time, the velocity undergoes a sudden change. It is shown in the following equation:

$$D_e(q_e^+)\dot{q}_e^+ - D_e(q_e^-)\dot{q}_e^- = F_{ext}, \quad (3)$$

where $(q_e^-,\dot{q}_e^-)$ and $(q_e^+,\dot{q}_e^+)$ represent the position and the velocity just before and after the impact, respectively; F_{ext} is a forces vector which represents the effect of the ground reaction on each joint. Let us define $F_2 := [F_2^v \quad F_2^h]^T$ as the reaction force in the swing leg, where F_2^vrefers to the tangential reaction and F_2^h to the normal one. Following the the procedure of Westervelt et al. (2007), the principle of virtual work could be used, which yields:

$$F_{ext} = E_2(q_e^-)^T F_2, \quad (4)$$

where $E_2 := \frac{\partial p_2}{\partial q_e}$ and p_2 is the position of the swing leg end. Replacing the Eq. (4) into (3) it produces:

$$D_e(q_e^-)\dot{q}_e^- = D_e(q_e^+)\dot{q}_e^+ + E_2(q_e^-)^T F_2. \quad (5)$$

According to the assumption of the impact of rigid bodies the following are true:

- $q_e^- \equiv q_e^+$
- The velocity of the end swing leg would be equal to zero

Then the following equation can be extracted:

$$\frac{dp_2}{dt} = \frac{\partial p_2(q_e^-)}{\partial q_e}\dot{q}_e^+,$$
$$E_2(q_e^-)\dot{q}_e^+ = 0. \quad (6)$$

The Eq. (5) and (6) are collected in a matrix form to obtain

$$\begin{bmatrix}\dot{q}_e^+ \\ F_2\end{bmatrix} = \begin{bmatrix}D_e(q_e^+) & -E_2(q_e^-)^T \\ E_2(q_e^-) & 0_{2\times 2}\end{bmatrix}^{-1}\begin{bmatrix}D_e(q_e^-)\dot{q}_e^- \\ 0_{2\times 1}\end{bmatrix}, \quad (7)$$

which can be summarized in

$$\dot{q}_e^+ = \Delta(q_e^-, \dot{q}_e^-), \tag{8}$$
$$F_2 = \Sigma(q_e^-, \dot{q}_e^-). \tag{9}$$

Finally, the hybrid system is described in the Eq. (1) and (8) as:

$$\Sigma : \begin{cases} \ddot{q}_s = f(q_s, \dot{q}_s) + D_s(q_s)^{-1} B_s(q_s) u & p_2^v \neq 0 \\ \dot{q}_e^+ = \Delta(q_e^-, \dot{q}_e^-) & p_2^v = 0 \end{cases} \tag{10}$$

where

$$f(q_s, \dot{q}_s) = D_s(q_s)^{-1} \left[-C(q_s, \dot{q}_s)\dot{q}_s - G_s(q_s) + \delta(q_s, \dot{q}_s) + \zeta \right].$$

3. CONTROL DESIGN

The control of bipedal robots has several challenges. For instance, it has to deal with the fact that this system is nonlinear and Multi-Input Multi-Output (MIMO). One of the biggest challenges in the control of bipedal robots is that they are hybrid systems, because of their continuous and discrete dynamic. Finally, the point-feet structure of our prototype makes it an underactuated mechanism with its nominal state as an unstable system. Bellow a feedback control design is provided to tackle all the difficulties that bipedal robots with dynamic walking, like the one studied in this paper faces. First, the Eq. (1) is simplified as follow:

$$D_s(q_s)\ddot{q}_s + \Omega(q_s, \dot{q}_s) = B_s(q_s)u + \tilde{\xi}(q_s, \dot{q}_s), \tag{11}$$

where:

$$\Omega(q_s, \dot{q}_s) := C(q_s, \dot{q}_s)\dot{q}_s + G_s(q_s),$$
$$\tilde{\xi}(q_s, \dot{q}_s) := \delta(q_s, \dot{q}_s) + \zeta.$$

The Eq. (11) is fragmented to separate the actuated part of the system from the underactuated part. Now, let us fragment the position vector and the output matrix as

$$q_s := \begin{bmatrix} q_{b_{(N-1\times 1)}} \\ q_{(N_{1\times 1})} \end{bmatrix}, \qquad B_s(q_s) := \begin{bmatrix} B_1(q_s)_{(N-1\times p)} \\ 0_{(1\times p)} \end{bmatrix},$$

where p is the number of actuators. Once the new structure of the variables was established, the model was fragmented as:

$$\begin{bmatrix} D_{11}(q_s) & D_{12}(q_s) \\ D_{21}(q_s) & D_{22}(q_s) \end{bmatrix} \begin{bmatrix} \ddot{q}_b \\ \ddot{q}_N \end{bmatrix} + \begin{bmatrix} \Omega_1(q_s, \dot{q}_s) \\ \Omega_2(q_s, \dot{q}_s) \end{bmatrix} = \dots$$
$$\dots = \begin{bmatrix} B_1(q_s) \\ 0 \end{bmatrix} u + \begin{bmatrix} \tilde{\xi}_1(q_s, \dot{q}_s) \\ \tilde{\xi}_2(q_s, \dot{q}_s) \end{bmatrix}. \tag{12}$$

The fragmented model representation is used to express the model as a function of the actuated variables, which is

$$\left[D_{11}(q_s) - D_{12}(q_s) D_{22}^{-1}(q_s) D_{21}(q_s) \right] \ddot{q}_b + \Omega_1(q_s, \dot{q}_s) + \dots$$
$$\dots + D_{12}(q_s) D_{22}^{-1}(q_s) \left[\tilde{\xi}_2(q_s, \dot{q}_s) - \Omega_2(q_s, \dot{q}_s) \right] = \dots$$
$$\dots = B_1(q_s)u + \tilde{\xi}_1(q_s, \dot{q}_s),$$

then, it is simplified to obtain

$$\ddot{q}_b = \kappa(q_s)u + \xi, \tag{13}$$

where

$$\xi = \left[D_{11}(q_s) - D_{12}(q_s) D_{22}^{-1}(q_s) D_{21}(q_s) \right]^{-1} * \dots$$
$$\dots * \left\{ -D_{12}(q_s) D_{22}(q_s)^{-1} \left[\tilde{\xi}_2(q_s, \dot{q}_s) - \Omega_2(q_s, \dot{q}_s) \right] + \dots \right.$$
$$\left. \dots + \tilde{\xi}_1(q_s, \dot{q}_s) - \Omega_1(q_s, \dot{q}_s) \right\} \tag{14}$$

and

$$\kappa(q_s) = \left[D_{11}(q_s) - D_{12}(q_s) D_{22}^{-1}(q_s) D_{21}(q_s) \right]^{-1} B_1(q_s). \tag{15}$$

Based on the simplified model described in the Eq. (13) the following assumptions are proposed.

- First, here and now ξ will be considered as a total disturbances vector, in which the endogenous and exogenous disturbances are lumped.
- Second, the total disturbances ξ and their m derivatives are considered bounded.
- Finally, it is assumed that each component of the vector ξ can be approximated by a time polynomial with order m, so that the $\frac{d^{(m+1)}}{dt^{(m+1)}} \xi_i \approx 0$, $\forall i = 1, 2, 3, 4$.

With the above assumptions, the following control law is proposed:

$$u = \kappa^{-1}(q_s(s)) \left(s^2 q_d(s) - K_{GPI}(q_b(s) - q_d(s)) \right), \tag{16}$$

where q_d is the references vector for the controlled joints and K_{GPI} is a diagonal transfer function matrix with the following structure:

$$K_{GPI} = \begin{bmatrix} K_{GPI_{11}} & 0 & 0 & 0 \\ 0 & K_{GPI_{22}} & 0 & 0 \\ 0 & 0 & K_{GPI_{33}} & 0 \\ 0 & 0 & 0 & K_{GPI_{44}} \end{bmatrix}, \tag{17}$$

and

$$K_{GPI_{ii}} = \frac{\alpha_{i,n+m} s^{n+m} + \dots + \alpha_{i,1} s + \alpha_{i,0}}{s^{m+1} \left(s^{n-1} + \alpha_{i,2n+m-1} s^{n-2} + \alpha_{i,2n+m+1} \right)}, \forall i = 1, \cdots, 4. \tag{18}$$

If the control law in the Eq. (16) is inserted into the Laplace transformation of the simplified model Eq. (13) the close loop system takes the following form:

$$s^2 q_b(s) = \kappa(q_s(s)) \left[\kappa^{-1}(q_s(s)) \left(s^2 q_d(s) \cdots \right. \right.$$
$$\left. \left. \cdots - K_{GPI}(q_b(s) - q_d(s)) \right) \right] + \xi(s), \tag{19}$$

by simplifying this equation we get

$$s^2 q_b(s) = s^2 q_d(s) - K_{GPI}(q_b(s) - q_d(s)) + \xi(s). \tag{20}$$

The Eq. (20) describes the close loop behavior and has a special feature that corresponds to transforming of the system into a decoupled one. This feature is important because, based on that transformation the tracking control problem can be studied as a bunch of Single-Input and Single-Output (SISO) problems, instead of a more complex MIMO system. The fragmentation of the dynamic produces the following system of equations:

$$s^2 q_{b,1}(s) = s^2 q_{d,1}(s) - K_{GPI_{11}}(q_{b,1}(s) - q_{d,1}(s)) + \xi_1(s),$$
$$\vdots$$
$$s^2 q_{b,4}(s) = s^2 q_{d,4}(s) - K_{GPI_{44}}(q_{b,4}(s) - q_{d,4}(s)) + \xi_4(s).$$

Let us define the tracking error as:

$$e_i(s) = q_{b,i}(s) - q_{d,i}(s), \quad \forall i = 1, \cdots, 4,$$

thus, the close loop dynamic takes the following form:

$$s^2 e_i(s) + K_{GPI_{ii}} e_i(s) = \xi_i(s), \quad \forall i = 1, \cdots, 4, \tag{21}$$

and the error's dynamics can be described by:

$$\begin{gathered}\left(s^{2n+m} + \alpha_{i,2n+m-1} s^{2n+m-1} + \cdots + \alpha_{i,1} s + \alpha_{i,0}\right) e_i(s) = \cdots \\ \cdots = s^{m+1} \xi_i(s) \left(s^{n-1} + \alpha_{i,2n+m-1} s^{n-2} + \cdots + \alpha_{n+m+1}\right), \cdots \\ \cdots \forall i = 1, \cdots, 4,\end{gathered} \tag{22}$$

At the right side of the Eq. (22) the product $s^{m+1}\xi_i(s)$ is close to zero, for which the error's dynamics take the following form:

$$\begin{gathered}\left(s^{2n+m} + \alpha_{i,2n+m-1} s^{2n+m-1} + \cdots + \alpha_{i,1} s + \alpha_{i,0}\right) e_i(s) \approx 0 \cdots \\ \cdots \forall i = 1, \cdots, 4,\end{gathered} \tag{23}$$

Finally, the constants α can be arbitrarily chosen in order to impose a dominant dynamic in the error behavior and carry it out asymptotically to zero, which produces a stable close loop dynamic.

4. VIRTUAL HOLONOMIC CONSTRAINT FOR TRAJECTORY GENERATION

Given the control strategy developed in the previous section, it is necessary to propose a trajectory for the tracking control. Between the possibilities to solve this task is the time-based trajectory, which is an unappropriated technique for underactuated robots. And this is so because this strategy does not take into account the possible external disturbances which could cause the instability of the walking. Instead of time-based trajectories, here it is used a state-based trajectory, which has the aim of synchronizing the controlled joints with the underactuated ones. This strategy is called a virtual holonomic constraint because the restrictions are imposed via feedback control in lieu of physical or mechanical constraints, Chevallereau et al. (2009).

The trajectory generation proposed here bases its strategies in producing q_{d_i} as a function of the evolution of $\theta(q)$ and the step length, sl. Here a set of gait patterns were developed for samples of step length between 5 and 24cm. Those gait patterns were used to compute four polynomial regressions which produces 5×5 order polynomials, $q_{d_i}(\theta, sl)$. The relation in the polynomials $q_{d_i}(\theta, sl)$, which describe the references functions are given by the surfaces shown in the Fig. 3.

Few conditions to generate the gait patterns were taken into account to fulfill the unilateral constraint in the support foot reaction forces, bellow they are listed:

- The hip has a sinusoidal movement and the maximum hip high is the 98% of the high of the extended leg
- The minimum high of the hip is a function of the step length, and is chosen adequately to avoid the knee overextension
- The end of the swing leg has a movement starting and ending with a smooth trajectory with a maximum gap over 2.6 cm

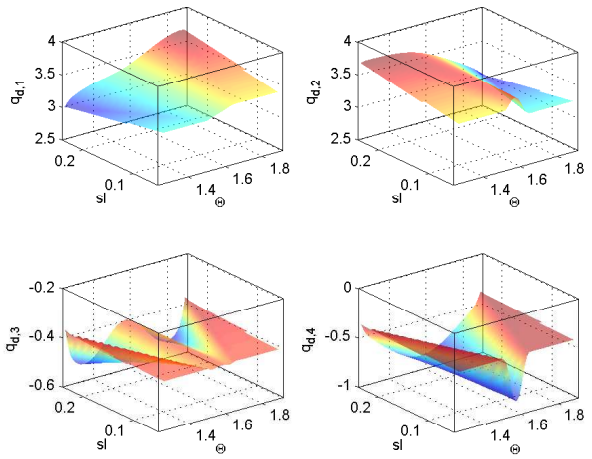


Fig. 3. Trajectory reference surfaces based on $\Theta(q)$ and step length, sl

- To prevent the robot's body take-off from the ground, the vertical reaction in the support leg must be bigger than zero, $F_1^v > 0$
- To avoid sliding of the support leg, its horizontal reaction force must be less than the friction force, $F_1^h < \mu F_1^v$,

5. SIMULATION TEST

A numerical simulation is performed to test the control law proposed, as well as the trajectory planning strategy. The controller tuning was based on the assumption that the lumped disturbances ξ_i can be approximated by five order polynomials, so that $m = 5$. Additional to the approximation of the disturbances, the coefficients α are chosen to achieve a stable and dominant behavior of the error dynamic. This in order to get an asymptotic movement of the controlled joints to the trajectory references. The root to select the α values are listed in the Table 1.

Table 1. Alpha values

	i			
	1	2	3	4
roots	-17.01	-13.09	-16.36	-13.09
	-28.92	-22.25	-27.81	-22.25
	-49.17	-37.83	-47.28	-37.83
	-83.60	-64.31	-80.38	-64.31
	-142.12	-109.32	-136.66	-109.32
	-241.61	-185.85	-232.32	-185.85
	-410.74	-315.96	-394.95	-315.96
	-698.27	-537.13	-671.41	-537.13

The robot walk was tested through the simulation of 50 steps (100 switching feet). The test started with the robot in the following convenient initial conditions.

$$q_s = [\, 3.6200 \;\; 3.1284 \;\; -0.4686 \;\; -0.4593 \;\; -0.0029 \,]^T,$$
$$\dot{q}_s = [\, 0.6985 \;\; -4.6678 \;\; -6.9445 \;\; 3.6576 \;\; 0.1669 \,]^T.$$

The first 22 steps are given with a constant step length of 20 cm. After that, the robot starts to reduce the length of the step until 10 cm, in which it stabilize its gait pattern for the rest of the simulation. Fig. 4 shows the behavior of the controlled joints,

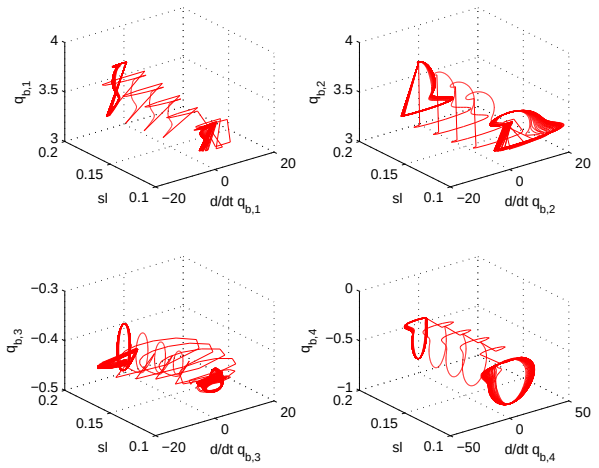


Fig. 4. States' convergence to stable cycles in the changes of step length

there, it is possible to see the evolution from one step length to another, and additionally it is shown the cycles or stable orbits at each step length.

The evolution of the walk is shown in a stick diagram in the Fig. 5. The behavior of the robot's gait with the longest step length is presented in the left side of the Fig. 5, while in the right side it is shown the gait pattern for the step length corresponding to 10 cm.

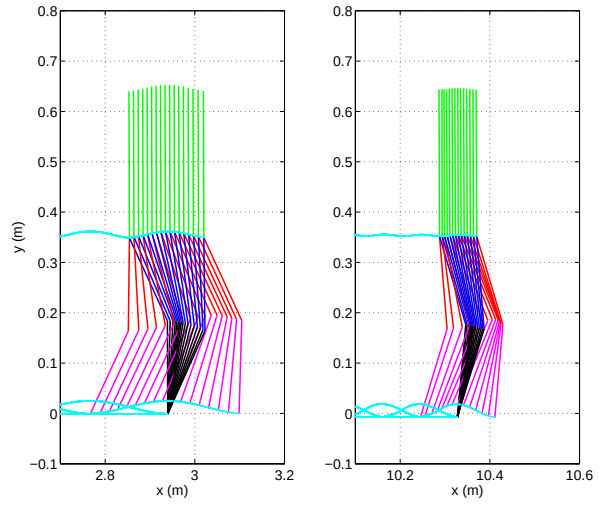


Fig. 5. Evolution of the walk in different gait patterns

5.1 *Evaluation of the gait stability*

Although the stability of the tracking control can be ensure with the selection of the right roots for the error's dynamics, the gait stability depends too of the walking pattern. Several approaches to deal with the global stability of the walking have been proposed, two of the more accepted are: the Zero Moment Point (ZMP) proposed by Vukobratović and Borovac (2004), which has been successful testes with robots with flat feet like ASIMO Sakagami et al. (2002); in the other hand it is the global stability of the dynamic walking of robot with point feet, which has been checked by the search of the existence of a periodic behavior in the robot's states, this methodology was effectively proved by first time in McGeer (1990), which uses the Poincaré's method to search for the presence of a periodic orbit in the evolution of the states of a passive bipedal robot.

Based in the fact that our prototype is a point feet robot, the more convenient approach to analyze the global stability of the robot is the Poincaré's method. But, with the aperiodic walk performed in the simulation test in this section, the Poincaré's method can not be applied without several conditions. Therefore, the method of Poincaré's section is used to identify periodic behavior in the dynamic of the biped at each gait pattern.

The computation of the return map for the each step length is done with the procedure presented in Chen et al. (2007), it is done through the sampling of the states in a time when a condition called the Poincaré's section is satisfied. Poincaré's section could either be a function of the robot states or can be determined by external events. Here the Poincaré's section is determined by the switching foot event, which depends of the ground level.

First, let us define the state vector as $x := [q_s^T \quad \dot{q}_s^T]^T$, then a sampling done in the Poincaré's section is divided in two sets, one for each step length in the simulations, after that, they are used to build a couple of functions that take values just before the impact time and are defined as

$$x_{sl}^-(k+1) = P_{sl}(x_{sl}^-(k)) \qquad \forall sl = 10cm, 20cm, \tag{24}$$

where $x_{sl}^-(k)$ are the k^{th} samples of the state vector just before the impact, $x_{sl}^-(k+1)$ are the state vector just before the next impact and $P_{sl}(x^-(k))$ are functions called the Poincaré return map. The stability test is summarized in the tasks of evaluating the existence of two fix points $x_{sl}^{-^*}$ such that

$$x_{sl}^-(k+1) = P(x_{sl}^{-^*}(k)) = x_{sl}^{-^*} \tag{25}$$

as well as in tasks of demonstrating the discrete systems stability of (24) in the classical sense of Lyapunov for the equilibrium points $x_{sl}^{-^*}$.

The sets of states of the robot in each switching foot are saved in vectors as bellow

$$\chi_{sl}(k) = \begin{bmatrix} x_{sl}^-(k-(\rho-1)) - x_{sl}^{-^*} \\ x_{sl}^-(k-(\rho-2)) - x_{sl}^{-^*} \\ \vdots \\ x_{sl}^-(k) - x_{sl}^{-^*} \end{bmatrix}, \tag{26}$$

where ρ represents the number of switching feet executed in the simulation. Linear regressions are performed between the $\chi_{sl}(k)$ and $\chi_{sl}(k-1)$ values. The result of the regressions give linear approximations of (24) with the following form

$$(x_{sl}^-(k+1) - x_{sl}^{-^*}) = \Phi_{sl} * (x_{sl}^-(k) - x_{sl}^{-^*}) \tag{27}$$

where the eigenvalues of Φ_{sl} must have a magnitude lower than one to guarantee the asymptotic stability of the walking. It was

evaluated and the stability of each gait pattern was proved, so it is possible conclude that the robot has stable walk for both step lengths.

6. CONCLUSION

A time-invariant GPI controller was developed to control a planar bipedal robot with point feet structure. The robot keeps stable walking even with gait pattern changes during the robot is in operation. The controller designed achieves stable walking in the reconfiguration of trajectories and it was confirmed by the Poincaré test at each walking pattern.

The trajectory generation strategy allows switching from one step length to another in a short period of time. The use of the planning trajectories strategy permits the changes between short to long steps and in the opposite way, without the design of a controller for each step length wanted. Thus, the risk to fall in an instability state caused by aperiodic gait, produced by the step length changes, were successfully tackled to ensure the dynamic walking stability.

REFERENCES

Arcos-Legarda, J., Cortes-Romero, J., and Tovar, A. (2016). Active disturbance rejection control based on generalized proportional integral observer to control a bipedal robot with five degrees of freedom. In *American Control Conference (ACC), 2016*. IEEE.

Arcos-Legarda, J., Tovar, A., Cortés, J., Díaz, H., and Sarmiento, L. (2013). Multivariable gpi control of a gait exoskeleton for people with walking disabilities. In *II International Congress of Engineering Mechatronics and Automation (CIIMA)*.

Chen, H. et al. (2007). *Passive dynamic walking with knees: A point foot model*. Ph.D. thesis, Massachusetts Institute of Technology.

Chevallereau, C., Grizzle, J.W., and Shih, C.L. (2009). Asymptotically stable walking of a five-link underactuated 3-d bipedal robot. *Robotics, IEEE Transactions on*, 25(1), 37–50.

Grizzle, J.W., Abba, G., and Plestan, F. (2001). Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. *Automatic Control, IEEE Transactions on*, 46(1), 51–64.

Grizzle, J.W., Chevallereau, C., Sinnet, R.W., and Ames, A.D. (2014). Models, feedback control, and open problems of 3d bipedal robotic walking. *Automatica*, 50(8), 1955–1988.

Grizzle, J., Westervelt, E., and Canudas-de Wit, C. (2003). Event-based pi control of an underactuated biped walker. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 3, 3091–3096. IEEE.

Hamed, K.A. and Grizzle, J.W. (2013). Robust event-based stabilization of periodic orbits for hybrid systems: Application to an underactuated 3d bipedal robot. In *American Control Conference (ACC), 2013*, 6206–6212. IEEE.

Hill, J. and Fahimi, F. (2015). Active disturbance rejection for walking bipedal robots using the acceleration of the upper limbs. *Robotica*, 33(02), 264–281.

Hurmuzlu, Y. and Marghitu, D.B. (1994). Rigid body collisions of planar kinematic chains with multiple contact points. *The international journal of robotics research*, 13(1), 82–92.

Martínez-Fonseca, N., Castañeda, L.Á., Uranga, A., Luviano-Juárez, A., and Chairez, I. (2016). Robust disturbance rejection control of a biped robotic system using high-order extended state observer. *ISA transactions*, 62, 276–286.

McGeer, T. (1990). Passive walking with knees. In *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, 1640–1645. IEEE.

Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N., and Fujimura, K. (2002). The intelligent asimo: System overview and integration. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 3, 2478–2483. IEEE.

Tzafestas, S., Raibert, M., and Tzafestas, C. (1996). Robust sliding-mode control applied to a 5-link biped robot. *Journal of Intelligent and Robotic Systems*, 15(1), 67–133.

Vukobratović, M. and Borovac, B. (2004). Zero-moment point-thirty five years of its life. *International Journal of Humanoid Robotics*, 1(01), 157–173.

Westervelt, E.R., Grizzle, J.W., Chevallereau, C., Choi, J.H., and Morris, B. (2007). *Feedback control of dynamic bipedal robot locomotion*, volume 28. CRC press.

Westervelt, E.R., Grizzle, J.W., and Koditschek, D.E. (2003). Hybrid zero dynamics of planar biped walkers. *Automatic Control, IEEE Transactions on*, 48(1), 42–56.

Graph Transfer Function Representation to Measure Network Robustness

David Martínez * **Eduardo Mojica Nava** **

* *National University of Colombia, Bogotá, Colombia (e-mail: daamartinezva@unal.edu.co).*
** *National University of Colombia, Bogotá, Colombia (e-mail: eamojican@unal.edu.co)*

Abstract: In this work, we use coates determinant to obtain transfer functions from graphs and in this way analyze robustness behaviour of networks against perturbations. Furthermore, we make a comparison between obtained results and values of Sinai-Kolmogorov and Loop Entropies to observe network performance in terms of noise resilience.

Keywords: Sinai-Kolmogorov entropy, Loop entropy, robustness, Information theory.

1. INTRODUCTION

Network robustness and Entropy have been widely associated to compare network configurations and topologies aiming to define which ones are safer against perturbations. This reveals the importance of Shannon Information theory in network analysis. However, most of the Entropy network studies have been based on static network representations, i.e, graphs where Laplacian spectral analysis is not performed. In this work, we explore robustness connection between networks and a pair of measures defined through Laplacian and Adjacency matrices eigenvalues, known as Loop and Sinai-Kolmogorov (S-K) Entropies. To demonstrate how each of them behave against a noise source, we use a tool named Coates determinant and the procedure proposed in Desoer (1960) to represent five different network configurations composed by four nodes and variable spanning tree number as a Transfer function.

1.1 Preliminaries

Shannon Information Theory Information theory formulation introduced by Claude E. Shannon in 1948 through his seminal work (Shannon, 1948), which has been widely studied in many research fields beyond code optimization in Communications, such as Ecology, Microbiology and Biology, deepening specifically in two important but inversely related concepts, Entropy and Mutual Information. The former represents how much uncertainty is generated inside a communication Channel due to external factors (e.g noise), and the later measures how much information is shared between the sender and receiver in a communication process.

To understand the above description consider Fig. 1, which shows a classical representation of a communication channel according to Shannon, where the random variable $\boldsymbol{X}$ represents the delivered message by the sender **A** and

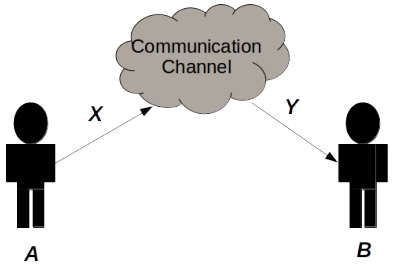


Fig. 1. Channel communications from Shannon perspective.

the the random variable $\boldsymbol{Y}$ denotes the received message in **B**. Mutual information measures how much $\boldsymbol{Y}$ seems to $\boldsymbol{X}$ and Entropy measures the difference between them (uncertainty in **B** about what $\boldsymbol{X}$ is).

To measure Entropy it is necessary to have a probability distribution which describes the likelihood $p(x)$ for every value $\boldsymbol{x}$ that $\boldsymbol{X}$ can take. Then, if we want to measure Entropy of $\boldsymbol{X}$, denoted as $H(X)$, we use the expression

$$H(X) = -\sum_{x} p(x) \log_2 p(x).$$

On the other hand, if we want to measure the Entropy of $\boldsymbol{X}$ once the value of $\boldsymbol{Y}$ is known, we need the conditional probability $p(x|y)$, through which conditional Entropy $H(X|Y)$ is given by

$$H(X|Y) = -\sum_{y} p(y) \sum_{x} p(x|y) \log_2 p(x|y).$$

Finally, to measure how much information about $\boldsymbol{X}$ is contained in $\boldsymbol{Y}$ (mutual information) we use the expression

$$I(X,Y) = H(X) - H(X|Y).$$

The above equations have been used for a long time in communications engineering, however, these have had a big application in population evolution, molecular chains, social behaviours between others, in areas such as Biol-

⋆ Sponsor and financial support acknowledgment goes here. Paper titles should be written in uppercase and lowercase letters, not all uppercase.

ogy, Microbiology, Ecology, Sociology and other fields of Engineering through graph analysis. In next sections we are going to show one of this applications, specifically in network robustness.

Complex Networks and Information Theory Graph and Complex Networks theories have had high relevance in last years in Engineering, specifically in applications such as social and sensor Networks, Internet, and smart grids, to model and design them. On the other hand, there exists a raising interest to involve Information theory definitions, specially Entropy and Mutual Information, to analyze topology, robustness and security from a static point of view, i.e, graphs whose behaviour is not described by spectral analysis but through algebraic mechanisms (Rashevsky, 1955; Solé R. V., 2004). As a result, some new perspectives have appeared to include not only Shannon Entropy but also physical and statistical definitions such as Von Neumann and Kolmogorov-Sinai entropies, to explore network behaviour by means of Laplacian spectral analysis to study dynamics and evolution. The following two sections describe the most important works which relates Entropy, Complex Networks and Graph theories, developed both in statical and dynamical environments.

1.1.2.1. A static perspective As we mentioned before, Entropy requires a probability distribution to be calculated, and in this regard, some methods have appeared. First, Rashevsky (1955) and Trucco (1956) use node dependence or automorphism concept to split the graph in vertex sets, then, dividing the number of nodes in each set by the total amount of nodes in the graph, it is possible to assign a probability for every one. In contrast, in Dehmer (2008) the probability distribution is obtained attributing a probability to each node instead of a set of them, which avoids difficult algorithms necessary to obtain graph automorphisms or vertex partitioning. Other related works have used degree distributions of network topologies such as Scale Free or Random networks to find Entropy values, meanly to describe graph heterogeneity, which is inversely correlated with network symmetry (Xiao et al., 2008).

1.1.2.2. A dynamic perspective A different research have shown ways to obtain Shannon Entropy using spectral graph analysis instead of probability distributions. Some of them have emerged to find similarities between Quantum mechanics concepts (Gibbs and Von Neumann Entropies) and Shannon theory, meanly aiming to employ MaxEnt principle in other research areas such as Ecology, Biology and Physics. First, Passerini and Severini (2008) compute Von Neumann Entropy (an extension of Gibbs Entropy) through a probability distribution taken from the normalized graph Laplacian, demonstrating that graphs with long paths (less quantity of short paths), connected components and regular shapes have higher Entropy values. Secondly, Anand and Bianconi (2009) calculate Gibbs Entropy through graph adjacency matrix and a probability distribution of having connected two nodes. This work shows a way to find a "canonical network ensemble" which satisfies structural constraints as link or node number, i.e, a set of network configurations that can accomplish the same function using the same resources.

In addition to Gibbs and Von Neumann Entropy approaches, Sinai-Kolmogorov Entropy is implemented in Demetrius and Manke (2005) to measure how much skill has a network to reject fluctuations. Its value is calculated from the largest Adjacency matrix eigenvalue (λ) through the following expression:

$$H = log(\lambda). \tag{1}$$

According to Demetrius et al. (2004), network Entropy (H) is directly correlated with the fluctuation decay rate (R), which is defined as:

$$R = \lim_{t\to\infty}\left[-\frac{1}{t}logP_\epsilon(t)\right],$$

where $P_\epsilon(t)$ is the probability that the sample mean [1] deviates more than ϵ from its unperturbed value at time $\boldsymbol{t}$. Therefore, for high values of R, network fluctuations tend to be short, whereas for low values, those tend to be longer. Consequently, since H and R are positively correlated, Entropy is a measure of network robustness [2].

On the other hand, a novel definition, named loop Entropy, has been established in De Badyn et al. (2016), which is given by the following expression:

$$S_\mathcal{G} = \sum_{i=2}^{n} log\lambda_i,$$

where $\lambda_i' s$ are the eigenvalues of the graph Laplacian $L_\mathcal{G}$ and n is the number of nodes. The sum begins from 2 because $\lambda_1 = 0$. There is a set of non negative eigenvalues associated to a sub-matrix $L_\mathcal{G}^{\{n-1\}}$ (Dirichlet Matrix) of $L_\mathcal{G}$, which is obtained by arbitrary elimination of the $\boldsymbol{i}$ row and the $\boldsymbol{i}$ column of $L_\mathcal{G}$. In this sense it is possible to obtain an Entropy expression in terms of this Dirichlet Matix as follows:

$$S_\mathcal{G} = \sum_{i=1}^{n-1} log\lambda_i(L_\mathcal{G}^{\{n-1\}}) + log(n),$$

which results in:

$$S_\mathcal{G} = log\left(n\tau(\mathcal{G})\right), \tag{2}$$

where $\tau(\mathcal{G})$ is the number of spanning trees in the graph. The above result shows a direct relation between Entropy and the number of spanning trees of a graph, which means that the network robustness increases with the number of spanning trees.

1.2 Coates Determinant

Following the procedure in Desoer (1960), it is possible to obtain a transfer function for every graph turning it into a flow graph, which is a directed graph where each node has a self-loop. This process is based in the Coates Determinant, which is given by the next expression:

$$\Delta_c = (-1)^n \sum_p (-1)^{L_\rho} C(G_o)\rho,$$

[1] Sample mean indicates that $\boldsymbol{R}$ is taken in a process where data collection is performed.

[2] Results in Demetrius and Manke (2005) shows that also in this case, Entropy and short paths are inversely proportional, therefore a Scale Free network is more robust than a Regular or Erdös-Renyi case.

where L_ρ is the number of directed loops in the $\rho_t h$ connection, Go is the flow graph G with the source node 0 deleted and $C(Go)$ is the summation of the connection gains (Desoer, 1960).

The next example explains this procedure(De Badyn et al., 2016).
Example: Initially we have the graph in Fig. 2.

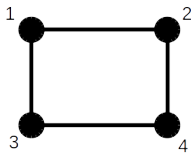


Fig. 2. 4 node graph with 4 spanning trees.

Using the procedure shown in Desoer (1960), we transform the previous graph into that shown in Fig. 3, which represents the denominator for the Transfer Function.

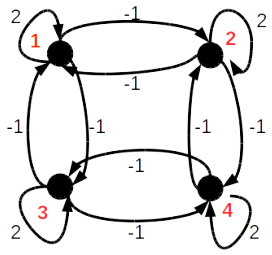


Fig. 3. Denominator graph.

In the same way, Fig. 4 represents the numerator graph for the Transfer Function. Observe that node 1 has a link from an external agent (source node), which means that it is taken as a leader and receives an input connection. In other words, the transfer function is solved for x_1.

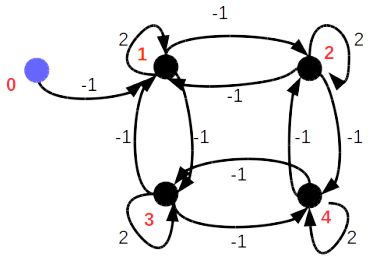


Fig. 4. Numerator graph.

According to Mesbahi and Egerstedt (2010) the ***controlled consensus dynamics*** expression is given by

$$\dot{x} = -L_{\mathcal{G}} + Bu(t) + G\omega(t), \tag{3}$$

where $\omega(t)$ represents a random disturbance to the network, x(t) the states of the nodes, and u(t) is the control input. On the other hand, in De Badyn et al. (2016) is shown an expression that represents a transfer function representation for (3), which has the form:

$$T(s) = \frac{\begin{vmatrix} sI + L_{\mathcal{G}} & -B \\ B^T & 0 \end{vmatrix}}{|\, sI + L_{\mathcal{G}} \,|},$$

where every self-loop of the flow graph has a weight of s+$L_{\mathcal{G}ii}$. Therefore, following the above procedure and using (3) to find Numerator and Denominator expressions as is exposed in Desoer (1960), we obtain the transfer function for the graph of Fig. 2, which is given by

$$T(s) = \frac{s^3 + 6s^2 + 10s + 4}{s^4 + 8s^3 + 20s^2 + 16s}.$$

2. METHODOLOGY

In this section we find Loop and Sinai-Kolmogorov entropy values for 4 different graphs whose spanning tree number varies. Then, using coates determinant and flow graphs representations for each one, we obtain a transfer function representation for each case and show the network behaviour against a disturbance emulated by means of a noise source.

2.1 Transfer Functions

In Fig. 2.1 and Fig. 6 are shown each analyzed graph. Following the procedure exposed in Desoer (1960) and in De Badyn et al. (2016) we find each transfer function, which are presented in Table 1.

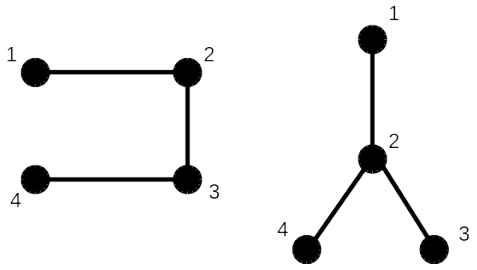


Fig. 5. **Left:** 4 nodes and 1 spanning tree graph.**Right:** The same graph including branches.

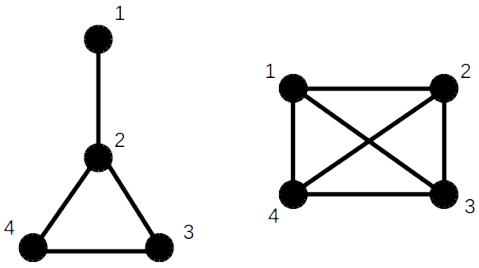


Fig. 6. **Left:** 4 nodes graph with 3 spanning trees. **Right:** complete graph with 4 nodes and 16 spanning trees.

Table 1. Transfer functions for each graph

Graph	Transfer function $T(s)$
4 nodes, 1 spanning tree	$\frac{s^3+5s^2+6s+5}{s^4+6s^3+10s^2+4s}$
4 nodes, 1 spanning tree with branches	$\frac{s^3+5s^2+5s+1}{s^4+6s^3+9s^2+4s}$
4 nodes, 3 spanning trees	$\frac{s^3+7s^2+13s+7}{s^4+8s^3+19s^2+12s}$
4 nodes, 16 spanning trees	$\frac{s^3+9s^2+24s+20}{s^4+12s^3+48s^2+64s}$

2.2 Entropy Values

Using the expressions given by (2) and (1), which corresponds to Loop and Sinai-Kolmogorov entropies respectively, we obtain the values depicted in Tables 2 and 3.

Table 2. Loop-Entropy values for each graph

Graph	Loop-Entropy
4 nodes, 1 spanning tree	1.3863
4 nodes, 1 spanning tree with branches	1.3863
4 nodes, 3 spanning trees	2.4849
4 nodes, 4 spanning trees	2.7726
4 nodes, 16 spanning trees	4.1589

Table 3. S-K Entropy values for each graph

Graph	S-K Entropy
4 nodes, 1 spanning tree	0.4812
4 nodes, 1 spanning tree with branches	0.5493
4 nodes, 4 spanning trees	0.6931
4 nodes, 3 spanning trees	0.7748
4 nodes, 16 spanning trees	1.0986

The results shown in Table 2 indicate that there is no difference in Loop entropy values for graphs with the same spanning tree numbers. It is observable in the case of one spanning tree graph. However, in Table 3 there is a distinction between those values, indicating a difference in robustness in spite of having the same quantity of spanning trees.

2.3 Relation Between Robustness and Spanning Trees Number

As we mentioned in previous sections, entropy and robustness have a direct relation. In this section we probe this assumption observing perturbation responses of each transfer function, demonstrating that almost in all cases, those networks with more spanning trees have better performances against noise. However, some cases where networks have more connections and therefore more spanning trees, do not present good performances in terms to noise response as we can observe in Figs. 7, 8, 9, 10 and 11. On the other hand, it is possible to note that the Kolmogorov-Sinai Entropy describes more precisely robustness in graphs, which is visible if we compare the data in Table 3 with results shown in the next set of figures. Besides of this, we can note that a graph with three spanning trees has better response to noise than a graph with four spanning trees and that those graphs with more connections tend to have unfavourable noise responses in spite of having high entropy values.

3. CONCLUSIONS

The results of this work have shown that Sinai-Kolmogorov Entropy is more precise than Loop entropy to elucidate

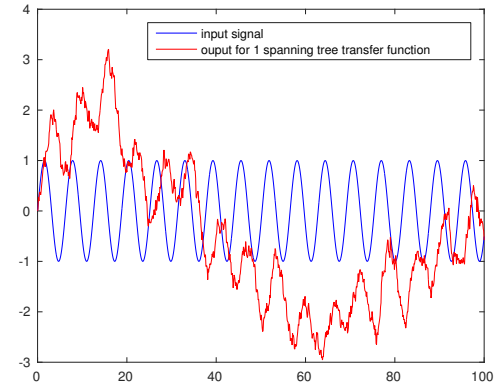


Fig. 7. Input vs. response for a 4 nodes and 1 spanning tree.

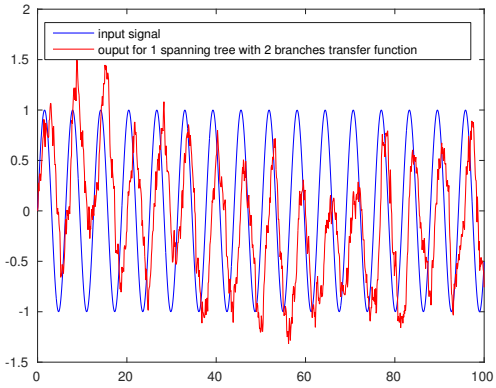


Fig. 8. Input vs. response for a 4 nodes and 1 spanning tree with 2 branches.

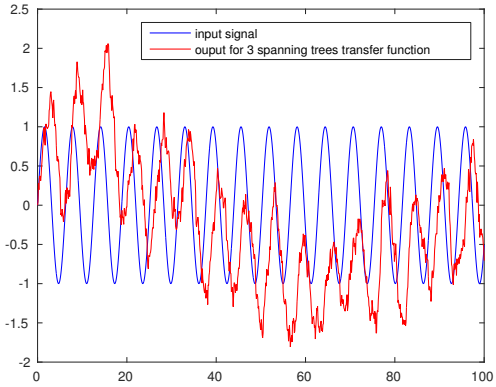


Fig. 9. Input vs. response for a 4 nodes and 3 spanning trees.

how much robustness has a network, as we can note comparing the case of a graph with two branches and the cycle graph, both with four nodes and one spanning tree.

On the other hand, it is possible to observe that in spite of having more spanning trees and therefore more node connections, some networks do not present good noise responses, which could be associated to the increase of paths for its spreading. In contrast, networks with

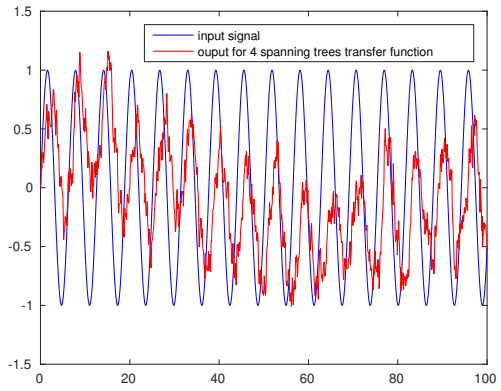


Fig. 10. Input vs. response for a 4 nodes and 4 spanning trees.

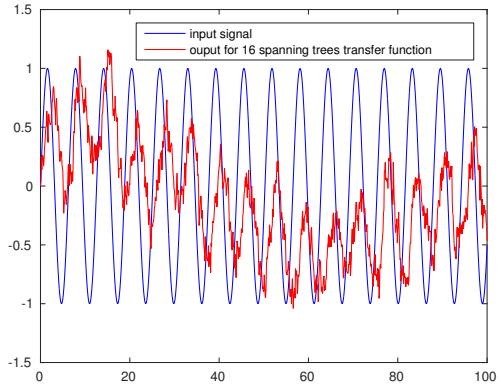


Fig. 11. Input vs. response for a 4 nodes and 16 spanning trees.

branches and low number of spanning trees can offer good possibilities to improve noise responses.

REFERENCES

Anand, K. and Bianconi, G. (2009). Entropy measures for networks: Toward an information theory of complex topologies. *Physical Review E*, 80(4), 045102. doi: 10.1103/PhysRevE.80.045102.

De Badyn, M.H., Chapman, A., and Mesbahi, M. (2016). Network entropy: A system-theoretic perspective. In *Proceedings of the IEEE Conference on Decision and Control*, volume 2016-Febru, 5512–5517. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/CDC.2015.7403083.

Dehmer, M. (2008). A NOVEL METHOD FOR MEASURING THE STRUCTURAL INFORMATION CONTENT OF NETWORKS. *Cybernetics and Systems*, 39(8), 825–842. doi:10.1080/01969720802435925.

Demetrius, L., Gundlach, V.M., and Ochs, G. (2004). Complexity and demographic stability in population models. *Theoretical population biology*, 65(3), 211–25. doi:10.1016/j.tpb.2003.12.002.

Demetrius, L. and Manke, T. (2005). Robustness and network evolution?an entropic principle. *Physica A: Statistical Mechanics and its Applications*, 346(3-4), 682–696. doi:10.1016/j.physa.2004.07.011.

Desoer, C. (1960). The Optimum Formula for the Gain of a Flow Graph or a Simple Derivation of Coates' Formula. *Proceedings of the IRE*, 48(5), 883–889. doi: 10.1109/JRPROC.1960.287625.

Mesbahi, M. and Egerstedt, M. (2010). *Graph theoretic methods in multiagent networks*.

Passerini, F. and Severini, S. (2008). The von Neumann entropy of networks. 5. doi:10.4018/978-1-60960-171-3.ch005.

Rashevsky, N. (1955). Life, information theory, and topology. *The Bulletin of Mathematical Biophysics*, 17(3), 229–235. doi:10.1007/BF02477860.

Shannon, C.E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3), 379–423. doi:10.1002/j.1538-7305.1948.tb01338.x.

Solé R. V., V.S. (2004). *Complex Networks*, volume 650 of *Lecture Notes in Physics*. Springer Berlin Heidelberg, Berlin, Heidelberg. doi:10.1007/b98716.

Trucco, E. (1956). A note on the information content of graphs. *The Bulletin of Mathematical Biophysics*, 18(2), 129–135. doi:10.1007/BF02477836.

Xiao, Y.H., Wu, W.T., Wang, H., Xiong, M., and Wang, W. (2008). Symmetry-based structure entropy of complex networks. *Physica A: Statistical Mechanics and its Applications*, 387(11), 2611–2619. doi: 10.1016/j.physa.2008.01.027.

Greenhouse Temperature Modeling and Control Based on Timed Continuous Petri Nets

J. Fraustro-Valdez *, **G. Desirena-López** *, **J. Ruiz-León** *, **O. Begovich** *

* *CINVESTAV-IPN Unidad Guadalajara, Av. del Bosque 1145, Colonia el Bajío, Zapopan, 45019, Jalisco, México.*

Abstract
This work deals with the modeling and climate control of greenhouses. In order to achieve these objectives, this paper extends previously reported greenhouse modeling methodologies and proposes a control strategy. More precisely, the model methodology is based on a modular technique using timed continuous Petri nets, and the proposed discrete control law is based on the contribution degree of controllable transitions. The model and control methodology aim to contribute by simplifying the manipulation of greenhouse variables. Thus, the possible human errors introduced during the modeling stage or operation stage are reduced. A simulation example is presented in order to illustrate the modelling methodology and the control design and its application.

Keywords: Hybrid systems, Timed Continuous Petri Net, Temperature control, Greenhouse, Modeling, Control, Product semantic, Contribution degree.

1. INTRODUCTION

Nowadays, agricultural production has managed to meet demands from a rapid growth of the world population. In particular, urban farming is getting relevant. Nevertheless, the development of crops in farms or city-farms needs specific knowledge about a greenhouse and its operation. This becomes difficult because of the complexity of the environment and the understanding of multi-variables relation involved in this kind of systems. However, with a graphical representation and an efficient modular modeling for greenhouses, it could be possible to simplify the understanding and the control of these systems.

Some advantages can be obtained when climate control is implemented in a greenhouse. Using a control methodology, it is possible to reduce risks due to randomness of weather indoor greenhouse and provide the optimal environment conditions for growing plants. In order to obtain the optimal plant environment, it is essential to automatically adjust the environment factors (temperature, humidity, light, CO_2, etc). The control of greenhouse climate is characterized by the fact that several processes, such as crop growth and greenhouse climate change, occur on different time scales. The development of the crop occurs on time scale of weeks or months, whereas most of the greenhouse climate variables change on a daily basis. Both greenhouse climate and crop growth are influenced by light, which may change on a time scale of seconds or minutes, specially on cloudy days. To obtain optimal conditions in indoor greenhouses, several mathematical models have been developed to predict the growing conditions as a function of the micro-climate variables. Most of the existing models are based on mass-energy balances. For example, in (Boulard and Baille, 1993) a greenhouse model, including natural ventilation and evaporative cooling, is presented. The authors use heat and mass balance equations to derive the model. However, this work includes a linearization stage, and the model is valid only around the operating point. In (Cunha et al., 2003), neural networks are used to deal with the linear and nonlinear identification of the behavior of a greenhouse. This method uses a large amount of data samples due to its large number of synthesis parameters, and it also requires a large computation time for training the neural network.

Greenhouse control devices are in general on-off (events) while climate variables (temperature, humidity, solar radiation, etc.) are continuous; thus, a greenhouse model represents a hybrid system. Hybrid systems, such as greenhouses, can be modeled by Discrete Event Systems (DES) using Petri Nets (PN) with a big population. Petri nets (Dessel and Esparza, 1995) are a formalism for the modeling and analysis of discrete dynamic systems. Unfortunately, the problem of modeling dynamic systems such as real systems cannot be applied in heavily marked Petri nets. However, a technique used to overcome this problem is to relax the system by *fluidification* obtaining the Timed Continuous Petri Nets ($TCPN$). $TCPN's$ are a relaxation of the Petri nets where the marking becomes continuous and the state equation is represented by a set of positive and bounded linear differential equations, which depend on the semantic of the net. Several $TCPN$ models have been proposed in the literature, for instance (Ross-León et al., 2014) for biological systems, (Tovany et al., 2013) for a greenhouse, (Desirena-Lopez et al., 2014) considers ther-

mal systems, (Júlvez and Boel, 2010) for traffic systems and (Mahulea et al., 2012) for manufacturing systems.

In this paper, in order to represent the internal environment (climate variables), the continuous part of the greenhouse is modeled by $TCPN$ using product semantics ($TCPN - PS$). This semantic has been widely used to model biological systems in (Heiner et al., 2008), (Silva and Recalde, 2002), (Baldan et al., 2011), (Tovany et al., 2013), where the nonlinear part is introduced by the join transitions (Silva and Recalde, 2004), (Garcia-Malacara et al., 2013). For these reasons, the incremental model methodology from (Tovany et al., 2013) is used to represent the temperature and humidity behaviour adding a thermal actuator to the greenhouse model. On the other hand, an On/Off local control law is presented. In (Ross-León et al., 2014) the *contribution degree* is defined as the dot product between the marking error, i.e. the difference between the required and the actual markings, and the columns of the incidence matrix. Thus, if the result is a positive scalar value, the firing of these transitions contributes to reduce the marking error for $TCPN$ models with big populations and all transitions are controllable. Based on this, the control strategy proposed by (Ross-León et al., 2014) is used and applied to $TCPN - PS$ model with controllable and uncontrollable transitions. A simulation example is presented to illustrate the modelling methodology, the control design and its application.

The paper is organized as follows. In Section 2, basic concepts related to Timed Continuous Petri Nets ($TCPN's$) systems and controllability concepts are presented. Section 3 presents a Petri net modeling procedure for greenhouses. Section 4 deals with the problem of reaching a required state and synthesizes a Lyapunov function for solving this problem. Finally, Section 5 presents an example of a greenhouse model as a case of study and the temperature control using the proposed control law.

2. FUNDAMENTALS

In this section, we provide the basic background on Petri nets (PN) and Timed Continuous Petri Nets ($TCPN$) under product server semantics (PS). Notice that $TCPN's$ under PS belong to the class of positive Nonlinear Systems (Murata, 1989), (Silva and Recalde, 2002).

2.1 Continuous and Timed continuous Petri nets

Definition 1. A Continuous Petri Net ($ContPN$) is the pair $ContPN = (N, m_0)$ where $N = (P, T, Pre, Post)$ is a Petri net (PN) and $m_0 \in \{\mathbb{R}^+ \cup 0\}^{|P|}$ is the initial marking, $P = \{p_1, \ldots, p_n\}$ and $T = \{t_1, \ldots, t_k\}$ are finite sets of elements named places and transitions, respectively. Pre, $Post \in \{\mathbb{N} \cup 0\}^{|P|\times|T|}$ are the Pre- and Post- incidence matrices, where $Pre[i, j]$ ($Post[i, j]$) represents the weight of the arc going from p_i to t_j (from t_j to p_i).

The $ContPN$ evolution rule is different from the case of discrete PN systems, since in continuous $PN's$ the firing of an enabled transition is not restricted to integer values; thus the marking $m \in \{\mathbb{R}^+ \cup 0\}^{|P|}$ is not forced to be integer. Instead, a transition t_i in a $ContPN$ is *enabled* at marking m iff for every $p_j \in \bullet t_i, m[p_i] > 0$; and its *enabling degree* depends on the continuous PN semantic. The firing of the enabled transition t_i in a certain amount $\alpha_i \leq enab(t_i, m)$ leads to a new marking $m' = m + \alpha_i C[P, t_i]$, where C is the incidence matrix, defined by $C = Post - Pre$.

If m is reachable from m_0 by firing each enabled transition t_i in a certain amount $\sigma_i \leq enab(t_i, m)$, then $m = m_0 + C\overrightarrow{\sigma}$ and it is named the fundamental equation where $\overrightarrow{\sigma} \in \{\mathbb{R}^+ \cup 0\}^{|T|}$ is the firing count vector.

Definition 2. A P-invariant (P-semiflow) of a net N is a rational-valued solution Y of the equation $Y^T \cdot C = 0$.

Proposition 3. Let (N, m_0) be a system, and let I be a P-invariant of N. If $m_0 \xrightarrow{*} m'$, then $I \cdot m = I \cdot m_0$ (Dessel and Esparza, 1995).

Definition 4. A T-invariant (T-semiflow) of a net N is a rational-valued solution of the equation $C \cdot X = 0$.

Proposition 5. Let σ be a finite sequence of transitions of a net N which is enabled at marking m. Then the *Parikh* vector $\overrightarrow{\sigma}$ is a T-invariant iff $m \xrightarrow{\sigma} m$ (i.e., iff the occurrence of σ reproduces the marking m) (Dessel and Esparza, 1995).

The set of all reachable markings from m_0 is called the reachability set and it is denoted by $RS(N, m_0)$.

A $ContPN$ is bounded when every place is bounded, i.e. $\forall p \in P, \exists b_p \in \mathbb{R}$ s.t $m[p] \leq b_p$ at every reachable marking m, and it is live when every transition is live (it can ultimately be fired from every reachable marking).

Definition 6. A Timed Continuous Petri Net ($TCPN$) is a time-driven continuous-state system described by the 3-tuple (N, λ, m_0) where (N, m_0) is a continuous PN and the vector $\lambda \in \{\mathbb{R}^+ \cup 0\}^{|T|}$ represents the transition rates that determine the temporal evolution of the system. Transitions fire according to certain speed, which generally is a function of the degree rates and the instantaneous marking. Such function depends on the semantics associated to the transitions. Under infinite server semantics the flow (the transitions firing speed, denoted as $f(m)$) through a transition t_i is defined as the product of the rate, λ_i, and $enab(t_i, m)$, the instantaneous enabling of the transition, i.e., $f_i(m) = \lambda_i enab(t_i, m) = \lambda_i min_{p\in\bullet t_i} \frac{m[p_j]}{Pre[p_j,t_i]}$.

The maximum transition firing rate matrix is denoted by $\Lambda = diag(\lambda_1, ..., \lambda_{|T|})$. When the flow is well defined, every continuous transition must have at least one input place, hence in the following we will assume $\forall t \in T, |^\bullet t| \geq 1$. The "$min$" in the above definition leads to the concept of configuration. A configuration of a $TCPN$, at marking m, is a set of (p, t) arcs describing the effective flow going through each transition. For each configuration, a configuration matrix is defined as follows:

$$\Pi(m) = \begin{cases} \frac{1}{Pre[i,j]} & \text{if } p_i \text{ is constraining } t_j \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The flow through the transitions can be written in a vectorial form as $f_k(m) = \Lambda\Pi_k(m)m$. The dynamical behavior of a $TCPN$ system is described by its state equation:

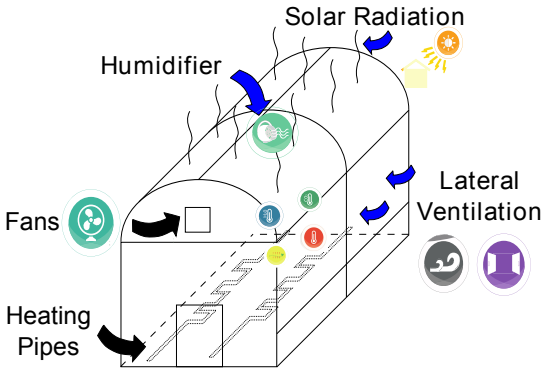


Figure 1. Greenhouse System.

$$\dot{m} = C\Lambda\Pi_k(m)m. \tag{2}$$

Under the product server semantic the flow f_i of a transition t_i is computed as

$$f_i(\tau) = \lambda_i \cdot \prod_{p \in \bullet t_i} \left\{ \frac{\mathbf{m}(p)}{\mathbf{Pre}[p, t_i]} \right\}. \tag{3}$$

Notice that $m(p)$ depends on time τ.

In product semantics, the flow through a transition t_i is the product of the marking in the input places of t_i.

The dynamical behavior of a $TCPN$ system is described by its state equation

$$\dot{m} = Cf(\tau) \tag{4}$$

where $f(\tau) = [f_1, ..., f_{|T|}]^T$.

A transition t_i in a $TCPN$ system is controllable if its flow can be reduced or stopped. In order to apply a control action in (4), a subtracting term u, such that $0 \leq u_i \leq f_i$ is added to every controllable transition t_i to indicate that its flow can be reduced. This control action is adequate because it captures the real behavior that the maximum device throughput can only be reduced. Thus, the controlled flow of transition t_i becomes $w_i = f_i - u_i$. Then, the forced state equation is given by

$$\begin{gathered} \dot{m} = C[f - u] = Cw \\ 0 \leq u_i \leq f_i. \end{gathered} \tag{5}$$

In order to obtain a simplified version of the state equation, the input vector u is rewritten as $u = I_u\Lambda \cdot \prod_{p \in \bullet t_i} \left\{ \frac{\mathbf{m}(p)}{\mathbf{Pre}[p,t_i]} \right\}$, where $\Lambda_{i,i} = \lambda_i$ (the firing rate of transition t_i) and $\Lambda_{i,j} = 0$, $i \neq j$ and $I_u = diag(I_{u_1}, \ldots, I_{u_{|T|}})$ with $0 \leq I_{u_i} \leq 1$ (notice that $0 \leq u_i \leq f_i$, as required in Equation (5)). Then the matrix $I_c = I - I_u$ is constructed and the state equation can be rewritten as

$$\dot{m} = CI_cf = Cw \tag{6}$$

where $I_{c_{i,i}} \in [0, 1]$ represents the control action, i.e. it represents the percentage of maximum flow allowed to go through the transitions.

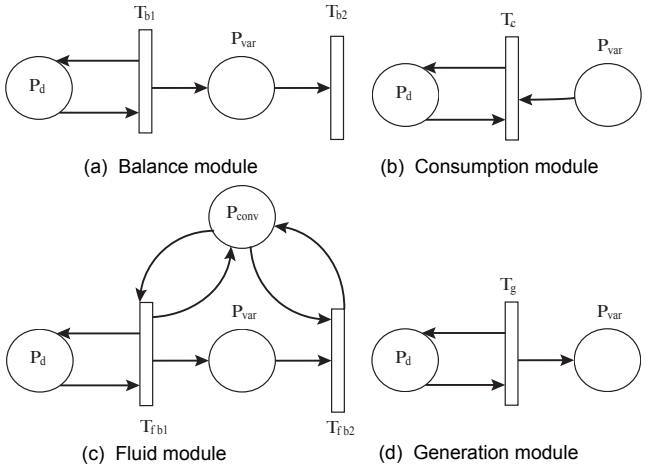


Figure 2. Elementary Modules.

3. MODELING METHODOLOGY

A greenhouse is a system composed by a building which isolates the crop from the outside environment, a set of devices such as humidifiers, coolers, fans, automatic windows, etc., allowing to adequate the indoor greenhouse climate and an irrigation system conveying water and fertilizers into the greenhouse. The inside temperature of the greenhouse is affected by external disturbances, such as solar radiation, outdoor temperature, wind speed, etc., as depicted in Fig. 1.

The indoor greenhouse temperature can be controlled by the signals turning on, or tuning off the greenhouse devices; thus, a set of events $E_g = \{a_1, \bar{a_1}, ..., \bar{a_n}\}$, which represent these signals, is defined.

The modeling methodology proposed in (Tovany et al., 2013) is incremental and uses elementary $TCPN$ modules. A Petri net elementary module is created for each device and disturbance. The elements of heat and mass balance equations can be modeled separately into: variables to be controlled, devices for control and external perturbation variables. Therefore, instead of using the $TCPN$ as a formalism to model a greenhouse, it is better to model with these elementary modules: **Generation module**, **Consumption module**, **Balance module** with infinite server semantic and **Fluid balance module** (see Fig. 2) with product semantic. The initial marking and transition firing rates are given by the parameters of plastic film and devices capacities or some of them can be measured by a meteorologic station and the rest of them can be estimated by a regression algorithm. Afterwards, these models are merged to form a single model.

Notice that the final model is a hybrid system. There is a set of differential equations to represent the indoor greenhouse temperature and a set of events to control it.

An elementary $TCPN$ module is called **balance module** as depicted in Fig. 2 (a) if it has a place p_{var} that is assigned with a marking m_{var} representing a variable x; a function p_d with a marking m_d is assigned to represent a function $d(\tau)$; and a transition t_c is created with the maximum flux λ_c=k representing a gain factor.

Module	p/Var	Transition
$q^{rad}(I_o)$	m_5	$t_1 = \eta_g$
$q^{vent}(T_o, v, T_g)$	m_6 - m_{10} - m_1	$t_2 = \rho_a c_{p,a}$ y
		$t_5 = \rho_a c_{p,a}$
$q^{cond}(T_o, T_g)$	m_6 - m_1	$t_3 = U_{g,o}$ y
		$t_6 = U_{g,o}$
$q_{p,s}(T_s, T_g)$	m_2- m_1	$t_4 = U_{g,s}$ y
		$t_7 = U_{g,s}$
$q(Fhum)$	m_{11} - m_1	$t_{10} = \gamma\rho_{H_2O}$
		$(1-k_1)U_{hum}/A_g$
$q(\varphi^{cons}_{H_2O_{g,r}})$	m_{12} -m_1	$t_{11} = \gamma(1-k_2)$
$q(T_g, T_s)$	m_1 - m_2	$t_8 = U_{g,s}$ y $t_9 = U_{g,s}$
$q(T_{ss}, T_s)$	m_7 - m_2	$t_{12} = U_{sss}$ y $t_{13} = U_{sss}$
$q(F_{hum}, C_{H_2O})$	m_{11} - m_3	$t_{16} = \rho_{ag}(1-k_1))/A_g$
$q(C_{H_2O_{g,r}}, v, C_{H_2O})$	m_8 - m_{10} - m_3	$t_{14} = 1$ y
		$t_{15} = 1$
$q(\varphi^{cons}_{H_2O_{g,r}}, C_{H_2O})$	m_{12} - m_3	$t_{17} = 1$
$q(CO_{2_{g,r}}, v, CO_2)$	m_9- m_{10} - m_4	$t_{18} = 1$ y $t_{19} = 1$
$q^{pipe}_{p,g}(T_p, T_g)$	m_1 - m_{13}	$t_{20} = t_{21} = U_{p,g}$
		$t_{22} = t_{23} = U_{p,g}$
		$t_{24} = \rho_{H_2O} c_{p,H_2O}$
		F_p/A_g
		$t_{25} = \eta_p$

Table 1. Relation between variables and function places.

An elementary $TCPN$ module is called **consumption module** as depicted in Fig. 2 (b) if it has a place P_{var} that is assigned with marking m_{var} representing a variable x; a function place p_d with a marking m_d is assigned to represent a function $d(\tau)$; and a transition t_c is created with a maximum flux λ_c representing the gain factor.

An elementary $TCPN - PS$ module is called **fluid balance module** as depicted in Fig. 2 (c) if it is a representation of a balance element with proportional fluid flow relation. It is based on a balance module with an extra place p_{conv} with marking m_{conv} representing the fluid flow.

Finally, an elementary $TCPN$ is called **generation module** as depicted in Fig. 2 (d) if it has a place P_{var} that is assigned with marking m_{var} representing a variable x; a function place p_d with marking m_d is assigned to represent a function $d(\tau)$; and a transition t_g is created with maximum flux $\lambda_g = k$ representing a gain factor.

According to this procedure, first we have to associate places for the involved variables: greenhouse temperature T_g (P_1), soil temperature T_s (P_2), C_{H_2O} concentration (P_3), $C_{C_{CO_2}}$ concentration (P_4) and pipe system temperature T_p (P_{13}). Function places are associated with the variables: solar radiation I_o (P_5), outside temperature T_o (P_6), subsoil temperature T_{ss} (P_7), outside water vapor concentration $C_{H_2O_o}$ (P_8), outside carbon dioxide concentration $C_{CO_{2_o}}$ (P_9), wind speed V (P_{10}), humidifier maximum water flow F_{hum} (P_{11}), water condensation φ_{cons} (P_{12}) and the maximum water flow inside the pipes F_p (P_{14}). Secondly, for the greenhouse temperature T_g we construct a generation module for solar radiation $q^{rad(I_o)}$ and condensation φ_{cons} q^{cons}; a consumption module for the humidifier F_{hum} (q^{hum}); a balance module for soil temperature T_s (q^{T_s}); and a fluid balance module for controlled natural ventilation (q^{vent}). For the greenhouse humidity C_{H_2O} we construct: a generation module for the humidifier F_{hum} (φ^{hum}); a consumption module for condensation φ_{cons} (φ^{cond}); a balance module for outside humidity C_{H_2O}; and fluid balance module for controlled natural ventilation (φ^{vent}).

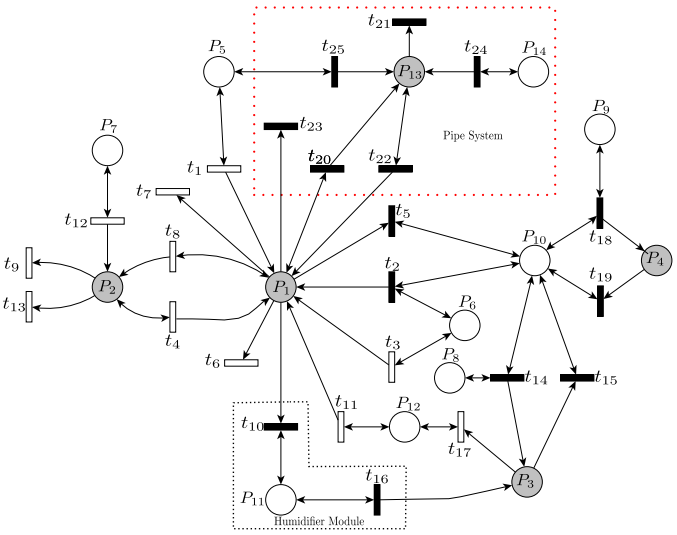


Figure 3. Greenhouse TCPN modeling.

Since soil temperature is also modeled, we construct a balance module for the greenhouse temperature T_g and soil temperature T_s. Then, we construct modules for each variable as shown in Table 1 (where parameter values are taken from (Van Straten et al., 2010)).

Then, by merging these modules we obtain the greenhouse model in $TCPN - PS$ as depicted in Fig. 3. The state equation is represented as follows:

$$\begin{aligned}
\dot{m}_1 =& (-I_{c1}\lambda_5 m_1 m_{10} - (\lambda_7 + \lambda_6 + \lambda_{23})m_1 + \lambda_4 m_2 \\
&+ (I_{c1}\lambda_2)m_6 m_{10} + \lambda_3 m_6 - I_{c2}\lambda_{10} m_{11} \\
&+ \lambda_{22} m_{13} + \lambda_{11} m_{12} + \lambda_1 m_5)/K_g \\
\dot{m}_2 =& (\lambda_8 m_1 - (\lambda_9 + \lambda_{13})m_2 + \lambda_{12} m_7)/K_s \\
\dot{m}_3 =& (-I_{c1}\lambda_{15} m_3 m_{10} + I_{c1}\lambda_{14} m_8 m_{10} \\
&+ I_{c2}\lambda_{16} m_{11} - \lambda_{17} m_3)[A_g/V_g] \\
\dot{m}_4 =& (-I_{c1}\lambda_{19} m_4 m_{10} + I_{c1}\lambda_{18} m_9 m_{10})[A_g/V_g] \\
\dot{m}_{13} =& (\lambda_{20} m_1 - \lambda_{21} m_{13} + \lambda_{24} m_{14} + \lambda_{25} m_5)/K_p.
\end{aligned} \tag{7}$$

4. CONTROL LAW

In this subsection, some controllability concepts of $TCPN$ are recalled from (Vázquez et al., 2008).

An induced equilibrium point is a marking m_a in which the system can be maintained using a specific control action u_a, i.e. $0 = C[f - u_a]$. A maximum throughput equilibrium point m_r can be computed. Thus, an important problem is that given an initial marking m_0 and a required target marking m_r (for instance, markings allowing maximum system throughput), to design a control law leading the $TCPN$ state from m_0 to m_r. This problem is formally defined as follows.

Definition 7. Let $(N; m_0; \lambda)$ be a $TCPN$ system and $(m_r; I_{cr})$ be an induced equilibrium point of the $TCPN$. Then the Regulation Control Problem (RCP) in $(m_r; I_{cr})$, denoted as $RCP(m_r; I_{cr})$, deals with the computation of

a control law $I_c(\tau)$; $0 \leq \tau \leq \tau_f$ feasible in the $TCPN$ such that $m(\tau_{ss}) = m_r$ and $I_c(\tau_{ss}) = I_{cr}$, $\forall \tau_{ss} \geq \tau_f$.

The greenhouse control problem is stated as a RCP, where the equilibrium point is given by target indoor temperature and the computed I_c. Since the set $E_g = \{a_1, \bar{a_1}, ..., \bar{a_n}\}$ is representing a set of events, then the control value of I_c in Equation (6) must be zero or one.

Notice that the modules in the modeling methodology are of type balance, generation or consumption module. Generation modules (solar radiation, condensation of moisture on walls and roof) increase the indoor temperature, while consumption modules reduce it. For fixed environmental conditions if the greenhouse heaters are turned on and coolers are turned off, then the indoor temperature will reach a maximum value, named T_{max}; otherwise if the greenhouse heaters are turned off and coolers are turned on, then the indoor temperature will reach a minimum value, named T_{min}. Thus the indoor temperature can be controlled in the range $T_R = [T_{min}, T_{max}]$.

The control law solving the RCP problem will be obtained based on the results from (Ross-León et al., 2014), which considers the control problem in live and bounded $TCPN$, with controllable transitions and infinite server semantic. In this work, the implementation of the control strategy to nonlinear hybrid systems is presented. For this reason, the product semantics is used to represent the non linearity of the system.

Definition 8. Let (N, m_0, λ) be a live and bounded $TCPN$ system and $m_r \in int(T_R)$ be the required target marking. Then the marking error is computed as

$$e(\tau) = m_r - m(\tau). \tag{8}$$

Definition 9. Let (N, m_0, λ) be a live and bounded $TCPN$ system. The **Contribution Degree** $\Psi_k(\tau)$ of a transition t_k of N is defined as

$$\Psi_k(\tau) = e^T(\tau)c_k \tag{9}$$

where c_k is the k-th column of the incidence matrix C.

Notice that the contribution degree is representing the projection (dot product) of how the transition removes/adds tokens (column c_k) over the error. Thus, it provides information about how much a transition is capable of decreasing the error marking.

Lemma 10. (Ross-León et al., 2014) Consider the definition of the marking error and let (N, m_0, λ) be a live and bounded $TCPN$ system and v a T-semiflow of N. Then $\sum_{t_k \in |v|} v_k \Psi_k(\tau) = 0$.

Proof. The result follows from

$$\sum_{t_k \in |v|} v_k \Psi_k(\tau) = e^T(\tau) C v = 0.$$

From Lemma 10 we have that the sum of every contribution degree in a given T-semifow is zero. Therefore, there must be both transitions with positive and transitions with negative contribution degrees in it.

Theorem 11. (Ross-León et al., 2014) Let (N, m_0, λ) be a live and bounded $TCPN$ system where all transitions are controllable. Then, the control law

$$I_{c_k} = \begin{cases} 1 & \text{if } \Psi_i > 0 \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

leads the initial marking $m_0 = m(0)$ to the required marking $m_r \in int(T_R)$.

Proof. Let $V(\bullet)$ be the function

$$V(e) = \frac{1}{2} e^T(\tau) e(\tau). \tag{11}$$

We claim that $V(e)$ is a Lyapunov function, i.e. it is positive definite, its derivative is negative definite and it fulfils $\dot{V}(x) = 0$ only for $x = 0$, as shown below.

Clearly, (11) is positive definite. Now, the differentiate of $V(e)$ is

$$\dot{V}(e) = e^T(\tau)\dot{e} = -e^T(\tau) C I_c f(\tau). \tag{12}$$

Since $m_r \in int(T_R)$, then for every time τ there exists a transition firing vector $\sigma(\tau) > 0$ such that $e(\tau) = m_r - m(\tau) = C\sigma(\tau)$. Thus

$$e^T(\tau)e(\tau) = e^T(\tau) C \sigma(\tau) > 0 \tag{13}$$

for $e(\tau) \neq 0$. Rewriting (13) we have

$$\begin{aligned} e^T(\tau) C \sigma(\tau) &= e^T(\tau)[\sigma_1 c_1 + \sigma_2 c_2 + \cdots + \sigma_{|T|} c_{|T|}] \\ &= \sigma_1 e^T(\tau) c_1 + \sigma_2 e^T(\tau) c_2 + \cdots + \sigma_{|T|} c_{|T|} > 0. \end{aligned} \tag{14}$$

Also σ_i depends on τ. Since $\forall i, \sigma_i \geq 0$, then there exists at least one c_k such that $\sigma_k e^T(\tau) c_k > 0$ for $e(\tau) \neq 0$. Notice that $\sigma_k > 0$ is strictly positive, thus $e^T(\tau)c_k > 0$ is also strictly positive. Then from (10) we have that $I_c(k) = 1$. Substituting these values in (12), then $\dot{V}(e)$ could be zero or negative. If Equation (12) is negative, then the derivative of the Lyapunov function is negative, decreasing the system error.

However, Equation (12) could be zero when the error is different from zero iff $f(\tau) = 0$ or $I_c f(\tau) = 0$ or $I_c f(\tau) \neq 0$ is in the kernel of C or $CI_c f(\tau)$ is in the kernel of e^T. Now, we will show that these cases are not possible.

In the first case if $f(\tau) = 0$ means that the $TCPN$ system is blocked, then it is not live; a contradiction.

In the second case $I_c f(\tau) = 0$, then the flow of transitions is being blocked by I_c since its contribution degree is negative or the contribution degree is positive and I_c is not blocking, but the flow of these transitions is zero. Since the $TCPN$ is live and bounded, then by Lemma 10 there are positive and negative contribution degrees. Then there is an $I_c(k) = 1$ with its corresponding flow $f_k = 0$. Thus, at least one input place to t_k has zero tokens, meaning that the marking m is in the border, i.e. it is not in $int(T_R)$, and $\dot{V} = 0$. Then N is not evolving at all. Since the error is not zero, there exists a transition t_{j_1} such that $I_c(\tau)(j_1) = 1$, and t_{j_1} cannot be fired, thus at least one input place p_{k_1} to

Places
$I_o = 400\sin(0.00011t)\ W/m^2$
$T_o = 298 + 7\sin(0.00011t)\ K$
$T_{ss} = 293.15 + 3\sin(0.00011t)\ K$
$C_{H_2O_{,o}} = 0.0060692 + 0.002\sin(2t)\ kg/m^3$
$v = \begin{cases} h(t) \text{ for } h(t) = 10\sin(0.001t) \geq 1 \\ 1 \qquad\qquad \text{otherwise} \end{cases}$
$\varphi_{cons} = 3 \times 10^{-10} + 2 \times 10^{-10}\sin(t)\ kg/^2s$

Table 2. Simulation parameters for **places** of the greenhouse modelled by $TCPN$.

t_{j_2} is unmarked. This procedure can be repeated until t_{j_1} is revisited. Since N is live, two cases are possible, places $p_{k_2}, \ldots, p_{k_1}$ require tokens, thus m_r is not reachable, or there exists an empty siphon (unmarked P-semiflow) and N is not live. A contradiction.

In the third case $I_c f(\tau) \neq 0$ is in the kernel of C, then a T-semiflow Υ composed by transitions $t_a, \ldots, t_n$ is being fired. It means that $\Psi_a(\tau) > 0, \ldots, \Psi_n(\tau) > 0$. However, from Lemma 10, we know that in a T-semiflow Υ there exist positive and negative $\Psi_\bullet(\tau)$. Thus not all $\Psi_\bullet(\tau)$ are positive and $I_c f(\tau)$ cannot form a T-semiflow at time τ. A contradiction.

In the fourth case $CI_c f(\tau)$ is in the kernel of e^T, but this is not possible because I_c was chosen to enforce (14) to be positive by selecting only the positive terms, thus even if f_k is not equal to σ_k, it is positive and (12) cannot be zero.

Thus $\dot{V}(e) = -e^T(\tau)CI_c f(\tau) < 0$ in all the cases, therefore (11) is a Lyapunov function. Then the error is asymptotically stable and the control (10) leads the marking from m_0 to m_r.

5. CASE OF STUDY

In this section the case of study is presented. We use greenhouse parameters values from (Van Straten et al., 2010) as in Table 2 and Table 3, but also, parameters can be estimated by a regression algorithm (Pérez-González et al., 2014). The greenhouse model is constructed with the following actuators: windows, fans, humidifier and pipe system (see Fig. 1). Using these actuators and parameters the $TCPN - PS$ greenhouse model is built according to Section 3. Notice that, the windows, fans and pipe system are represent by balance $TCPN$ modules. However, the humidifier is represented by a consumption $TCPN$ module.

This modeling methodology was implemented in a Graphical User Interface (GUI) programed in Matlab©. Each elementary Petri net module is represented by an element (icon) in the interface (for instance, actuators, perturbations) that can be added in a modular way while the $TCPN$ model is built. Therefore, the greenhouse-GUI provides facilities to operate the greenhouse system; any operator can easily manage the greenhouse, just by selecting the actuator or sensor icon in the interface and link it with another element.

The temperature T_g inside the greenhouse is depicted in Fig. 4; the system is simulated in a period of 8 hours. The figure shows that the temperature is increasing during the day and decreasing at sunset. However, the problem of

Transitions	Initial conditions
$t_1 = 0.5882$	$T_g = 288\ K$
$t_2 = t_5 = 1.3029 \times 10^3$	
$t_3 = t_6 = 7.9$	$T_s = 298\ K$
$t_4 = t_7 = 5.75$	$C_{H_2O} = 0.0026\ kg/m^3$
$t_{10} = 39470900$	
$t_{11} = 565000$	
$t_8 = t_9 = 5.75$	
$t_{12} = t_{13} = 2$	
$t_{16} = 17.4650$	
$t_{14} = t_{15} = t_{17} = t_{18} = t_{19} = 1$	
$t_{20} = t_{21} = t_{22} = t_{23} = 4.7652$	
$t_{24} = 41.7164$	
$t_{25} = 0.1120$	

Table 3. Simulation parameters for **transitions** of the greenhouse modelled by $TCPN$.

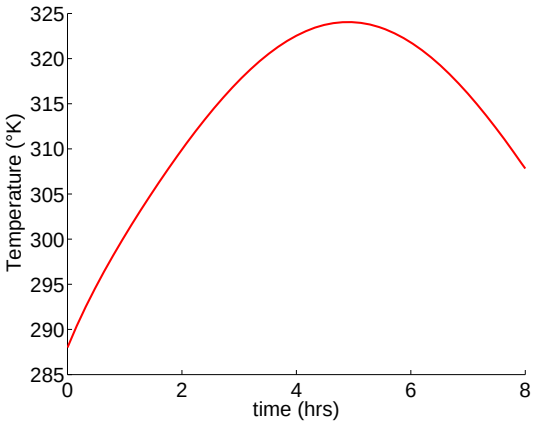


Figure 4. Temperature T_g indoor greenhouse without control.

obtaining an optimal environment for crops production in greenhouses depends on the ability to control the temperature inside; therefore, we will show that the previously proposed control law is able to control the temperature T_g.

Now, in order to apply the control law strategy, it is considered that the windows and fans are activated with the same control signal I_{c1}, in the case of the humidifier, the control signal was I_{c2}, similarly the control signal I_{c3} activate the pipe system. The system is simulated using the control strategy and establishing a set point for the temperature T_g (marking required $m_1 = 320^o K$) and it is considered that the other climate variables are not restricted.

Fig. 5 presents the temperature T_g behavior inside the greenhouse using the proposed control strategy. Notice that the simulation results show that with the control strategy the requirements are reached. Fig. 6 shows the evolution of the proposed control law for the fans and windows, humidifier and pipe system.

6. CONCLUSIONS

This work addressed the problem of modeling and reaching a required temperature environment in a greenhouse system. The modeling methodology presented provides a graphical representation of climate variables which allows an easy understanding of their interaction; moreover, the

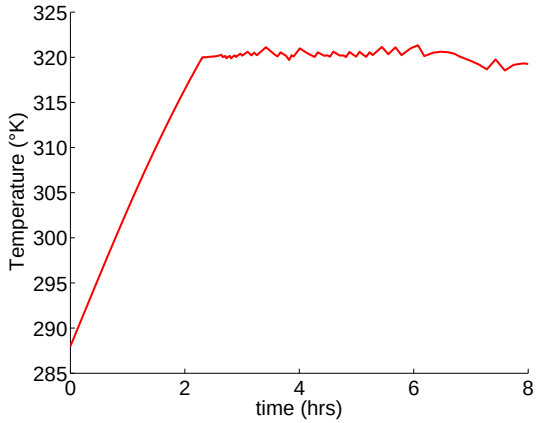


Figure 5. Temperature T_g controlled indoor greenhouse (marking required $T_g = m_1 = 320^oK$).

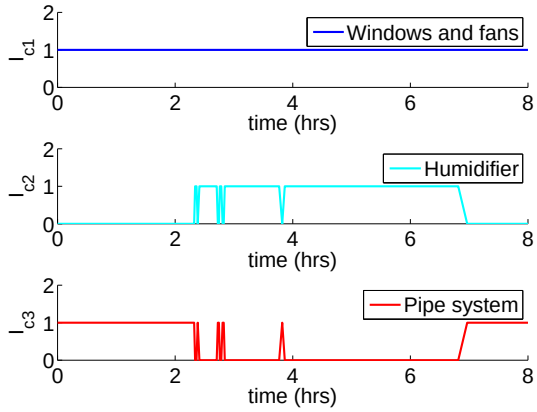


Figure 6. Control law for Pipe system, Humidifier, Windows and Fans.

proposed methodology is incremental and modular. Thus, $TCPN$ elements can be implemented in a GUI added or removed as necessary. The non-linear On-Off control law that exploits the $TCPN$ structure was presented. This control law allows to reach a required temperature when the target temperature is an interior point of the $int(T_r)$ and it is computed in polynomial time. The proposed control law uses local information of controllable transitions, using the marking error of its input and output places to determine if the firing decreases the marking error. Future work will address the problem of reaching a required temperature considering plants inside the greenhouse as a modular model in the global greenhouse $TCPN$ model.

ACKNOWLEDGMENTS

The research of J. Ruiz-León was performed during his sabbatical stay at the Institute of Information Theory and Automation, Czech Academy of Sciences, supported by CONACyT, Mexico, project 265475.

REFERENCES

Baldan, P., Cocco, N., De Nes, F., Segura, M.L., and Simeoni, M. (2011). Mpath2pn-translating metabolic pathways into petri nets. In *BioPPN2011 Int. Workshop on Biological Processes and Petri Nets, CEUR Workshop Proceedings*, volume 724, 102–116.

Boulard, T. and Baille, A. (1993). A simple greenhouse climate control model incorporating effects of ventilation and evaporative cooling. *Agricultural and Forest Meteorology*, 65(3), 145–157.

Cunha, J.B. et al. (2003). Greenhouse climate models: An overview. In *European Federation for Information Technology in Agriculture Conference, Debrecen, Hungary*.

Desirena-Lopez, G., Vazquez, L., Irevino, A.R., and Gomez-Gutierrez, D. (2014). Thermal modelling for temperature control in mpsoc's using timed continuous petri nets. In *Control Applications (CCA), 2014 IEEE Conference on*, 2135–2140. IEEE.

Dessel, J. and Esparza, j. (1995). *Free Choice Petri Nets*. Press Syndicate of the University of Cambridge.

Garcia-Malacara, J., Ramirez-Trevino, A., Vazquez, C.R., and Aguayo-Lara, E. (2013). Observer design for continuous timed petri nets with product server semantics. In *Emerging Technologies & Factory Automation (ETFA), 2013 IEEE 18th Conference on*, 1–8. IEEE.

Heiner, M., Gilbert, D., and Donaldson, R. (2008). Petri nets for systems and synthetic biology. In *Formal methods for computational systems biology*, 215–264. Springer.

Júlvez, J. and Boel, R.K. (2010). A continuous petri net approach for model predictive control of traffic systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 40(4), 686–697.

Mahulea, C., Seatzu, C., Cabasino, M., and Silva, M. (2012). Fault diagnosis of discrete-event systems using continuous petri nets. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 42(4), 970–984.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4), 541–580.

Pérez-González, A., Begovich, O., and Ruiz-Leon, J. (2014). Modeling of a greenhouse prototype using pso algorithm based on a labview tm application. In *Electrical Engineering, Computing Science and Automatic Control (CCE), 2014 11th International Conference on*, 1–6. IEEE.

Ross-León, R., Ramirez-Trevino, A., Ruiz-Leon, J., and Aguayo-Lara, E. (2014). Local control law for live and bounded continuous petri nets. In *Discrete Event Systems*, volume 12, 129–134.

Silva, M. and Recalde, L. (2002). Petri nets and integrality relaxations: A view of continuous petri net models. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 32(4), 314–327.

Silva, M. and Recalde, L. (2004). On fluidification of petri nets: from discrete to hybrid and continuous models. *Annual Reviews in Control*, 28(2), 253–266.

Tovany, J.L., Ross-León, R., Ruiz-León, J., Ramírez-Treviño, A., and Begovich, O. (2013). Greenhouse modeling using continuous timed petri nets. *Mathematical Problems in Engineering*, 2013.

Van Straten, G., van Willigenburg, G., van Henten, E., and van Ooteghem, R. (2010). *Optimal control of greenhouse cultivation*. CRC press.

Vázquez, C.R., Ramírez, A., Recalde, L., and Silva, M. (2008). On controllability of timed continuous petri nets. In *Hybrid Systems: Computation and Control*, 528–541. Springer.

Load Balancing System to Low Voltage Grid using Petri Nets ⋆

Sicchar J. R * **Silva J. R** ** **Costa C. T., Jr.** ***

* *Control Automation Depart.,Technology High College,Amazonas State University,1200 Darcy Vargas Av, CEP 69050-020, Manaus,AM,Brazil (e-mail: jvilchez@ uea.edu.br).*
** *University of São Paulo, 2231 Prof. Melo Morais Av, CEP 5508-9000, Butantã,SP,Brazil (e-mail: reinaldor@usp.br)*
*** *ITEC, Electrical Engineering Department,Federal University of Pará, Belém,PA,Brazil, (e-mail: cartav@sufpa.br)*

Abstract: Current advances of smart grids are causing a demand for new operations and services, specially to the low voltage consumers grid. Among these processes,load balancing has a prominent capability for ensuring stable states between feeders. This paper presents some results about the design of an automated process to load balancing feeders in final consumption units of a small urban smart grid using a timed sliced Hierarchical Petri net. The main objective is to analyze the proposed system and establish an efficient and reliable workflow to automate load balancing and ensure stability while minimizes intervention. As a result it improves the quality of power service to the low voltage final consumers.

Keywords: Smart Grids,Load Balancing,Timed Hierarchical Petri Nets,Final Consumption Units.

1. INTRODUCTION

The perspective of having urban smart grids became closer since new approaches were developed relying in small urban smart grid (SUSG), Shahgoshtasbi (2014) or even to mix with the legacy system centered in hydro-power in what concerns low voltage (LV) units.

In this context, load balance feeders is an important issue to the quality of energy providing service, and several direct algorithms were proposed, Shahnia et al. (2014), Sharma et al. (2014), Sicchar et al. (2015) which should now be arranged in an automated process. These "hybrid algorithms" are in fact focused on distributed energy consumption management system (EMS). Thereby forming functional flow for automated processes within SG vision, Huang et al. (2015). Its extends energy consumption management until the final consumer units in LV circuits. Where among others services, are offered the voltage stability evaluation and the load balance for for residential feeders,Sicchar et al. (2015).

In a special way, it is observed that some consumer units loads, cause imbalance between feeders in LV grid. Then, identifying a problem that affects state equilibrium feeders grid, and energy quality supplied caused by power and electrical current cause imbalances among feeders,Sharma et al. (2014).Currently, some possibles alternatives like identification and minimization of losses are being developed as Automatic Feeders Reconfiguration (AFR). It is applied in LV grid and identifies some power losses, load imbalance and switches final consumers feeders between grid feeders,Siti et al. (2011), Shahnia et al. (2014).

However, previously alternatives mentioned show a wide gap in formal modeling for load balancing system design. This does not present a formal workflow validation for the AFR process,Alt et al. (2006), Wang (2015).For this reason, we indicated as an hypothesis: Petri nets (PN) can be used into balancing process performance in LV grid.That is, by formal modeling system is possible to obtain some process that improves load balancing efficiency. At where algorithms form a system and service that supporting the final consumers.

This system also have a supervision an information systems for maintaining the LV system (but are not addressed in this work). Which managed the demand of energy consumption. Petri nets, represent in this work the structure and system architecture and workflow tasks and control in system.

This article, explains in second section background related in Petri nets definitions and the scientific review in automatic feeders reconfiguration; in third section presents a proposal to the DPMS system model; in fourth section shows proposed algorithm design in PN; in the fifth section shows design analysis and validation based on the operational flow performance followed by its respective discussion; finally, last section presents the conclusion and and points to future work lines.

2. BACKGROUND

In this section, we put background in specific review related load balancing algorithms development in LV grid.

⋆ Author José R. Sicchar, thanks to FAPEAM and Amazon State University. Author Da Costa C.T.Jr.,thanks to Federal University of Pará.

Then, we have also review about PN use in SG. It will addressed, some specific definitions of PN that will be important for the development of this proposal.

2.1 Review Stage

As load balancing method within AFR context, we first can mention the transfers overload concentrated method (losses and loads) into specific feeder, working from three-phase final consumption units. This method, is based on minimizing current consumption achieved by *Fuzzy* machine inference and *Newton-Rhampsom's* optimization algorithm, between power consumption and power variation in each feeder,Siti et al. (2011).

Another method, focused only on single-phase final consumption units, minimizing power and voltage consumption use an hybrid genetic algorithm. In this case, also taking up load transfer but reconnecting single-phase consumers in same feeder in grid, with lower load level,Shahnia et al. (2014).

We can also mention, the hybrid load consumption algorithm model for final consumption units in LV grids,based on the Unified Modeling Language (UML)-PN paradigm, Sicchar et al. (2011). Which connects data acquisition, classification, programming and consumption forecast, and sending best selection for arrangement switching feeders in load balancing.

In currently article, We will continue load balancing in final consumption units feeders model but, based on hierarchical PN paradigm, using balancing diagnostic,current and load consumption prediction, minimization consumption and optimal arrangement sequence selection flow algorithms as hierarchical sub-processes in main PN.

Thereby, contributing with efficient process in load balancing, which can be used as an alternative method and/or interface in existent LV grid, or as support process in supervision center of a small urban smart grid (SUSG).

2.2 Definitions

- *Definition 2.2.1.Petri Net.*A Petri net structure is a directed weighted bipartite graph,according " (1)",Silva (2012):

  $$N = (P, T, A, w). \quad (1)$$

  where: "P" is the finite set of places,$P \neq \emptyset$. "T" is the finite set of transitions, T$\neq \emptyset$."A"$\subseteq$(PxT)$\cup$(TxP) is the set of arcs from places to transitions and from transitions to places."w":A $\rightarrow$ {1,2,3,$\cdots$} is the weight function on the arcs. where: "P" is the finite set of places,$P \neq \emptyset$. "T" is the finite set of transitions, T$\neq \emptyset$."A"$\subseteq$(PxT)$\cup$(TxP) is the set of arcs from places to transitions and from transitions to places."w":A $\rightarrow$ {1,2,3,$\cdots$} is the weight function on the arcs.
- *Definition 2.2.2. Timed Petri Net.*Defined by "(2)",Silva (2012):

  $$N = (P, T, A, w, M0, f). \quad (2)$$

  where: (P, T, A,w,$M0$) is a marked Petri net, Silva (2012),"f": T$\rightarrow$R$^+$ is a firing time function that assigns a positive real number to each transition on the net.
- *Definition 2.2.3. Hierarchical Petri net by Place Bounded Substitution.*In a PN structure give by "(3)",Silva (2012):

  $$N = (P, T, F). \quad (3)$$

  There is, an Y sub-net which limited by place so the replacement of this Y sub-net, generates another net N' = (P', T', F'), where: i) P'= $P \setminus T \cup \{S_y\}$, S_y is the new element that replaces Y; ii) T'=$T \setminus Y$;iii)F'=$F \setminus$Int(Y),Int(Y)is the inner Y arcs set, Silva (2012).

3. PROPOSAL

According, of hypothesis and the opportunity to improvement the load balancing process in the current structure of the LV distribution grid. We propose in this article: modeling an intelligent process for load feeder balancing in final consumption units using a Timed Hierarchical Petri Net (THPN), in order to obtain reliably and efficiently workflow, formally validated.

In fact, this process is based on algorithms of artificial intelligence as fuzzy logic and optimization systems; and also forecast algorithms based on stochastic process as Markov chains,Sicchar et al. (2015).

The contribution of the validated model, could be useful in reshaping modernization of the existing LV distribution grid structure (legacy system). It can be used as the implementation of a specific service in the process of automatic feeders reconfiguration in final consumption consumer units, within the scope of small urban smart grid.

Through hierarchical PN will be performed validation system design including its inner sub-processes,that make feeders reconfiguration system, called "**DPMS**". Which is explained in following sub-section.Similarly, use of Timed transitions in proposal model are intended to represent the most realistic way possible an entire period by simulation processing system, for consumer units feeders balancing.

In this particular case, takes a granular period of 60 minutes, i.e., seeking feeder reconfiguration lasting one hour, depending on sample consumption obtained at 10 minute intervals derived from real-time measurement, in the case of a SUSG. Thus, it is not covered in this case, the load balancing flow in existing secondary grid, which is the subject of future work.

In this article, will be developed according to initial proposal of the authors,Sicchar et al. (2011) however having a contribution a broad and integrated PN with hierarchical description of its sub-process. It will be based on the system developed to consumer units energy consumption diagnose,Sicchar et al. (2015) but now considering beyond imbalance diagnostic and consumption forecasting stage, more two stages: minimizing current consumption and switching sequence selection.

3.1 DPMS Architecture

The process system, is called "**DPMS**" because of its four stages or sub-processes:"Diagnose", "Prevision", "Minimization" and "Selection", each with a specific algorithm. So, these formed the DPMS system.

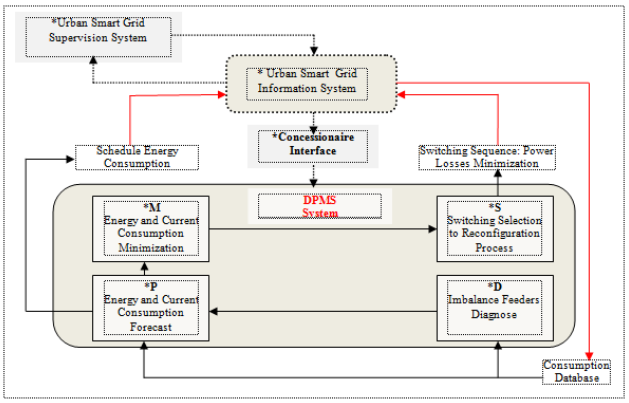


Fig. 1. DPMS Architecture

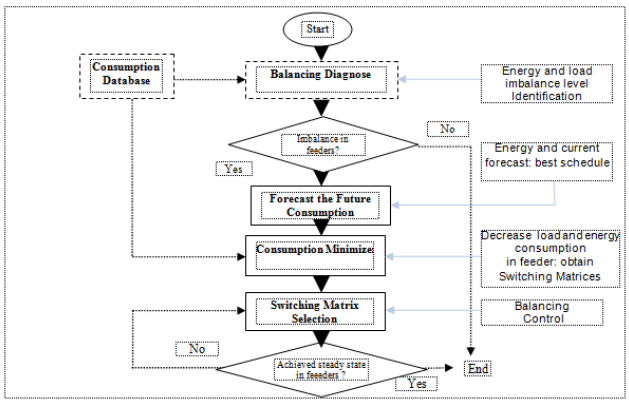


Fig. 2. DPMS Flowchart

Thus, the DPMS system has an architecture formed as mentioned above, four specific processing (whose specific algorithms are not covered in this article):

- Balancing Diagnose or only "**D**" stage, that identifies the imbalance level in each feeder, Watching two situations: "balanced feeder", thus having the algorithm operating finish; and "unbalanced feeder" that activates the remaining stages of system, in sequence, starting with the consumption forecast step.
- Prevision Consumption or simply "**P**" stage, which only is activate when an imbalance is identified (in some feeder). It forecasts the current and energy consumption in feeders and returning this processing to the SG information system, that later develops the energy future consumption matrix indicating the trend of consumption to the final consumers.
- Minimization Consumption or just "**M**" stage, which procedure some combination of switching between feeders from the current and the future consumption of the energy and electrical current obtained by "P" stage, in order to minimize power losses effects and ensuring the equilibrium state in feeders.
- Switchin Sequence or only "**S**" stage. Which chooses the best switching combination from the "M" stage.
  This, selection is based on a correlation ratio analysis between the real value of consumption and with their values from the minimization stage.The final processing is sent to the information system SG as switching sequence, to procedure in fact the AFR.

In Fig. 1 we can see the DPMS architecture system model. It can be seen as a support process for system information to supervision center in SUSG environment.It can also be inserted, as an interface in the existing secondary grid system.

Then, we have the operation flow of the DPMS system. Which is shown in more details in Fig. 2. This flow, is started from consumption data processing, and after consumption diagnosis are identified possible losses and load imbalances.

In positive case, starts energy and current consumption prediction process, in each final consumer feeder. The main objective is to obtain,the future consumption matrix of electrical current.Furthermore, the prediction results serve to minimizing consumption process.

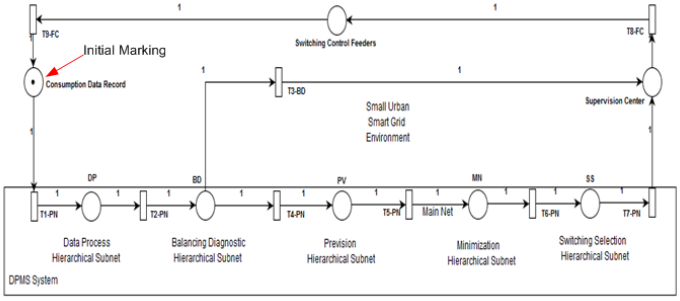


Fig. 3. DPMS THPN:Main Petri net

Followed by minimize consumption, looking for same sequences combinations of switching between feeders. Which are calculated from current and future values of electrical current consumption.

Finally, switching matrix selection chooses which through, send best combination for final consumer feeders balancing implementation. If imbalance minimizing process ends, otherwise proceeds in choosing other combinations for switching.

4. DPMS SYSTEM IN PETRI NETS

In Fig. 3 it is shown the "DPMS" system modeling in THPN. It describes the main PN of DPMS system. The hierarchical extension used is place bounded substitution (PBS) according to definition 2.2.3, being indicated with red arrow, the initial state "S0".

The inner workflow of DPMS system is formed as follows (see Fig.5):

- "**DP**".Data Process Hierarchical subnet.Which has all statistical treatmen sub-processes.It is formed by:"L_1-DP",that classifies power, current and energy consumption data."L_2-DP",which calculates average consumption."L_3-DP", that forms discrete consumption states.
- "**BD**".Balancing Diagnose Hierarchical subnet. Which has all consumption diagnose inference sub-processes. Which is represented by "L_4-BD", a PBS place that representing the inference machine design: load and energy consumption as input variables in inference system; load and energy consumption variation as

input variable; current consumption as output variable; the inference rules. And the output variable that obtain conditions to imbalance diagnose. In this work will not be covered in detail this sub-process. **BD** exit has two conditions: (final consumption unit) FCU balanced "T_9-BD"and, No balanced FCU "T_{10}-PN" that actives **PV** place.

- "**PV**".Prevision Hierarchical subnet, with electrical forecast current consumption sub-processes. It is formed by:"L_5-PV",which inserts discrete states consumption. "L_6-PV"that calculates incidences consumption. "L_7-PV", which obtain transition matrix. "L_8-PV",that obtain forecast electrical current consumption.
- "**MN**". Minimization Hierarchical subnet.It is formed by:"L_9-MN",which inserts measured electric current vector."L_{10}-MN",that inserts forecast current vector."L_{11}-MN", which forms minimization consumption vector."L_{12}-MN", that inserts minimization consumption formula."L_{13}-MN" which obtain arrangement switching matrices.
- "**SS**".Switching Selection Hierarchical subnet. Which has all switching selection inference sub-processes. Which is represented by "L_{18}-SS", a PBS place that representing the inference machine design. In this work is represented in detail the design of SS subprocess in section 5.3. After, this flow goes to supervision center (CS), and then feeder switching control (FC). The process goes through a measurement by data measurement (DM), whose flow is transferred as consumption data record (CDR).

5. RESULTS-VALIDATION ANALYSIS

In this section,we will show the validation results of DPMS system, modeled in THPN,using reachability graph, invariant analysis applied in Main DPSN THPN and also in Hierarchical DPMS THPN which shown all processes of system and also use siphon and traps analysis for validation workflow. For experimental results, was used as tool a free version of Pipe 4.3.0.

For simulation, timed transitions were used. It is distributed fixed time intervals, for each operation of subprocess,it was used T = 10 seconds,and for total integration operations add up to a full period of T = 60 seconds.

5.1 Main DPMS THPN validation

a) Reachability Graph. The Main DPMS Reachability graph is shown in Fig. 4. So, it represents the PN reachable diagram obtained from Its initial state "S0" highlighted in black, that also represents initial marking of PN.

Through Main DPMS simulation, if verifies that does not exist deadlock.However, checking a possible conflict highlighted with a black arrow on "S2" **BD** place output, between transitions "T_3-BD" and "T_4-PN", due to balancing result that evaluates two conditions: balanced FCU or unbalanced FCU. However, this "conflict", will not be controlled due to consider a random order in system simulation when it executes balancing feeders analysis.

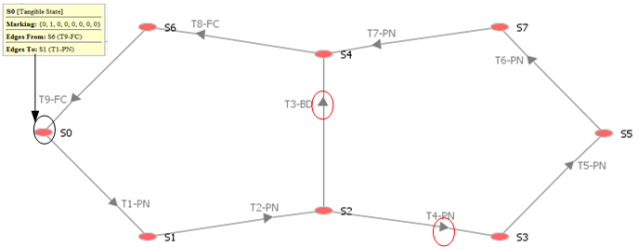


Fig. 4. Main DPMS Reachability Graph

b) Invariants Analysis. Still looking for reachability graph, we note that all processes of system are sequential.However, BD process is the most critical because to determine the end of process, if it is found that FCU feeders are balanced, or the continuation of process otherwise, activating PV process.This condition form, a specific place invariant: "PV" sub-process can not happen before "BD" sub-process.

Through, with the invariants place (P-Invariants) it can be seen that the sequence of system sub-processes will only run one at a time, up until to develop the whole workflow system.

Thus, we have the follow condition invariant: "BD", "PV","MN","SS" places cannot happen before "DP"sub-process. Thus, we have the follow condition:DP marking,BD marking,PV marking, MN marking and SS marking, should be less or equal to 1,"(4)":

$$M(DP) + M(BD) + M(PV) + M(MN) + M(SS) = 1 \quad (4)$$

We have also the follow condition: "Supervision Center"(SC),"Feeder Switching Control"(FC), "Consumption Data Record" (CDR) places cannot happen before "BD"sub-process. If any as a result of the process, FCU balanced:

$$M(BD) + M(SC) + M(FC) + M(CDR) \leq 1 \quad (5)$$

By P-Invariants result, we can verified that invariants indicates aboven, are true:

$$M(BD) + M(CDR) + M(DP) + M(FC) + M(MN) + M(PV) + M(SS) + M(SC) = 1 \quad (6)$$

The transition invariants (T-invariants) can be used to verify the order of each subprocess system. Showing the sequence of the flow system is well structured. Through, the PN Invariants is shown in $2x9$ constant"T-Invariants" matrix depending on "T_{1-PN}","T_{2-PN}","T_{3-BD}","T_{4-PN}","T_{5-PN}", "T_{6-PN}","T_{7-PN}","T_{8-FC}","T_{9-FC}" transitions:

$$T_{Inv} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & 1 \\ 1 & 1 & 0 & \cdots & 1 & 1 \end{bmatrix} \begin{bmatrix} T_{1-PN} \\ T_{2-PN} \\ T_{3-BD} \\ T_{4-PN} \\ T_{5-PN} \\ \vdots \\ T_{9-FC} \end{bmatrix} \quad (7)$$

We can see, also that the net is covered by positive T-Invariants, therefore it might be textbfbounded and **live**.

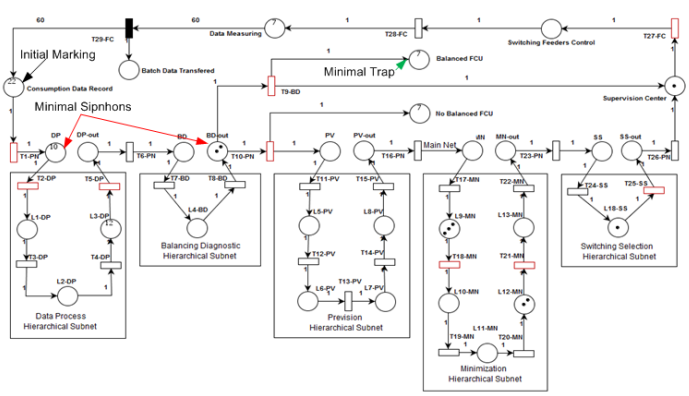


Fig. 5. DPMS THPN Simulation

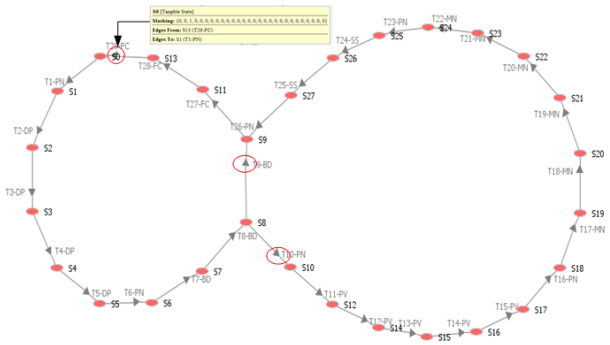


Fig. 6. DPMS THPN Reachability Graph with initial marking and conflicts

5.2 Hierarchical DPMS THPN validation

In Figure 5 it is shown the Hierarchical DPMS THPN simulation.Were performed 1500 "firings" with a period of 60 seconds between each, in order to have a complete operating in 1 minute. So, Were considered 60 FCU in SUSG, to be targets of balancing feeders analysis process. Thus, in "**CDR**" place there are electrical current and energy consumption of this 60 FCU.During DPMS system simulation is verifying efficiently workflow, and there is not deadlock or conflicts (marked in red in Figure 6) like main PN, that needs regulatory control.

However, it is found some "**siphons focus**" especially on "**DP**" and "**BDout**"(in red arrows), where are accumulated and consumed several tokens continuously, but it normalizes and adjusts over time.

It is, also found some "traps focus" in "**Balanced FCU**"(in green arrow) where which are accumulated several tokens, but this is actually equal to "**No balanced FCU**": both are only process counters, to facilitate processing system count.

The DPMS THPN Reachability graph is shown in Figure 6. It represents the diagram of reachable states of the PN. Its initial state S_0 highlighted in red. If verifies that does not exist deadlock, and that the PN it might be textbfbounded and **live**.

Looking for reachability graph, we note that all some specifics place invariant: "PV" sub-process can not happen before "BD" sub-process.But also, we have the follow

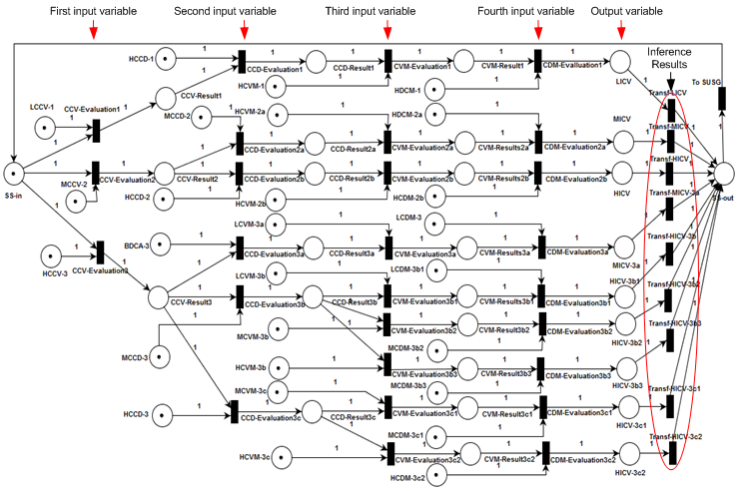


Fig. 7. Switching Selection sub-net

condition invariant: "BD", "PV","MN","SS", "SC","FC", "Data Measuring (DM)", and "CDR", places cannot happen before "DP"sub-process. Thus, we have the follow condition shown in "8":

$$\begin{aligned} &M(DP) + M(BD) + M(PV) + \\ &M(MN) + M(SS) + M(SC) + \\ &M(FC) + M(DM) + M(CDR) = 1 \end{aligned} \tag{8}$$

By P-Invariants result, we can verified that invariants indicates aboven, are true:

$$\begin{aligned} &M(BD) + M(BD - out) + M(CDR) + M(DM) + \\ &M(DP) + M(DP - out) + M(L10 - PV) + \\ &M(L11 - MN) + M(L12 - MN) + M(L13 - MN) + \\ &M(L14 - MN) + M(L15 - MN) + M(L16 - SS) + \\ &M(L1 - DP) + M(L2 - DP) + M(L3 - DP) + \\ &M(L6 - BD) + M(L7 - PV) + M(L8 - PV) \\ &M(L9 - PV) + M(MN) + M(MN - out) + M(PV) + \\ &M(PV - out) + M(SS) + M(SS - out) + \\ &M(SC) + M(FC) = 1 \end{aligned} \tag{9}$$

5.3 Switching Selection sub-net validation

For the Switching Selection sub-process was modeled an hierarchical sub-net, as illustrated in Figure 7. It is represents the model of inference selection of minimized consumption values of electric current performed by MN subprocess.

This is formed by four input variables: Current Consumption Value (CCV), Current Consumption Variation (CCD), Current Value Minimized (CVM), Current Variation Minimized (CDM)and a output variable, Current Correlation Value (ICV). All variables with three performance levels: low (L), medium (M) and high (H). With results highlighted in red, are selected the optimal switching matrix. Which can by means of this sub-net is implemented "Mamdani Fuzzy inference" model for selection of output combinations having the highest current correlation value.

In the case of the first inference rule is composed of the following stream: the first input variable "CCV" is evaluated (CCV-Evaluation1 transition), presenting as LCCV-1 result. The second input variable "CCD" is evaluated (CCD-Evaluation1 transition) presenting as HCVM-1 result. The third input variable "CVM" is evaluated

(CVM-Evaluation1 transition) and presented as HDCM-1 result. The fourth variable enters "CDM" is evaluated (CDM-Evaluation1 transition) presenting the final result of the output variable "ICV" LICV. Then transferred (transferrin-LCCV) to the output of the subnet to forward this information to the monitoring center of SUSG.

This sub-process represents "nine possible solutions" to the current variation correlation (inference rules) showed in Table 1. Which can assume low, medium and high correlation of current variation.

Table 1. Inference rules of Switching Selection sub-net

If	and	and	and	then
LCCV	HCCD	HCVM	HCDM	LICV
MCCV	MCCD	HCVM	HCDM	MICV
MCCV	HCCD	HCVM	HCDM	HICV
HCCV	LCCD	LCVM	LCDM	MICV
HCCV	MCCD	MCVM	MCDM	HICV
HCCV	MCCD	HCVM	MCDM	HICV
HCCV	MCCD	LCVM	LCDM	HICV
HCCV	HCCD	MCVM	MCDM	HICV
HCCV	HCCD	HCVM	HCDM	HICV

In this case by means of transition invariants it is possible to confirm the formation of each inference rules. Checking that they are well formed with all conditional and actions.

Rule 1 Training:

$$\begin{array}{l} CCD - Ev2b + CCV - Ev2 + CDM - Ev2b \\ CVM - Ev3a + TransfHICV + ToSUSG \end{array} \quad (10)$$

Rule 2 Training:

$$\begin{array}{l} CCD - Ev3b + CCV - Ev3 + CDM - Ev3b1 \\ CVM - Ev3b1 + TransfHICV3b + ToSUSG \end{array} \quad (11)$$

Rule 3 Training:

$$\begin{array}{l} CCD - Ev3b + CCV - Ev3 + CDM - Ev3b2 \\ CVM - Ev3b2 + TransfHICV3b2 + ToSUSG \end{array} \quad (12)$$

For the other rules can also be applied the same procedure.

6. CONCLUSION

A model Petri net, of load balancing process, called DPMS system to final consumption units in low voltage secondary grid, has been presented from which it is shown workflow formal validation. Having been used a timed hierarchical Petri net, to represent a dynamic abstraction of the operation flow of main and internal processes that form the DPMS system.Through the net synthesis,the reachability graph,invariant analysis and, workflow simulation, among others the vividness and limited network properties were checked.Verifying in summary efficiently operation of system without deadlocks and conflicts, that requested implementation of any regulatory control.

It was also presented in detail the inference procedure for switching selection sub-process. Noting that from the transition invariants is possible to verify the formation of the inference system rules.

Suggested future steps develop model DPMS system using timed hierarchical colored Petri nets,to further improve computational efficiency.

ACKNOWLEDGEMENTS

The authors thank UEPA, UFPA, USP and FAPEAM, for allowing scientific achievement of this proposal.

REFERENCES

Alt, M., Gorlatch, S., Hoheisel, A., and Pohl, H.W. (2006). Using high level petri nets for hierarchical grid workflows. In A. Round (ed.), *Advances in Enzymology*, volume 2, 1–87. Academic Press, New York, 3rd edition.

Huang, Y., Mao, S., and Nelms, R. (2015). Smooth scheduling for electricity distribution in the smart grid. volume 9, 966–977. IEEE System Journal, New York, 3rd edition.

Shahgoshtasbi, D. (2014). A new intelligent neuro–fuzzy paradigm for energy-efficient homes. In A. Round (ed.), *Advances in Enzymology*, volume 8, 664–673. Systems Journal, IEEE Transactions, New York, 3rd edition.

Shahnia, F., Wolfs, P., and Ghosh, A. (2014). Voltage unbalance improvement in low voltage residential with rooftop pvs using custom power devices. In E. Power and E. Systems (eds.), *Advances in Enzymology*, volume 55, 362–377. Elsevier, New York, 3rd edition.

Sharma, I., Cañizares, C., and Bhattacharya, K. (2014). Smart charging of pevs penetrating into residential distribution systems. In A. Round (ed.), *Advances in Enzymology*, volume 5, 1196–1209. IEEE Transactions on Smart Grid, New York, 3rd edition.

Sicchar, J., Jr., C.D.C., Salmon, A., Silva, J., Pina, I., and Gomes, R. (2011). Sistema inteligente para análise de consumo de energia elétrica em smart grid de baixa tensão. In P. in X SBAI 2011 (ed.), *Advances in Enzymology*, volume 1, 1–6. SBAI 2011, New York, 3rd edition.

Sicchar, J., Jr., C.D.C., Silva, J., and Freitas, R.D. (2015). Gerenciamento de consumo de energia em residências com frame gcr. In P. in XII SBAI 2015 (ed.), *Advances in Enzymology*, volume 1, 1–6. SBAI 2015, New York, 3rd edition.

Silva, J. (2012). Timed petri nets. chapter 16. petri nets manufacturing and computing science. In INTECH (ed.), *Advances in Enzymology*, volume 1, 359–378. Intech, New York, 3rd edition.

Siti, W., Jimoh, A., and Nicolae, D. (2011). Distribution network phase load balancing as a combinatorial optimization problem using fuzzy logic and newton-raphson. In E. Electric Power Systems (ed.), *Advances in Enzymology*, volume 22, 1079–1087. Elsevier, New York, 3rd edition.

Wang, L. (2015). Knowledge representation and general petri net models for power grid fault diagnosis. In I.J.. Magazines (ed.), *Advances in Enzymology*, volume 9, 866–1087. IET, New York, 3rd edition.

Load Frequency Control of a Multi-area Power System Incorporating Variable-speed Wind Turbines ⋆

Semaria Ruiz Alvarez * **Julián Patiño** ** **Jairo Espinosa** ***

* *Universidad Nacional de Colombia, Medellín, Colombia (e-mail: seruizal@unal.edu.co).*
** *Universidad Nacional de Colombia, Manizales, Colombia (e-mail: juapatinomu@unal.edu.co)*
*** *Facultad de Minas, Universidad Nacional de Colombia, Medellín, Colombia (e-mail: jjespino@unal.edu.co)*

Abstract: The increasing use of renewable technologies in power generation may require its participation on ancillary services like frequency regulation. For the specific case of wind sources, this may lead to participation in frequency control loops. This paper focuses on the simulation of the performance of the LFC scheme for a multi-area power system, with participation of DFIG turbines in the frequency control loops, through the *synthetic inertia method*.

Keywords: Power systems, load frequency control, wind turbine, pitch angle, multi-area system.

1. INTRODUCTION

The behavior of generation systems based on unconventional energy sources like wind and solar energies may impact several aspects related with the operation and control of power systems; one of the ongoing research topics is related with understanding the impact of these new sources on the system frequency ((Bevrani, 2009; Valencia et al., 2012; Rahmann and Castillo, 2014; Horta et al., 2015)). As wind constitutes the most extensively used renewable energy source in the world, there are many studies about control strategies for the inclusion of wind turbines in the load frequency control loops of power systems (a complete rewiew of grid requirements and control methodologies can be seen in (Daz-Gonzlez et al., 2014)). In (de Almeida et al., 2006), an optimization is proposed in order to schedule the active and reactive power that wind turbines must deliver to meet the grid requirements. Additionally, a controller for the pitch angle in the turbine was also presented, forcing the turbine to operate over a de-load curve.

In (Ramtharan et al., 2007b) and (Moore and Ekanayake, 2009), the synthetic inertia method is used, where an additional control loop is proposed to emulate the behavior of conventional units in the frequency response of wind-turbines. Also, in (Camblong et al., 2014), the dynamic model of a DFIG (Doubly-fed Induction Generator) wind turbine is proposed in order to design an LQR controller to provide frequency support using reference torque and reference pitch angle as inputs. In (Bernard et al., 2013), a MPC (Model Predictive Controller) is developed through a simplified model of the DFIG, having the quadrature-axis rotor voltage of the wind-turbine dynamic system as an input.

In spite of the considerable efforts of the previously mentioned studies, those works did not consider the application over multi-area power systems, which are increasingly common representations as power grids continue to grow in size and complexity. Additionally, it is necessary to perform a comparison between the conventional control methods used for the load frequency control as PI (Proportional-Integral) and other more complex control strategies such as as the LQR controllers or MPCs in multi-area power systems.

Based on these requirements, this paper presents the simulation of the performance of the LFC (Load Frequency Control) scheme for a multi-area power system, having into account the participation of DFIG turbines in the frequency control loops, through the *synthetic inertia method*, with PI controllers for the AGC (Automatic Generation Control), the quadrature rotor voltage and the pitch angle and employing the simplified wind turbine model proposed in (Moore and Ekanayake, 2009) and (Bernard et al., 2013). The performance of these models and its contribution are illustrated by simulation using the IEEE nine bus system benchmark.

The paper is organized as follows: in section 2, a short description about the load frequency control is presented, and next the models and control loops required for wind modeling contribution are described in section 3. Section 4.1 shows the selected benchmark. The results of including wind-turbines in the LFC control loop are presented in section 4.2. At the end, some concluding remarks can be found.

⋆ This work was supported by Colciencias through the programs "Jóvenes investigadores - Convocatoria N.645 of 2014" and "Convocatoria 528 - Convocatoria Nacional para Estudios de Doctorados en Colombia 2011".

2. LOAD FREQUENCY CONTROL

In an electric power system, the frequency of the voltage wave must rely between rigorous limit values in order to keep the quality of the electric service inside an acceptable operational margin. This is a difficult task since the frequency is related to the load-generation balance and both variables are changing during the daily operation. This difficulty increases even more if the penetration of renewable energy sources is considered. Therefore, generating units require a control system able to respond to the generation or load changes, which is known as the load frequency control (LFC), (Saadat, 1999). This control system is organized in three levels: primary, secondary an tertiary frequency control. These levels are explained in (Bevrani, 2009), as follows:

- **Primary control (LFC).** Primary control is the fastest. It operates in a time band between 2-20 seconds and acts at the local level over each generation unit.
- **Secondary control (AGC).** Secondary control operates in a time band between 2 secondsand 2 minutes. It allows the correction of the steady-state errors in system frequency. For multi-area systems, it also regulates the power exchanged between areas.
- **Tertiary control.** Tertiary control is used in large power systems. It operates in a time margin above 10 minutes and specifies the set-points to the generation units by optimizing some cost function. Because the time and the tasks involved, this control stage is often considered as part of dispatch operations, and not fully mentioned as a frequency control loop.

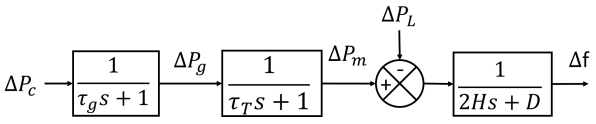


Fig. 1. Model of the set: speed-governor, turbine, generator and load (based on (Saadat, 1999)).

With the aim of designing these control systems, power system elements are modeled. For this, an equivalent model of an isolated power system is presented, where the response of all loads and generators are represented by a single damping and an equivalent inertia (see (Saadat, 1999) for a deeper description of these models). Also the dynamics of the speed-governor and the turbine are represented by first order transfer functions as illustrated in Figure 1, where:

- ΔP_m is the change in mechanical power of the generator.
- ΔP_g is the change in the governor output.
- ΔP_l is the load perturbation.
- Δf is the frequency change.
- D is the damping coefficient, due to the frequency sensitive loads.
- H is the equivalent inertia, the sum of the inertia of all generators in the system.
- ΔP_c is the control action of the LFC.

In an isolated power system, the function of the LFC is to restore frequency to its nominal value after a perturbation,

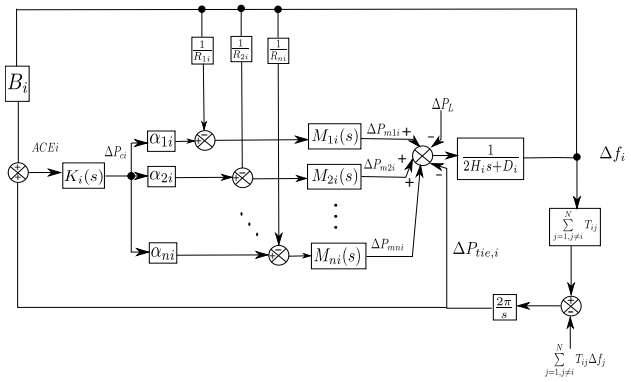


Fig. 2. Load frequency control scheme for a multi-area power system (based on Bevrani (2009)).

without taking into account power exchange. On the other hand, for interconnected systems, the control area concept has to be used, which is a group of generators and loads where the generators respond to load changes uniformly.

2.1 Load frequency control in multi-area power systems

A multi-area power system is composed by single area power systems that are interconnected by transmission lines or tie-lines. There is a LFC system on each of these single-area systems, which have the function of regulating, not only the frequency in the local area, but the power exchange with other areas (Saadat, 1999). Therefore, in the LFC dynamics, the tie-line power signal must be added in order to guarantee that the frequency deviation produced by load fluctuations in one area is controlled locally and does not propagate to other areas. Thus, each local area must be able to control its own load-generation perturbations.

Figure 2 depicts the load frequency control scheme for an N-area system, where (based on (Bevrani, 2009)):

- $M_{ni}(s)$ is the transfer function corresponding to the set of speed-governor, turbine, generator and load in area i.
- T_{ij} is the initial power exchange factor between area i and area j.
- ΔP_{tiei} is the change in the power exchange between the area i with other areas.
- Δf_j is the change in the frequency of the areas connected to area i.
- B_i is known as the *bias factor*, which allows minimize the power exchange with other areas through the input error signal in the AGC controller.
- $K_i(s)$ is the transfer function of the AGC controller.
- α_i is the participation factor of each generator in the AGC (these values are assumed equal to one for each generator if not established otherwise).

3. INCLUDING VARIABLE SPEED WIND-TURBINES IN LOAD FREQUENCY CONTROL

In order to include variable speed wind-turbines in the LFC, it is required to represent the power production of these generation units. For this, the model proposed in (Ramtharan et al., 2007a) is taken into account. There, the authors propose that the wind-turbine performs at an

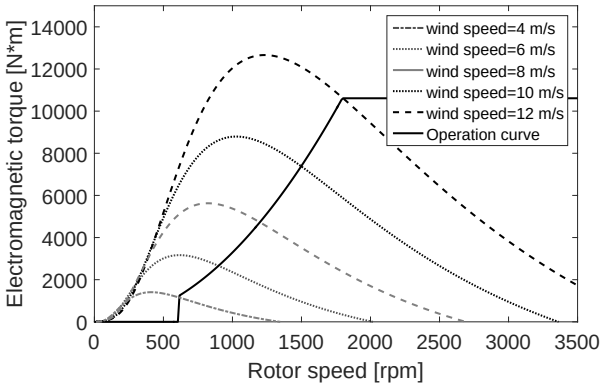


Fig. 3. Wind-turbine operation curve for different wind speeds (Thomsen, 2006)

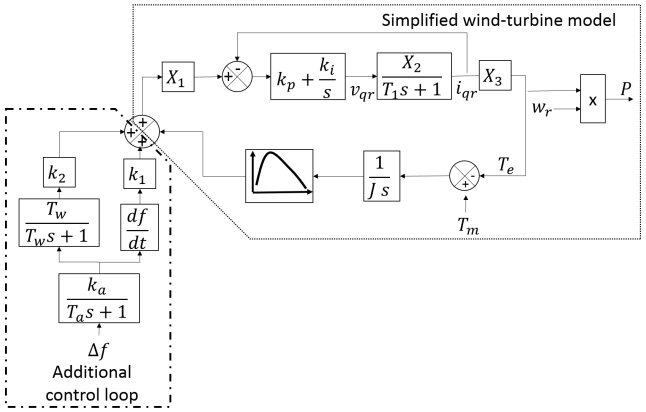


Fig. 4. Wind-turbine simplified model with additional control loop for LFC contribution (based on (Ramtharan et al., 2007a)).

operating point below its maximum power. As seen from figure 3, this results in the operating point of the wind-turbine being moved to the right regarding to its point of maximum power extraction.

The mechanical torque in each of the curves shown in Figure 3 is given by the equation (Thomsen, 2006):

$$T_m = \frac{Pm}{w_{shaft} G_b p}. \tag{1}$$

In equation (1), ω_{shaft} is the rotational speed of the wind-turbine shaft, G_b is the gearbox ratio, p denotes the number of pole pairs in the generator and P_m is the mechanical power, which is defined as:

$$P_m = \frac{1}{2}\rho\pi R^2 v^3 C_p. \tag{2}$$

There, $\rho = 1.225\ kg/m^3$ is the air density, $R = 45\ m$ is the blade's length, v is the wind speed and C_p is the efficiency coefficient described below:

$$C_p = 0.22((\frac{116}{\lambda_t}) - 0.4\beta - 5)e^{\frac{-12.5}{\lambda_t}}, \tag{3}$$

with λ_t a parameter given by Thomsen (2006):

$$\lambda_t = \frac{1}{\frac{1}{\frac{R\omega_{shaft}}{v}+0.08\beta} - \frac{0.035}{1+\beta^3}} \tag{4}$$

In this way, the value of the coefficient C_p will be depending on the pitch angle β, the wind speed v and the rotational speed ω_{shaft}. Thus, for each wind speed value an operating point slightly moved to the right is taken. At this operating point, the torque is given by equation (5), where the value of $K_{op} = 0.3$ is a constant.

$$T_{op} = K_{op} v^2 \tag{5}$$

Besides the simplified model previously described, is is also necessary to use a fraction of the power generated by the wind turbine in order to contribute to the LFC. Using the so-called *synthetic inertia model* (see (Ramtharan et al., 2007a) and Moore and Ekanayake (2009), a couple of additional loops are aggregated to the LFC: one containing a transfer function offering an output proportional (multiplying by K_1) to the frequency change rate (corresponding to the primary response of the wind turbine); and the other loop having the task of restoring the power delivered by the wind turbine after its participation in the load frequency control. The deviation frequency signal is pre-filtered, previously, by a filter with gain K_a.

Both the simplified model and the synthetic inertia model are depicted in figure 4, where:

and

- ω_r is the rotor angular speed,
- n is the number of wind-turbines,
- iq_r is the quadrature rotor current,
- vq_r is the quadrature rotor voltage, and
- X_1, X_2, X_3 and T_1 are constant values representing relationships between the internal wind generator parameters (see (Ramtharan et al., 2007a) for a detailed description of them).

A $2MW$ wind-turbine, is selected to perform the simulations, the parameters of this turbine are presented in table A.2 in section 5.

Moreover, for wind speeds equal or over the rated wind speed of the wind-turbine, a pitch angle control should be performed to maintain the angular speed of the wind-turbine at its nominal value. Additionally, when a frequency event occurs, and the turbine is operating under the action of pitch control, it is proposed to add a loop of additional control where the pitch angle is increased by a value proportional to the frequency deviation (this value is denoted by the constant R_β). This pitch control scheme is illustrated in Figure 5 with a PI controller.

The whole wind turbine model with the inclusion of control loops described above is presented in figure 6, where the variable n indicates the number of generating units, and the variable v_w indicates the wind speed. This model has an additional element compared with the models of (Ramtharan et al., 2007a) and (Moore and Ekanayake, 2009), the variable P_{ref}, which constitutes the power that the wind-turbine would deliver if it would not be contributing to system frequency control.

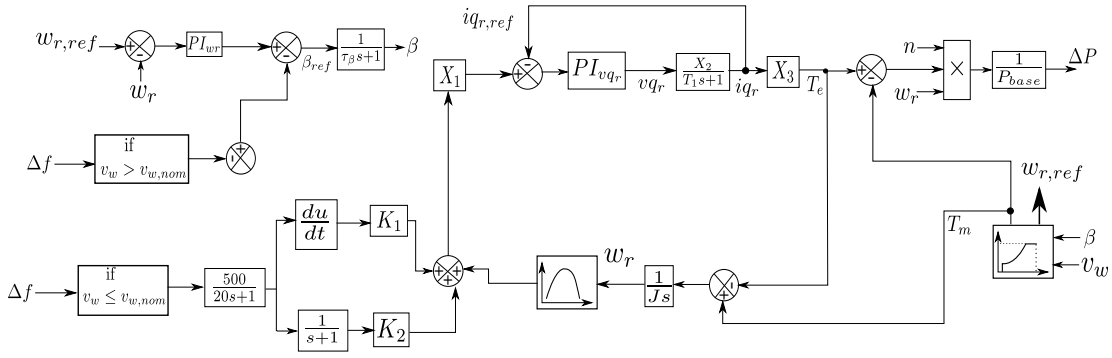


Fig. 6. Schematic diagram for wind-turbine integration in the LFC for a single area system (Moore and Ekanayake, 2009).

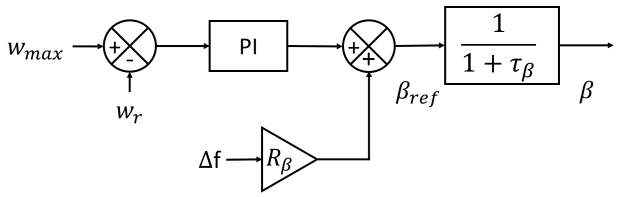


Fig. 5. Pitch control scheme for participation of wind-turbines in the LFC (based on (Ramtharan et al., 2007a)).

4. SIMULATION RESULTS

4.1 Benchmark Description

To perform the simulation of the LFC in a multi-area power system, the IEEE nine bus system is used (see (Anderson and Fouad, 2002)). This system is arbitrarily divided into in three areas, as illustrated figure 7. Each area has a generation unit and an associated load. Saturation blocks represent the operational limits of the units.

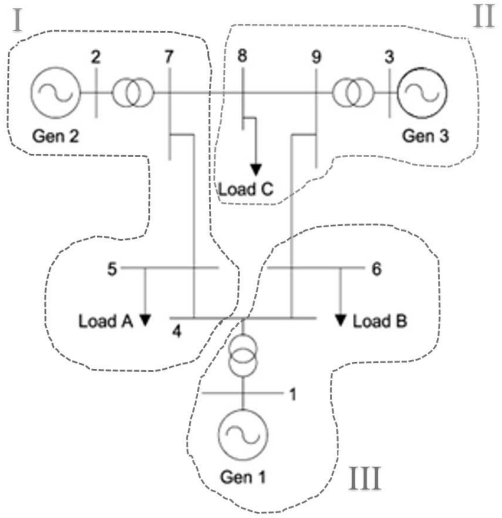


Fig. 7. IEEE 9 bus system with an arbitrary multi-area partition.

Simulations were performed using the parameters presented in table A.1 (see Appendix), taken from (Anderson and Fouad, 2002) and using 100 MVA as base power. The work is performed under the assumption of area 1 having a hydraulic turbine and generators 2 and 3 being gas turbines. Thus, each area could be represented by the scheme shown in Figure 2. Saturation blocks represent the generation units operational margins.

However, in area 3 the 50% of the conventional generation is replaced by wind-turbines with an equivalent power. A wind farm composed of 32 total turbines with a power of 2 MW for each unit. Figure 8 shows the modification of the LFC scheme to include wind turbines in the frequency regulation system.

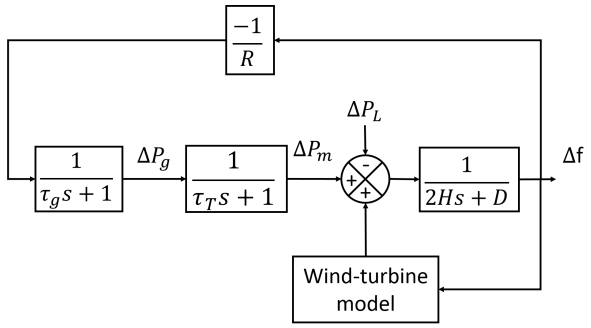


Fig. 8. LFC scheme including wind turbines for primary frequency regulation.

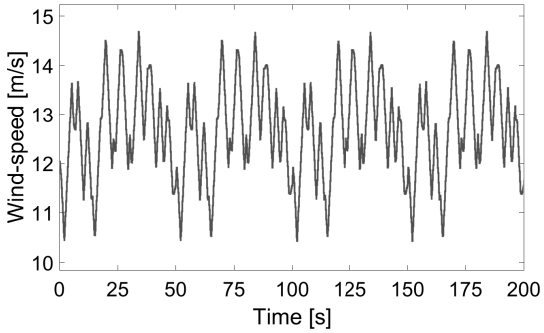


Fig. 9. Wind speed profile (from (Bernard et al., 2013)).

For simulation purposes, load disturbances are applied in each area as follows: In area 1 a perturbation of 0.01 p.u. is made at 90 seconds, in area 2 a perturbation of 0.08 p.u. is performed at 60 seconds, and in area 3 a load disturbance of 0.06 p.u. is applied at 30 seconds. Also, the wind speed profile employed to feed the wind units is shown in figure 9 as considered in in (Bernard et al., 2013).

4.2 Results

PI controllers are used in in the secondary control (AGC) of each area , and also for the regulation of the quadrature rotor current iq_r, and the pitch angle β in wind turbines. The parameters for these controllers were calculated with the Gradient Descent method, in Matlab Design Optimization-Based PID Controller toolbox (see table 1).

Table 1. Parameters for different PI controllers

Parameter	Area 1	Area 2	Area 3	iq_r	Pitch
P	0	0	0	0	7.19
I	-0.05	-0.05	-0.028	2.70	0.53

Figures 10 - 12 present the frequency deviations for each area with and without contribution of wind turbines in the LFC system of area 3. As the AGC, by design, leads to the local control of disturbances minimizing the effects in the other areas, the expected consequence is no sensible operational difference between the frequency deviations of each area. However, for each area, the inclusion of the wind farm in LFC of area 3 actually helps to compensate frequency deviations in the LFC, but only when there is enough wind to produce the required power. During the periods of low wind, disturbance effects are harder to compensate than the conventional case, due to the lack of renewable power in the system. Besides this, it is important to highlight that the presence of hydro generation in area 1 implies a greater inertia, diminishing even more the effects of frequency disturbances. The effects over areas 2 and 3 are pretty similar, because these areas have similar conditions.

The power exchanged between area 3 and the other areas is shown in figure 12. As it was said before, both frequency and power deviations are low when there is enough wind speed to sustain the power contribution of the wind farm. Once wind speed is low, area 3 needs to increase the power absorbed from the order areas in order to reduce the effects of local disturbances in the local area frequency.

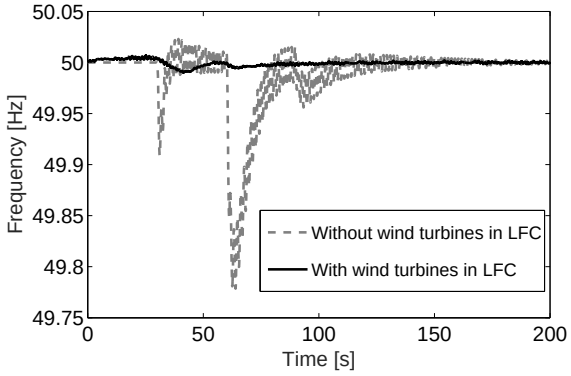


Fig. 10. Frequency deviation in area 1.

The responses of the rotor voltage vq_r and the pitch angle β_{ref} are depicted in figures 14 and 15, respectively. These figures show how the participation of wind turbines in LFC can be more stressful for them. However, a less aggressive performance could be obtained with a better tuning of the involved PI controller, or by the implementation of another kind of controllers.

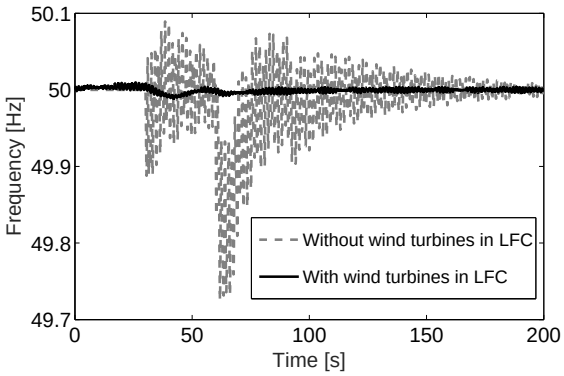


Fig. 11. Frequency deviation in area 2.

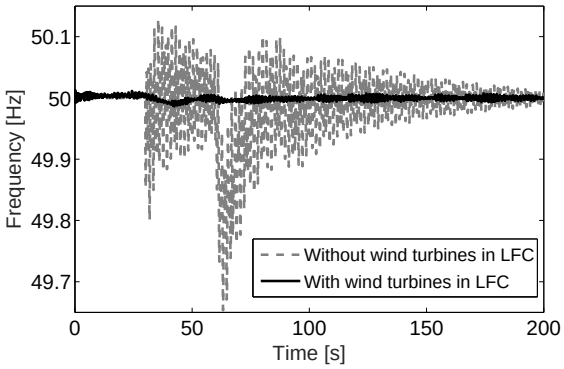


Fig. 12. Frequency deviation in area 3.

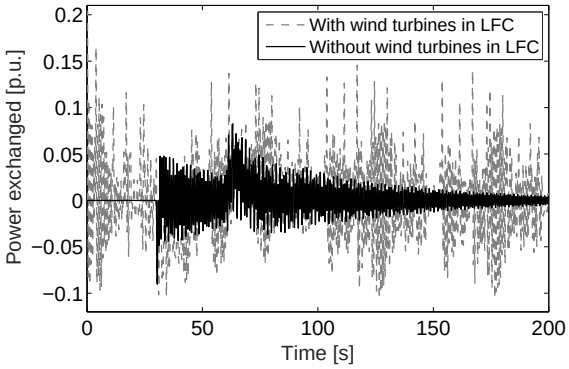


Fig. 13. Power exchange for area 3.

5. CONCLUSION

The simulation of the performance of the LFC scheme for a multi-area power system, with participation of DFIG turbines in the frequency control loops, through the *synthetic inertia method*, with PI controllers for the AGC control was presented. The results show that wind turbines are useful for frequency regulations tasks in primary control. However, the variability of wind and the effects of the decreasing inertia from conventional units can be dangerous for frequency performance. Additionally, it is also shown that the participation of wind turbines in LFC introduces more stress in the operation of these units, requiring the exploration of control techniques that help to reduce these efforts for the wind units.

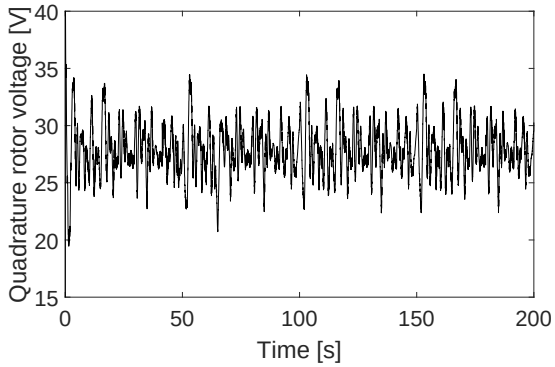


Fig. 14. Control action for the variable vq_r.

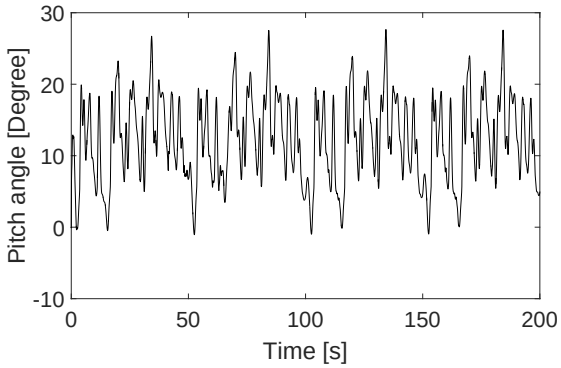


Fig. 15. Control action for the variable β_{ref}.

REFERENCES

Anderson, P.M. and Fouad, A.A. (2002). *Power System Control and Stability*. second edition.

Bernard, M.Z., Mohamed, T.H., Ali, R., Mitani, Y., and Qudaih, Y.S. (2013). PI-MPC Frequency Control of Power System in the Presence of DFIG Wind Turbines. 2013(September), 43–50.

Bevrani, H. (2009). *Robust Power System Frequency Control*. Lc. doi:10.1007/978-3-319-07278-4.

Camblong, H., Vechiu, I., Guillaud, X., Etxeberria, A., and Kreckelbergh, S. (2014). Wind turbine controller comparison on an island grid in terms of frequency control and mechanical stress. *Renewable Energy*, 63, 37–45. doi:10.1016/j.renene.2013.08.045.

Daz-Gonzlez, F., Hau, M., Sumper, A., and Gomis-Bellmunt, O. (2014). Participation of wind power plants in system frequency control: Review of grid code requirements and control methods. *Renewable and Sustainable Energy Reviews*, 34(0), 551 – 564. doi: http://dx.doi.org/10.1016/j.rser.2014.03.040.

de Almeida, R.G., Castronuovo, E.D., and Peças Lopes, J.A. (2006). Optimum generation control in wind parks when carrying out system operator requests. *IEEE Transactions on Power Systems*, 21(2), 718–725. doi: 10.1109/TPWRS.2005.861996.

Horta, R., Espinosa, J., and Patino, J. (2015). Frequency and voltage control of a power system with information about grid topology. In *Automatic Control (CCAC), 2015 IEEE 2nd Colombian Conference on*, 1–6. IEEE.

Moore, I. and Ekanayake, J. (2009). Frequency response from wind turbines. *Universities Power Engineering Conference (UPEC), 2009 Proceedings of the 44th International*, 1–5.

Rahmann, C. and Castillo, A. (2014). Fast Frequency Response Capability of Photovoltaic Power Plants: The Necessity of New Grid Requirements and Definitions. *Energies*, 7(10), 6306. doi:10.3390/en7106306.

Ramtharan, G., Ekanayake, J., and Jenkins, N. (2007a). Frequency support from doubly fed induction generator wind turbines. *IET Renewable Power Generation*, 1(1), 3–9. doi:10.1049/iet-rpg:20060019.

Ramtharan, G., Jenkins, N., and Anaya-Lara, O. (2007b). Modelling and control of synchronous generators for wide-range variable-speed wind turbines. *Wind Energy*, 10(3), 231–246. doi:10.1002/we.219.

Saadat, H. (1999). Power System Analysis Hadi Saadat.pdf.

Thomsen, S.C. (2006). *Nonlinear control of a wind turbine, ME Thesis*. Ph.D. thesis, Technical University of Denmark.

Valencia, F., Patino, J., and Espinosa, J. (2012). A performance comparison for wind power integration into the grid system. In *Alternative Energies and Energy Quality (SIFAE), 2012 IEEE International Symposium on*, 1–5. Barranquilla, Colombia. doi: 10.1109/SIFAE.2012.6478879.

Appendix A. SIMULATION PARAMETERS

Table A.1. IEEE nine bus system parameters (from Anderson and Fouad (2002))

Parameter	**Value**
H_1	23.64 s
H_2	6.4 s
H_3	1.505 s
MVA_{nom1}	247.5
MVA_{nom2}	192
MVA_{nom3}	128
$D1, D2, D3$	0.8
$Tg1, Tg2, Tg3$	0.2
$T_\tau 1, T\tau 2, T\tau 3$	0.3
T_{12}	2.064 p.u.
T_{13}	6.1191 p.u.
T_{23}	14.4353 p.u.
R_1	2 p.u.
R_2	10 p.u.
R_3	7.5019 p.u.
B_1	2.8 s
B_2	10.8 s
B_3	8.3 s

Table A.2. Wind-turbine model parameters (Ramtharan et al. (2007a) and Moore and Ekanayake (2009)).

Parameter	**Value**
P_{nom}	2 MW
V_{nom}	966 V
K_1	5000 Nm
K_2	2000 Nm
T_w	1
K_a	500
T_a	20
R_s(Stator resistance)	0.00491 p.u.
X_{ls}(Stator reactance)	0.09273 p.u..
X_m(Magnetization reactance)	3.96545 p.u.
R_r(Rotor resistance)	0.00552 p.u
X_{lr}(Rotor reactance)	0.1 p.u.
$H = \frac{1}{2} J \frac{wnom^2}{VA_{nom}}$	4.5 s
J (Inertia moment)	506.6059 Kgm^2.
P_{base} (Power base)	128 Mw.

Methods for General Motor Skills based on Neuroevolution to Stabilize a Biped Robot Simulation

Sergio Zapata*, Iván D. Mora**
Gustavo Suarez***

** Semillero de Investigación en Automática y Diseño A+D, Universidad Pontificia Bolivariana, Medellín, Colombia (e-mail: sergio.zapatave@upb.edu.co)*
*** Semillero de Investigación en Automática y Diseño A+D, Universidad Pontificia Bolivariana, Medellín, Colombia (e-mail: ivan.mora@upb.edu.co)*
**** Grupo de Investigación Matemáticas, Universidad Pontificia Bolivariana, Medellín, Colombia (e-mail: gustavo.suarez@upb.edu.co)*

Abstract: One of the main focuses of humanoid robotics is bipedal locomotion which requires a series of continuous phases, called gait cycle. Here is essential to maintain balance, compensate the dynamic of the whole multivariable and nonlinear system. Despite good bipedal gait algorithms have been achieved in various researches, performance is still not enough to be compared to biological systems. Several research projects have tried to compensate this through methods in machine learning, to autonomously generate gaits and stabilize robots. In this paper is firstly presented how a neuroevolution algorithm achieves the stability of a biped robot simulation in the sagittal plane that starts in unstable position. Then three different methods are proposed and analyzed in order to generate full body motions in any state perceived by the robot, complying with the specified task by means of the motor skills learned through neuroevolution together with the implementation of another feedforward neural network.

Keywords: Artificial Intelligence, Neuroevolution, Neural Network, Genetic Algorithm, Biped Locomotion, Stability, Nonlinear Control, Inverted Pendulum.

1. INTRODUCTION AND BACKGROUND

Humanoid robotics has made great strides in the last 30 years regarding the general control of all the robot limbs that allow them to perform complex tasks. Research has focused on human bipedal locomotion, because a machine with this capability easily works in human environments, rather than a conventional robot equipped with wheels.

However this presents great control challenges, caused by the large number of variables involved to achieve gaits without falling in a non-ideal environment. Maintaining balance is a fundamental and necessary requirement in implementing such systems (Siciliano & Khatib, 2008).

Various approaches have been studied to generate dynamically stable legged locomotion, such as the ZMP (Zero Moment Point) (Vukobratović & Borovac, 2004). Nevertheless, it is still difficult to find a robot that outperforms a biological system, leading this to research in self-teaching machines. In this way, many projects have focused on generating gaits in different kinds of robots using machine learning (Tedrake et al., 2005; Manoonpong et al., 2007; Allen & Faloutsos, 2009; Clune et al., 2009; Yosinski et al., 2011).

Though it is essential to generate whole body motions, learned by the own robot, the focus of this paper is to understand that is also important the implementation of some motor skills to achieve robot stabilization, in this case of a biped robot in a phase of the gait cycle, restricted by the sagittal plane.

While there are multiple methods in machine learning, neuroevolution algorithms are promising in this kind of problems (Gomez & Miikkulainen, 2003). Research in these algorithms has resulted in different methods in their application as it is NEAT (Stanley & Miikkulainen, 2002a), HyperNEAT (Stanley et al., 2009) or TWEANNS (Stanley & Miikkulainen, 2002b), among others. These modify the topology of the neural network as it is produced the evolution of the individuals (Back et al., 1997).

These methods have shown very good results in benchmarks such as the simultaneous stabilization of two inverted pendulums, exceeding in performance to other alternatives in unsupervised learning, e.g. reinforcement learning (Gomez & Miikkulainen, 1999).

However, the classic implementation of neuroevolution algorithms, where is evolved a neural network of fixed topology through genetic algorithms, can also have satisfactory results in several tasks related to robotics (Siciliano & Khatib, 2008).

Every individual in the genetic algorithm starts in an initial state and then when each iteration occurs it jumps to a new state, which will be the result of actions taken in the previous.

Thus the neural network acts as the agent of a reinforcement learning algorithm, in which for each state, is executed a particular action. However, while reinforcement learning

knows the actions to be made in every state, neuroevolution provides an overview of the problem and consequently the neural network is not learned for the state, but for the ultimate goal of the task.

When the optimum individual is found for the current task, a series of states and actions occurs, but the neural network is only functional for the evaluated situation and the states seen by the agent, i.e. the neural network might not be useful for other initial conditions or states (Whiteson, 2012).

In this regard, problems with large number of variables, such as humanoid robots (Benbrahim & Franklin, 1997), generate numerous states for which the neural network found, should work. Therefore if the goal is to make a robot control, capable to respond to certain situations properly, even unknown, it is required the generalization of what was learned by the trained neural network.

In this paper is presented the development of a neuroevolution algorithm to achieve the stability of a biped robot simulation in the sagittal plane that starts in an unstable position.

In this way is not taken into account the balance in the horizontal plane and also the robot foot does not slide, it is only evaluated the contact point of the foot with the surface on its front and rear boundaries that can be considered as two passive degrees of freedom, where is involved the interaction between the machine and the environment.

Furthermore three different methods of control are proposed and analyzed, based on a feedforward neural network, with the purpose to generate full body motions in any state or condition perceived by the robot, fulfilling with the specified task of balance.

Therefore, the main focus of this paper, which is the result of an ongoing research project in humanoid robotics at UPB, is to explore the generalization of what was learned from neuroevolution to achieve general motor skills for the biped robot in a controlled environment.

It should be pointed out that the learning process might be achieved with other strategies, neuroevolution was chosen because of the good results found in other tasks, as mentioned before, this means the focus of this paper is not to compare its performance with other unsupervised learning methods.

2. DESCRIPTION OF THE PROBLEM AND SIMULATION

In the context of humanoid robotics, biped locomotion is fundamental to make the robot move efficiently, emulating human behavior. In most of the gait phases implies that the robot stays on one foot, being essential a static equilibrium control, as a first step towards a subsequent locomotion algorithm. Thus it is intended that the robot stays in balance at all times, which is reduced to keep the system's center of mass within the support polygon of the foot.

To simplify the problem, the robot geometry is restricted to the sagittal plane, here is analyzed the center of mass, which position is function of seven revolute joints, i.e. flexion and extension of knee, legs, arms and the ankle of the supporting foot (see Fig. 1).

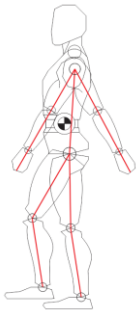


Fig. 1. Sagittal plane and joints of the robot simulation for the implementation of stability control. The red lines represent the links of the robot.

The analysis of the center of mass is based on a system of particles (Marion, 2000), according to the mass of each robot link. The weight and distances are taken from realistic data of a humanoid robot currently in development at UPB.

The center of mass position in $\hat{\imath}$ and $\hat{\jmath}$ (Landau & Lifshitz, 1988), is found as follows,

$$r_{CM}\hat{\imath} = \frac{1}{M}\sum_{i=0}^{n} \tau_i, \tag{1}$$

$$r_{CM}\hat{\jmath} = \frac{1}{M}\sum_{i=0}^{n} m_i |r_i| \sin\theta_i. \tag{2}$$

Where τ, m, θ, and r, are respectively the torque, mass, angle and distance of each particle, M is the total mass of the system. Based on this position, the corresponding to each edge of the foot is calculated, now the robot's behavior is simplified to a simple inverted pendulum.

3. NEUROEVOLUTION ALGORITHM

Every Markov state in the robot simulation is completely determined by 12 variables, including the 7 joints of the robot, the position error (e), angular position (θ), angular speed (ω), lever arm (l) and angular acceleration (α) of the inverted pendulum.

The neural network of fixed topology consists of 12 input neurons, corresponding to the variables listed above, 13 neurons in the hidden layer with hyperbolic tangent activation function and 7 neurons in the output layer with identity activation function. This output is a small increase in the angle of the 7 joints actuated, limited by the actual speed of the actuators (see Fig. 2).

A common criterion to choose the number of neurons in the hidden layer is the arithmetic mean between the inputs and outputs of the neural network. Nevertheless, a set of 90 tests with 7 to 15 neurons in this layer showed that 13 neurons presented slightly better results in the neuroevolution task.

A genetic algorithm evolves the neural network, i.e. optimizes its weights and biases to generate a stabilization move of the robot (Goldberg & Holland, 1988).

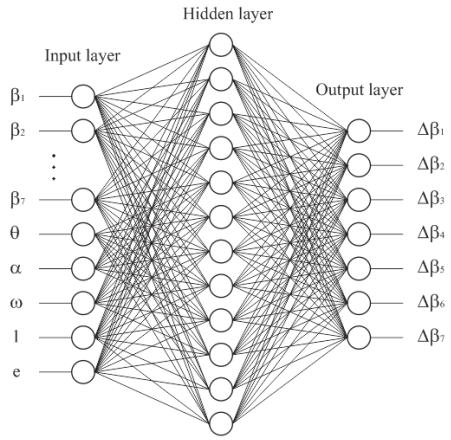


Fig. 2. Artificial neural network to be trained through genetic algorithms, it consists of 12 inputs, 13 neurons in the hidden layer and 7 neurons in the output layer.

The neural network is formed by 247 weights and 20 biases that determine the genotype of each individual of the genetic algorithm. A number of 50 individuals per generation with a crossover of 90% were configured in the genetic algorithm, to achieve good diversity and speed up the learning process. This kind of configuration converges rapidly in high dimensionality cases (Goldberg & Deb, 1991).

On the other hand, to keep the robot balanced, the inverted pendulum that represents it must remain completely vertical, i.e. in 0°, regardless if the foot lies flat on the floor or not. Thus, the fitness function is determined from the MSE (Mean Squared Error) between the reference of 0° and the pendulum angle during the simulation time.

With this, the genetic algorithm must find the individual that configures a neural network that generates a minimum MSE, and hence maintain the robot in balance.

4. RESULTS AND METHODS

Each time the neuroevolution is executed, the robot starts with a random set of angles, assumed as the initial conditions of the system, these assures a center of mass beyond the front edge of the foot, thus if no action is performed the robot will fall forwards because of gravity. Simulation is then run for 3 seconds in which is intended the robot remains stable. The control action is exemplified in Fig. 3.

To determine the effectiveness of the algorithm, were executed 150 tests, of which 89 achieved the expected balance, converging to a low MSE, in no more of 200 generations as shown in Fig. 4. The stabilization process in the simulation, for a particular condition, can be seen in Fig. 5. The corresponding progress in time of angle θ is presented in Fig. 6.

The randomness of the initial angles in some cases led to postures impossible to stabilize with a neural network, it can be said, that even a human would fall.

Thus is demonstrated that neuroevolution was effective for this task of robot stabilization, but this does not indicate that other methods in neuroevolution or in unsupervised learning achieve better or worse results for this simulation. This is not analyzed because the main focus of this paper is not to determine the performance of the various possibilities in this respect.

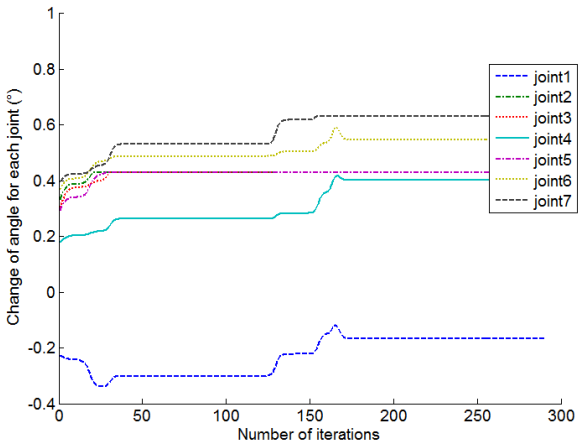


Fig. 3. Change of angles for each iteration for a particular case. The variations in the curves of each joint demonstrate the adjustments produced by the neural network to maintain the robot balanced during the simulation.

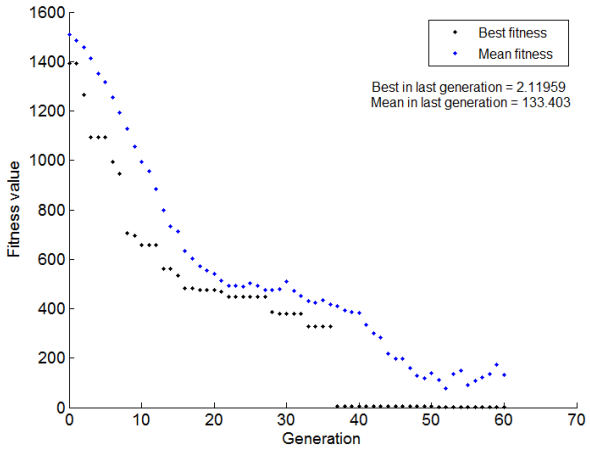


Fig. 4. Best and mean fitness for each generation of the genetic algorithm. For this particular case, the minimization of the MSE is achieved in 60 generations. The best fitness in the last generation is 2.11959, an acceptable value for the process, because with the movement of the robot to stop the falling and stay in 0°, is not possible a null MSE.

On the other hand, notorious changes were observed in the stabilization process under low variations in the states, demonstrating the high instability of the system. As pointed out previously, the algorithm is only useful while the initial conditions and the subsequent states do not change, otherwise the neural network is not robust enough to deal with other cases, and even less in systems like the evaluated.

The non-robustness of the neural network was evidenced when a slight change of 0.01° in only one angle of the initial conditions of a previous successful case, produced an entirely different result in the simulation, leading to the fall of the robot.

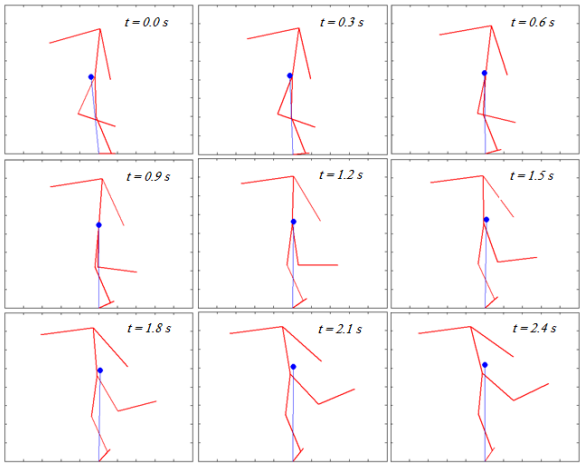


Fig. 5. Simulation of the bipedal robot in the sagittal plane, executing one stabilization process. The red lines represent the robot links and the blue line represents the inverted pendulum. As can be seen, the movement of the robot maintains it vertical.

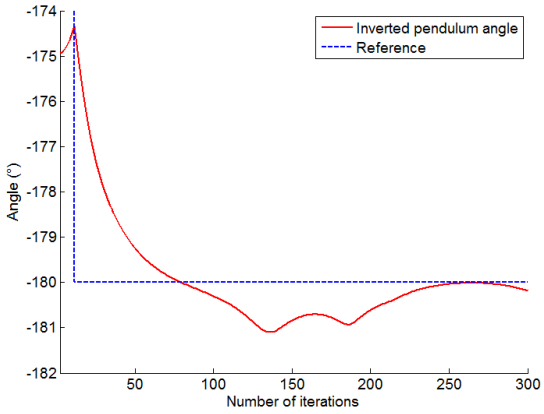


Fig. 6. Change of angle θ of the inverted pendulum representation. Here the reference is -180°, i.e. when the pendulum is vertical, moreover the actions of the neural network begin 10 iterations after the simulation is started, in this way the robot should recover from falling.

The mere implementation of neuroevolution in a real robot is unpractical, because the learning process with the genetic algorithm, for a particular case, requires a large number of tests, and the solution could be only useful for that situation.

In this sense, an offline learning of different cases in conjunction with the generalization of the motor skills and combining it with some online learning could be very useful in solving different control tasks on real robots. Therefore, to increase neural network robustness and in order to achieve an overall stability control, three approaches presented below were proposed.

4.1 Retraining of the Neural Network

One of the main causes of the low robustness is because the neural network was only trained for a particular case, therefore a possible solution is to retrain it again with the genetic algorithm, but starting with an initial population based on the individuals of the latest generation obtained in the previous case.

When evaluated this method was found that the genetic algorithm evolved again the neural network finding an optimum individual, but leaving behind the weights and biases that formed the previous neural network, being only useful again for the current case.

Although this method was not successful in the generalization of the motor skills learned previously, it provided a better overview in how to treat this problem. In this way the next two methods separate the learning process and the final motion control.

4.2 Individuals as Training Data for Neural Network

This method seeks to train a second neural network by backpropagation, its inputs are the 12 variables that define the states, and the outputs are the 267 weights and biases that determine the neural network that ultimately generates the actions for the robot. Therefore, it is intended that under a known initial condition, a specific neural network executes the right set of actions that achieve robot balance.

The neural network is made of 186 neurons in the hidden layer, chosen arbitrarily, requiring with this an updating process of 52347 weights and biases, leading to a slow learning rate and high computational cost.

Furthermore, is not guaranteed for unknown initial conditions that the neural network output defines correctly the 267 weights and biases for a neural network that achieves stability, this is because the successful solutions of the problems already executed, might be formed by very different weights, even for similar cases.

Because of the high computational cost and the other reasons exposed, this method was not tested.

4.3 Training of Neural Network with State-Action Data

When simulation is executed, the robot begins in an initial state and through the actions made inside the environment, passes to another state. In this way, although the neural network is learned according to the overview of the genetic algorithm, for each state executes an action that determines a subsequent state and its sequence will define the success rate of the process in the corresponding task.

Specifically, each simulation is executed for 3 seconds, in 300 iterations, thus having 300 different states and actions for every case evaluated. Therefore, in this method is intended to train a second neural network with sets of state-action. A diagram of this method is shown in Fig. 7.

This neural network, called neural network for robot control, is similarly formed to the executed in neuroevolution, with 12 input variables and 7 outputs, but now with 70 neurons in the hidden layer (see Fig. 8). Because the network is trained by backpropagation with the Levenberg–Marquardt algorithm, based on the state-actions sets obtained in the neuroevolution process, the possibility of losing usefulness for different cases is reduced, being with this a data fitting problem.

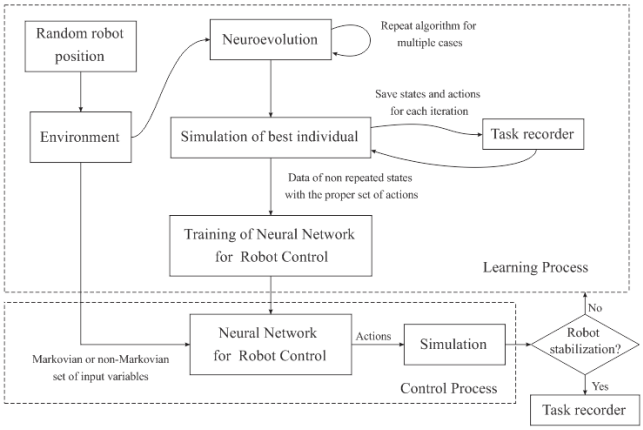


Fig. 7. Diagram of the third method proposed. Its main goal is to maintain a constant learning, initially should be prioritize the learning process, afterwards the control process, however at this stage whether stabilization is not achieved, learning must be done again.

The 89 cases that reached the stabilization produced 26700 sets of state-action, where the repeated states were deleted, according to the set of actions that achieved the least MSE.

The large number of neurons in the hidden layer was required to get a better performance in the data fitting of the neural network.

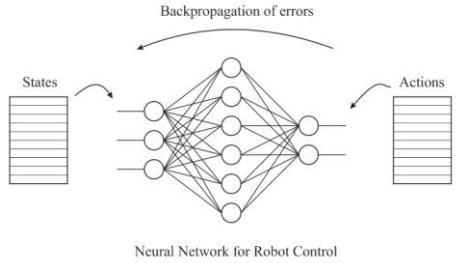


Fig. 8. Neural network for robot control, trained by backpropagation with the states-actions data.

Thereby, after being trained, tests were conducted with initial conditions already proved, obtaining actions for each state very similar to those obtained by neuroevolution, in most of the cases with an error not greater than 0.07% in the angular change of the seven joints. This similarity for a particular case can be observed in Fig. 9.

In order to have a good fitting of all the data, slight variations in the output are inevitable with the backpropagation training. Therefore, even with the low error observed, sometimes the robot simulation is no capable to maintain balance, when it should do. This demonstrates the instability of the system, where a minor change in actions, leads to a set of new states slightly different, and that together determine a different behavior than expected.

Nevertheless, this does not demonstrate the capability of the method to generate proper motor skills in other kind of tasks. With a less unstable system or unstable at all, the robot's behavior could be very close to the expected and effectuate good general motor skills (Peters, 2007), i.e. generalize a behavior through the neural network for robot control based on the experiments of the neuroevolution.

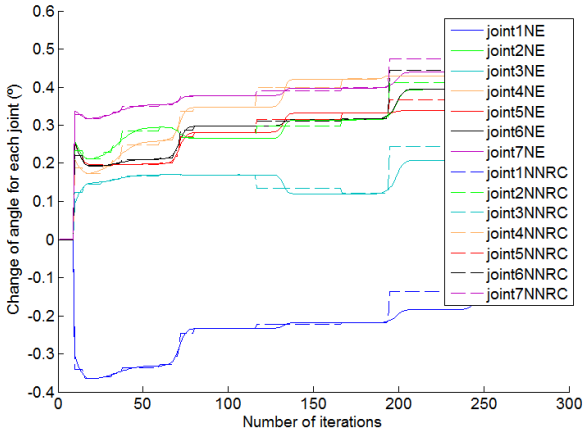


Fig. 9. Comparison between the actions commanded with neuroevolution (NE) and the actions obtained with the neural network for robot control (NNRC), denoted in dashed lines.

5. DISCUSSION

The neuroevolution process and the implementation of the third method was also applied to a simulation in which some variables of the system are unknown, specifically the angular acceleration (α) and the error (e). These cases could be considered as non-Markovian tasks.

For these cases, the neuroevolution process also achieved, in 80% of the tests, the system stabilization. The generalization of the task through the neural network for robot control was likewise effective giving the right actions for the states seen by the agent, but with the fails explained in the previous section.

Although the said failures, that are mostly caused for the system instability, as said before, a better performance could be seen in non-unstable systems, like the implementation in robots for objects manipulation (Pastor, 2009; Pastor, 2012; Lenz et al., 2015), or full body behaviors in order to create complex motor skills in locomotion tasks (Schaal, 1999), that are also of great interest for research in humanoid robotics.

6. CONCLUSIONS AND FUTURE WORK

In this paper was presented how through a neuroevolution algorithm, with a neural network of fixed topology, was achieved in 89 of 150 tests, that the simulation of a bipedal robot in the sagittal plane kept the balance over a period of 3 seconds, demonstrating the effectiveness of this class of evolutionary algorithms to provide reliable solutions in multivariable, nonlinear and unstable systems. Even in non-Markovian tasks.

Besides was demonstrated that the neuroevolution algorithm by itself is only useful for the agent with the corresponding states and initial conditions, being found that a variation of 0.01° in one joint, could lead the robot to lose the balance, this also caused by unstable nature of the system.

In order to generate a control useful for different states and then being able to produce general motor skills for the task, three approaches were proposed.

The first method, which consisted in retraining the neural network by the neuroevolution algorithm based on the previous case individuals, did not achieve good results, because the retraining process modifies the weights of the network and is not functional again for the previous case.

The second method, in which was intended to get at the output of a feedforward neural network the individuals that configured the neural network, that ultimately would give the actions, was not carried out due to the high computational cost and the limited guarantees that this method could find right individuals to perform the task.

In the third method, a neural network was trained by backpropagation with state-action datasets, obtained in the tests made through neuroevolution, had promising results being capable to output the right actions according to the state seen by the robot, even without being previously learned.

It is concluded that the third method is the most appropriate approach to generate the control of this system and other kind of tasks through full body motions.

This last approach allows the system from a simulation, learn by itself according to its surrounding environment, being able later to generalize a proper behavior to achieve a specific task, facilitating its implementation on the real robot.

In this way, research will continue around a larger number of tests to strengthen control and then assess its applicability in other types of robotic systems.

In addition the main focus of this research is to develop such controls in more complex simulations, with the system's behavior in a three dimensional environment and with the full dynamics of the robot, and thus being able to be later extrapolated to a real robot.

REFERENCES

Allen, B. F., & Faloutsos, P. (2009). Evolved controllers for simulated locomotion. In Motion in games (pp. 219-230). Springer Berlin Heidelberg.

Back, T., Fogel, D. B., & Michalewicz, Z. (1997). Handbook of evolutionary computation. IOP Publishing Ltd..

Benbrahim, H., & Franklin, J. A. (1997). Biped dynamic walking using reinforcement learning. Robotics and Autonomous Systems, 22(3), 283-302.

Clune, J., Beckmann, B. E., Ofria, C., & Pennock, R. T. (2009, May). Evolving coordinated quadruped gaits with the HyperNEAT generative encoding. In Evolutionary Computation, 2009. CEC'09. IEEE Congress on (pp. 2764-2771). IEEE.

Goldberg, D. E., & Deb, K. (1991). A comparative analysis of selection schemes used in genetic algorithms. Foundations of genetic algorithms, 1, 69-93.

Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning. Machine learning, 3(2), 95-99.

Gomez, F. J., & Miikkulainen, R. (1999, July). Solving non-Markovian control tasks with neuroevolution. In IJCAI (Vol. 99, pp. 1356-1361).

Gomez, F. J., & Miikkulainen, R. (2003). Robust non-linear control through neuroevolution. Computer Science Department, University of Texas at Austin.

Landau, L., & Lifshitz, E. (1988). Curso de física teórica. Barcelona [etc.]: Reverté.

Lenz, I., Lee, H., & Saxena, A. (2015). Deep learning for detecting robotic grasps. The International Journal of Robotics Research, 34(4-5), 705-724.

Manoonpong, P., Geng, T., Kulvicius, T., Porr, B., & Wörgötter, F. (2007). Adaptive, fast walking in a biped robot under neuronal control and learning. PLoS Comput Biol, 3(7), e134.

Marion, J. (2000). Dinámica de las partículas y sistemas. Barcelona: Editorial Reverté.

Pastor, P., Hoffmann, H., Asfour, T., & Schaal, S. (2009, May). Learning and generalization of motor skills by learning from demonstration. In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on (pp. 763-768). IEEE.

Pastor, P., Kalakrishnan, M., Righetti, L., & Schaal, S. (2012, November). Towards associative skill memories. In Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on (pp. 309-315). IEEE.

Peters, J. R. (2007). Machine learning of motor skills for robotics (Doctoral dissertation, University of Southern California).

Schaal, S. (1999). Is imitation learning the route to humanoid robots?. Trends in cognitive sciences, 3(6), 233-242.

Siciliano, B., & Khatib, O. (2008). Biped Robots in the ZMP Scheme. Handbook of robotics, Springer. ISBN: 978-3-540-23957-4.

Stanley, K. O., & Miikkulainen, R. (2002a). Evolving neural networks through augmenting topologies. Evolutionary computation, 10(2), 99-127.

Stanley, K. O., & Miikkulainen, R. (2002b). Efficient evolution of neural network topologies. In Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on (Vol. 2, pp. 1757-1762). IEEE.

Stanley, K. O., D'Ambrosio, D. B., & Gauci, J. (2009). A hypercube-based encoding for evolving large-scale neural networks. Artificial life, 15(2), 185-212.

Tedrake, R., Zhang, T. W., & Seung, H. S. (2005, June). Learning to walk in 20 minutes. In Proceedings of the Fourteenth Yale Workshop on Adaptive and Learning Systems (Vol. 95585).

Vukobratović, M., & Borovac, B. (2004). Zero-moment point—thirty five years of its life. International Journal of Humanoid Robotics, 1(01), 157-173.

Whiteson, S. (2012). Evolutionary computation for reinforcement learning. In Reinforcement Learning (pp. 325-355). Springer Berlin Heidelberg.

Yosinski, J., Clune, J., Hidalgo, D., Nguyen, S., Zagal, J., & Lipson, H. (2011). Evolving robot gaits in hardware: the HyperNEAT generative encoding vs. parameter optimization. In Proceedings of the 20th European Conference on Artificial Life (pp. 890-897).

Model Based Fault Detection and Isolation of a Reverse Osmosis Desalination Plant

Soto Angles Mario *. Pérez Zúñiga Gustavo*. Sotomayor Moriano Javier*

** Pontifical Catholic University of Peru, Av. Universitaria 1801, Lima, Perú Email: {mario.soto, gustavo.perez, jsotom} @pucp.pe*

Abstract: In this paper, a model based diagnosis system for a reverse osmosis desalination module with spiral wound configuration is developed. At first, a mathematical model based on differential and algebraic equations of the reverse osmosis system is obtained. From this model, a structural analysis technique is performed which allows us to obtain a structural model of the plant defined by a set of constraints. After the structural model a set of analytical redundancy relations (ARR) is obtained by applying the ranking algorithm of constraints. The comparison between the system in nominal operation with the system under different faults shows that all faults of interest were detectable and isolable.

Keywords: Fault Diagnosis, Reverse Osmosis, Analytical Redundancy Relations.

1. INTRODUCTION

According to the United Nations, close to 1.2 billion people already live in areas where freshwater is scarce. Another 1.6 billion people face chronic economic water shortage. While freshwater accounts for only 2.5%, seawater and brackish water found in oceans, seas and underground cover 97.5% of the total water in the world. Nowadays, obtaining this resource through reverse osmosis (RO) desalination method is an economically viable energy alternative (Dessouky, 2002).

Currently, installing RO desalination plants is the trending, however these systems are subject to different types of faults: in actuators such as high pressure pump, acid dosing pump, valves, and measures as flow, conductivity and temperature sensors; likewise common typical internal parameters faults of the membrane are weathering, fouling and scaling. (Gambier, 2009)

A Fault Tolerant Control (FTC) for a RO system based was developed by McFall (2007); this control was based on physical switching logic reconfiguration by installing redundant control valves, which is a disadvantage due to the high implementation costs. Gambier (2009) presents a mathematical model of a laboratory reverse osmosis plant for a FDI system design considering faults in sensors, actuators and faults in RO module such as block of a pipeline, scaling/fouling and leaks; a FTC based on control loops reconfiguration were developed in this work, but not clear isolability analysis between considered faults was considered.

Garcia (2011) performed a monitoring fault detection system based on principal component analysis (PCA) technique for a simulated RO desalination plant, this technique allows us to detect faults like offsets in pressure, temperature and concentration sensor, also blockages in filters and breakages in the membranes; but does not present faults in actuators like high pressure pump and acid dosing pump, and neither a fault isolation system is presented. Palacin (2011) presented an enhanced dynamic library of reverse osmosis plants (ROSIM) used for simulation, optimization, fault detection and a simple fault tolerant control; however faults in actuators and sensors were not considered in the work.

There are several methods for diagnosis and detection of faults with different characteristics and application fields; these include methods based on identification, in which the failures are reflected in changes of certain model parameters; knowledge-based methods using artificial intelligence techniques (Isermann, 2005).

Another approach is that concerning to the model based methods; one of them consists on the analytical redundancy method that is based on the knowledge of the set of algebraic and differential equations that conforms the process model. This method allows us to understand internal parameters that can be considered to simulate typical faults in the RO membrane; likewise this method also allows us to improve the isolability analysis of faults by adding additional redundant equations (Blanke, 2006).

This paper is structured as follows: in section 2 a detailed mathematical model is obtained where the main variables in the process are explained. In section 3 a model based fault diagnosis (MBFD) system for a RO desalination module with spiral wound configuration is developed. Section 4 consists on the simulation of the system under different faults and analysis of detectability and isolability. Section 5 presents conclusions of this

work.

2. MATHEMATICAL MODELLING OF THE REVERSE OSMOSIS SYSTEM

In Figure 1 the MBFD structure is presented and it consists in the continuous monitoring of the consistency between the output signals of the RO module with the mathematical model under the same input signals, the differences between these outputs are disclosed through a set of equations called *residuals*, from these residuals a set of analytical redundancy relations (RRA) are obtained, which depend only in terms of known variables. This set of RRA is those that allow us to detect changes when comparing the plant in normal mode of behavior and under faults. In general, the number of RRA must always exceed the number of faults considered. In this work, we consider the inclusion of an additional set of equations (redundancy) in the model for obtain a greater amount of RRA and improve isolability task.

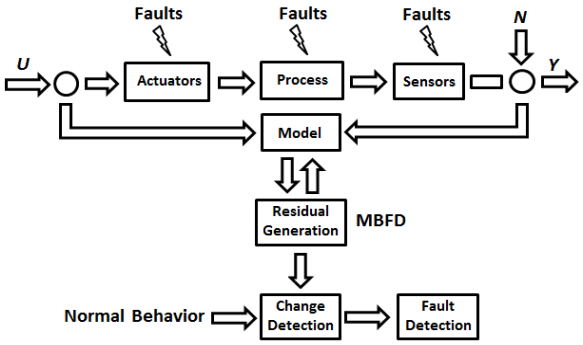


Figure 1. MBFD Structure Designed.

The MBFD method requires the thorough knowledge of the mathematical model of the plant or process under study; this is the reason for the development and complete understanding of the equations and parameters involved in the reverse osmosis desalination process. The mathematical model developed here is based on the spiral wound reverse osmosis membrane configuration because it allows us to seize the largest possible filter area compared with other configurations.

2.1. System Decomposition

System decomposition is necessary for obtaining a mathematical model of the RO module (Gambier, 2009); for the correct material balance, the system was divided into three subsystems: membrane subsystem, rejection subsystem and permeate subsystem. The considered variables are flows, pressures and concentrations.

No energy balances were made because RO works at room temperature and there is not any phase change. Subsystem decomposition is shown in Figure 2.

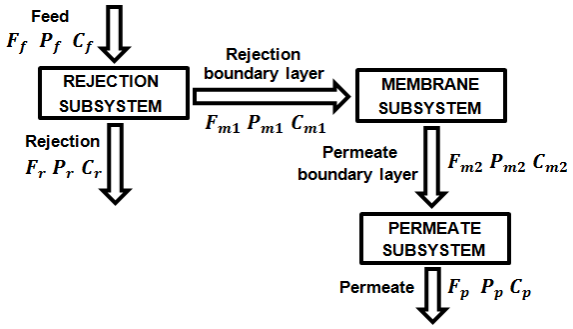


Figure 2. RO module decomposition.

2.2. Rejection subsystem

For Rejection Subsystem in Figure 2 is performed:
Global material balance:

$$dm_r/dt = F_f - F_{m2} - F_r \quad (1)$$

Where: m_r is the rejection holdup (kg), F_f is the feed flow water (kg/s), F_{m2} is the permeate boundary layer flow (kg/s), F_r is the rejection flow water (kg/s).
Partial material balance:

$$dC_r/dt = (1/m_r)\left[F_f(C_f - C_r) - F_p(C_p - C_r)\right] \quad (2)$$

Where: C_r is the rejection concentration (kg/m^3), C_f is the feed water concentration (kg/m^3), C_p is the permeate water concentration (kg/m^3), F_p is the permeate flow (kg/s).

2.3. Permeate Subsystem

For Permeate Subsystem in Figure 2 is performed:
Global material balance:

$$dm_p/dt = F_{m2} - F_p \quad (3)$$

Where m_p is the permeate holdup (kg)
Partial material balance:

$$dC_p/dt = (1/m_p)\left[F_p(C_{m2} - C_p)\right] \quad (4)$$

Where C_{m2} is the permeate boundary layer concentration (kg/m^3)

2.4. Membrane subsystem

The pressure balances are given by:

$$P_f = P_r + P_p \quad (5)$$

Where P_f is the feed pressure, P_r is the rejection pressure and P_p is the permeate pressure, all of them in Pascal (Pa).
The set of equations that define the membrane subsystem are given by:

2.4.1. Water transport equations

Permeate water flow is defined by (6) (Senthilmurugan, 2010)

$$F_p = AK_w(\Delta P - \sigma\Delta\pi) \quad (6)$$

Where A is the transfer area of the membrane (m^2), K_w is the water permeability coefficient (kg/m^2sPa), σ is osmotic pressure reflection coefficient.
The hydraulic pressure drop ΔP is given by (7)

$$\Delta P = 0.5(P_f + P_r) - P_p \quad (7)$$

The osmotic pressure drop $\Delta\pi$ is given by (8)

$$\Delta\pi = R_g T(C_{m1} - C_p)/M_m \quad (8)$$

Where R_g is the universal ideal gas constant ($m^3Pa/molK$), T is the feed solution temperature ($°K$), C_{m1} is the rejection boundary layer concentration (kg/m^3), M_m is the solute molar mass (kg/mol).

2.4.2. Salt transport equations

Permeate salt flow is defined by (9) (Dessouky, 2002)

$$F_s = K_s A(C_{m1} - C_p) \quad (9)$$

Where F_s is permeate salt flow (kg/s), K_s is the salt permeability coefficient (m/s).

Another expression for F_s is defined by (10)

$$F_s = F_p C_p \quad (10)$$

Expression for rejection boundary layer concentration of the membrane is given by equation (11) (Dessouky, 2002).

$$C_{m1} = \frac{F_f C_f + (F_f - F_p)C_r}{2F_f - F_p} \quad (11)$$

Another expression for concentration C_{m1} is also given by:

$$C_{m1} = C_{m2}\left[1 + \frac{K_w}{K_s}(\Delta P - \sigma\Delta\pi)\right] \quad (12)$$

Where the expression for permeate boundary layer concentration of the membrane can be deduced as the expression given in (13)

$$C_{m2} = \frac{C_f}{\left[1 + \frac{K_w}{K_s}(\Delta P - \sigma\Delta\pi)\left(1 - \frac{F_p}{2F_f}\right)\right]} \quad (13)$$

The model can be simplified by a assuming $C_{m2} = C_p$ and $F_{m2} = F_p$.This assumption will be considered in this work.

2.5. Additional equations for RO module

There is another way to get an expression for permeate concentration (C_p) and is given by equation in function of rejection factor (14)

$$C_p = C_{m1}(1 - R) \quad (14)$$

Expression for rejection factor is defined by (15)

$$R = \frac{(1 - F)\sigma}{1 - \sigma F} \quad (15)$$

Where the flow parameter F is given by the exponential relation in (16) (Senthilmurugan, 2010)

$$F = e^{\left[\frac{-J_v(1-\sigma)}{K_s}\right]} \quad (16)$$

Similarly another expression for obtaining permeate flow (F_p) in terms of rejection factor is given by (17)

$$F_p = \frac{K_s A R}{(1 - R)} \quad (17)$$

During filtration, accumulation of solutes occurs on the surface of the membrane (Rejection boundary layer) this accumulation produces a layer of concentration which can be determined by the model of concentration polarization (Figure 3) (Ahmed, 2010)

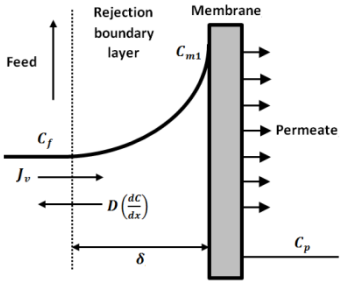


Figure 3. Concentration polarization model.

Fick's law is used to make the flow balance around the rejection boundary layer and is defined by (18)

$$J_v C_p = J_v C - D\ (dC/dx) \quad (18)$$

Expression for concentration polarization is obtained from (18) and is given by (19) (Khalaf, 2008)

$$\phi = \frac{C_{m1} - C_p}{C_f - C_p} = e^{\left(\frac{J_v}{D}\right)(\delta)} \quad (19)$$

Where ϕ is the concentration polarization factor, δ is the boundary layer thickness (m), J_v is the permeate flow velocity (m/s), D is the solution diffusion coefficient and is defined by the expression (20) (Jiang, 2014)

$$D = 6.725(10^{-6})e^{0.1546(10^{-3})C_f - \frac{2513}{273.15+T}} \quad (20)$$

Where C_f is the feed concentration, T is the feed solution temperature.

2.6. Relation between pH and conductivity

Expression that relates pH influence in permeate conductivity was taken from the experimental analysis made by Alatiqi (1989) and is given by equation 21.

$$Cd_{pH} = -0.03626(pH_f - pH_i) \quad (21)$$

Where Cd_{pH} is conductivity due to pH change, pH_f is the final value of pH, pH_i is the initial value of pH. Expression 21 shows that a positive change in pH produces a reduction of the final permeate concentration and conversely.

The final permeate concentration C_{ps} consists in the addition of concentration due to the feed pressure and the concentration due pH changes (Eq. 22).

$$C_{ps} = C_p + C_{pH} \quad (22)$$

In practice, salt content is obtained by measuring conductivity ($\mu S/cm$) instead of concentration (kg/m^3). Kohlrausch equation relates conductivity (Cd) and concentration (C) and is given by Eq. 23

$$Cd = \left[126.45 - 0.3692523\sqrt{0.001C}\right](C/58440) \quad (23)$$

2.7. Plant simulation Results

Figure 4 shows the dynamic of flows and concentrations. The simulation consider a constant value of feed pressure, where we note that flows have faster dynamic than concentrations. We can also see that brine concentration is greater than feed concentration and much higher than permeate concentration as it was expected. Boundary layer concentration is greater than feed concentration as we saw in Figure 3.

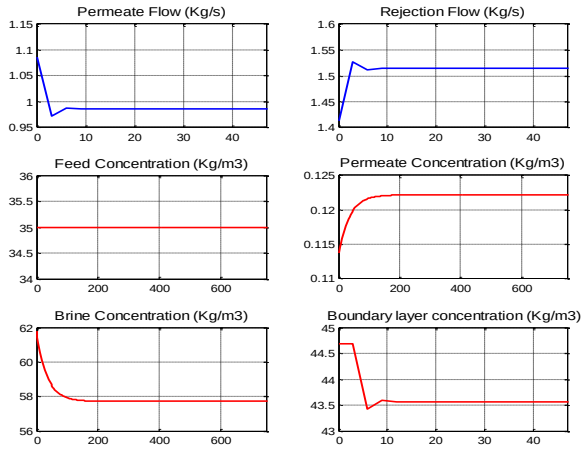


Figure 4. Plant Simulation Results.

3. FAULT DETECTION AND DIAGNOSIS SYSTEM DESIGN

3.1. Set of constraints

A set of 15 constraints that composes the structural model was obtained from the mathematical model and is shown in Table 1.

Table 1. Set of Constraints

R	EXPRESSION	R	EXPRESSION
r_1	$\frac{dC_r}{dt} = \frac{1}{m_r}[F_f(C_f - C_r) - F_p(C_p - C_r)]$	r_9	$F_p = \frac{F_s}{C_p}$
r_2	$\Delta P = 0.5(P_f + P_r) - P_p$	r_{10}	$F = e^{\left[\frac{-J_v(1-\sigma)}{K_s}\right]}$
r_3	$\Delta\pi = \frac{R_g T_f (C_{m1} - C_p)}{M_m}$	r_{11}	$R = \frac{(1-F)\sigma}{1-\sigma F}$
r_4	$J_v = K_w(\Delta P - \sigma\Delta\pi)$	r_{12}	$C_p = C_{m1}(1-R)$
r_5	$F_p = AJ_v$	r_{13}	$C_{ps} = C_p + C_{pH}$
r_6	$C_p = \frac{C_f}{\left[1 + \frac{K_w}{K_s}(\Delta P - \sigma\Delta\pi)\left(1 - \frac{F_p}{2F_f}\right)\right]}$	r_{14}	$C_{pH} = -0.03626(pH_f - pH_i)$
r_7	$C_{m1} = C_p\left[1 + \frac{K_w}{K_s}(\Delta P - \sigma\Delta\pi)\right]$	r_{15}	$C_{m1} = \frac{F_f C_f + (F_f - F_p)C_r}{2F_f - F_p}$
r_8	$F_s = K_s A(C_{m1} - C_p)$		

3.2. Set of faults

A suitable set of faults f to describe the state of the most important faults in main elements of the reverse osmosis module is defined. Fault signature vector (FSV) is given by the following expression:

$$f = (f_1, f_2, f_3, f_4, f_5, f_6, f_7)$$

Where: f_1 is the conductivity sensor fault, f_2 is the flow sensor fault , f_3 is high pressure pump fault , f_4 is the temperature sensor fault, f_5 is the membrane scaling fault, f_6 is the membrane weathering fault , f_7 is the acid pump fault (pH pump).

3.3. Incidence matrix

An incidence matrix is an arrangement that establish a relation between the known, unknown variables, and constraints. In Table 3 the incidence matrix arrangement is shown (Pérez, 2014).

3.4. Ranking Algorithm

The Ranking Algorithm is a matching algorithm that allow us to calculate the rank of each constraint, that is to say, the order in which the restrictions should be solved to find all unknown variables from the known ones. Table 2 shows the Ranking Algorithm (Blanke, 2006)

Table 2. Ranking Algorithm

Ranking Algorithm
1. Mark all known variables. Make $i = 0$
2. Find all constraints in the current table with exactly one unmarked variable. Associate rank i with these constraints and mark these constraints as well as the corresponding variable.
3. Set $i = i + 1$
4. If there are unmarked constraints whose variables are all marked, associate them with rank i, mark them and connect them with the pseudo-variable ZERO
5. If there are unmarked variables or constraints, continue with Step 2

3.5. Analytical Redundancy Relations

Constraints with the highest ranks represent the set of residual equations. As a result of application of the Ranking algorithm to the structural model we find a set of 4 residual equations that is shown highlighted in Table 3, we note that these residuals are according with the highest ranks (column R in Table 3).

Table 3. Incidence matrix

	KNOWN VARIABLES					UNKNOWN VARIABLES									
	P_f	pH	P_p	C_{ps}	T_f	C_p	C_r	ΔP	$\Delta\pi$	J_v	C_{m1}	F_s	F	R	C_{pH}	R	Z
r_1			x			x	x									2	
r_2	x							x								0	
r_3					x	x			x		x					2	
r_4								x	x	x						1	
r_5			x							x						0	
r_6			x			x		x	x							2	z
r_7						x		x	x		x					2	
r_8						x					x	x				3	z
r_9			x			x						x				2	
r_{10}										x						1	
r_{11}													x	x		2	
r_{12}						x					x			x		3	z
r_{13}				x		x									x	1	
r_{14}		x													x	0	
r_{15}			x				x				x					3	z

To get the set of analytical redundancy relations (ARRs), the set of residuals must be matched to zero (Eq. 24)

$$\begin{aligned}
&\text{ARR1: } C_p - \frac{C_f}{\left[1 + \frac{K_w}{K_s}(\Delta P - \sigma\Delta\pi)\left(1 - \frac{F_p}{2F_f}\right)\right]} = 0 \\
&\text{ARR2: } F_s - K_s A(C_{m1} - C_p) = 0 \\
&\text{ARR3: } R - \left(1 - \frac{C_p}{C_{m1}}\right) = 0 \\
&\text{ARR4: } C_{m1} - \frac{F_f C_f + (F_f - F_p)C_r}{2F_f - F_p} = 0
\end{aligned} \tag{24}$$

It was also obtained a set of 6 residuals from expressions for concentration polarization and rejection (Eq. 25). All the ARRs must be based on the reconstructed signals from the known variables.

$$
\begin{aligned}
&\text{ARR5:} = R_{r2} - \frac{C_{m1}r3 - C_{pi}}{C_{m1}r3} = 0\\
&\text{ARR6:} = C_{ps1} - C_{ps} = 0\\
&\text{ARR7:} = J_{vr3} - J_{vr1} = 0\\
&\text{ARR8:} = \frac{C_{m1}r4}{\left(1 + \left(\frac{K_w}{K_s}\right)D_{pres}r1\right)} - \frac{C_f}{\left[1 + \frac{K_w}{K_s}(D_{pres}r1)\left(1 - \frac{F_{ps}}{2F_f}\right)\right]} = 0\\
&\text{ARR9:} = \frac{C_{m1}r5}{\left(1 + \left(\frac{K_w}{K_s}\right)D_{pres}r1\right)} - \frac{C_f}{\left[1 + \frac{K_w}{K_s}(D_{pres}r1)\left(1 - \frac{F_{ps}}{2F_f}\right)\right]} = 0\\
&\text{ARR10:} = \phi_1 - \phi_{r1} = 0
\end{aligned}
\tag{25}
$$

The set of reconstructed signals from known variables is shown in Table 4; in this set we can see that all known variables are bolded.

Table 4. Reconstructed variables

$C_{m1}r3 = C_{pi} * \left(1 + \frac{K_w}{K_s}D_{pres}r1\right)$	$\Delta\pi_{r1} = \frac{(R_g \boldsymbol{T_s} C_{pi} K_w (\Delta P_r))}{M_m K_s + R_g \boldsymbol{T_s} C_{pi} K_w \sigma}$
$C_{pi} = \boldsymbol{C_{ps}} + 0.03626(\boldsymbol{pH_f} - \boldsymbol{pH_i})$	$C_{ps1} = C_{pr} + C_{pH}$
$D_{pres}r1 = \frac{J_{vr1}}{K_w}$	$C_{pH} = -0.03626(\boldsymbol{pH_f} - \boldsymbol{pH_i})$
$J_{vr1} = \frac{\boldsymbol{F_{ps}}}{A}$	$C_{pr} = \frac{\phi_{r1}(1 - R_{r1})C_f}{\phi_{r1} + R_{r1}(1 - \phi_{r1})}$
$C_{m1}r4 = C_{pi} + \phi_{r1}(C_f - C_{pi})$	$R_{r1} = \frac{(1 - F_{r1})\sigma}{1 - \sigma F_{r1}}$
$C_{m1}r5 = \phi_{r1}C_f$	$F_{r1} = e^{\left(\frac{-J_{vr1}(1-\sigma)}{K_s}\right)}$
$\phi_{r1} = e^{J_{vr1}/k}$	$C_{m_1}r1 = \frac{(F_f C_f + (F_f - \boldsymbol{F_{ps}}) * C_{ri})}{(2F_f - \boldsymbol{F_{ps}})}$
$R_{r2} = \frac{(1 - F_{r2})\sigma}{1 - \sigma F_{r2}}$	$C_{ri} = (F_f C_f - \boldsymbol{F_{ps}} C_{pi})/(F_f - \boldsymbol{F_{ps}})$
$F_{r2} = e^{\left(\frac{-J_{vr2}(1-\sigma)}{K_s}\right)}$	$J_{vr3=} k * \ln\left(\frac{C_{m1}r1}{C_f}\right)$
$J_{vr2} = K_w * (D_{pres}\, r2)$	$\phi_1 = (C_{m1}r1 - C_{pi})/(C_f - C_{pi})$
$D_{pres}\, r2 = \Delta P_r - \sigma\Delta\pi_{r1}$	
$\Delta P_r = 0.5(\boldsymbol{P_f} + P_r) - P_p$	

Two PID control loops were designed for permeate flow and permeate concentration output signals, because these are the variables of interest for quantity and quality of permeate water respectively (Alatiqi, 1989), in this way the 2 error signals of PID controllers were used to improve the isolability task. We must note that variables that make the error signals are also known.

$$
\begin{aligned}
ERR1 &= \boldsymbol{F_p ref} - \boldsymbol{F_{ps}}\\
ERR2 &= \boldsymbol{C_p ref} - \boldsymbol{C_{ps}}
\end{aligned}
\tag{26}
$$

Thus a total of 12 signals were used for fault detection and isolation analysis.

4. SIMULATION RESULTS

Results of detectability and isolability analysis are shown in the following Figures.

4.1. Normal Mode Behavior

Figure 5 shows that 12 signals designed remain at zero or very close to zero in case of normal mode behavior of the plant, which indicates no presence of faults affecting the system.

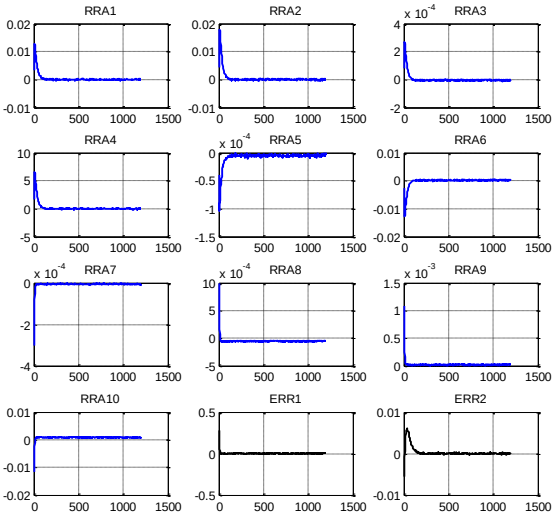


Figure 5. Normal mode behavior

For simulation, in all cases faults will be pulse type, that is to say those that occur only for a specified time interval.

4.2. Conductivity sensor fault

Figure 6 shows that this fault was detected by the RRA1, RRA2, RRA3, RRA4, RRA5, RRA6, RRA7, RRA8 and ERR2. In this fault, an additive offset of 50% of the variable occurred during the time interval of 600 to 650 seconds.

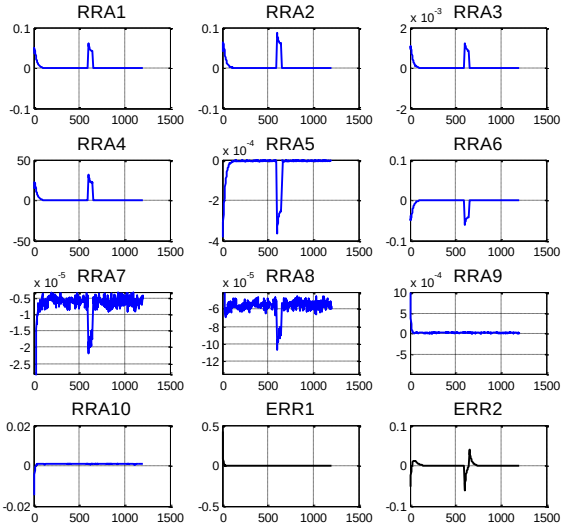


Figure 6. Conductivity sensor fault

4.3. Scaling/fouling membrane Fault

Scaling means the deposition of particles on a membrane, causing it to plug, it depends on numerous factors like pH, temperature and the presence of other ions. Scaling fault can be simulated by reducing the cross-flow velocity J_v; this fault was simulated during the time interval of 400 to 450 seconds. Figure 7 shows that scaling membrane fault was detected by RRA3, RRA4, RRA5, RRA6, RRA7, RRA8, RRA9 and RRA10.

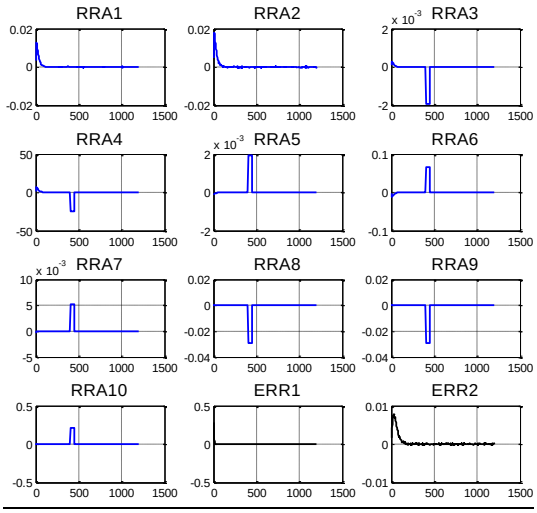


Figure 7. Scaling Fault

4.9. Summary Results

Another five faults were simulated during different time intervals. From figures 6 and 7 we see that each simulated fault corresponds to a pattern of RRAs for which is detected. A summary table of results is necessary to evaluate detectability and isolability of faults. Table 5 presents the summary results.

Table 5. Summary results

FAULT	RRA1	RRA2	RRA3	RRA4	RRA5	RRA6	RRA7	RRA8	RRA9	RRA10	ERR1	ERR2
High pressure pump	x	x	x	x	x	x	x	x	x	x	x	x
pH pump	x	x	x	x	x		x	x				x
Flow sensor	x	x	x	x	x	x	x	x	x	x	x	
Conductivity sensor	x	x	x	x	x	x	x	x				x
Temperature sensor	x				x							
Membrane weathering			x			x						
Membrane scaling/fouling			x	x	x	x	x	x	x	x		

From Table 5 we can see that all faults considered in the design corresponds to different patterns of RRAs, the patterns found not repeat each other, thus we can conclude that all elements of FSV are detectable and isolable.

5. CONCLUSIONS

A model based on analytical equations of the RO desalination plant was proposed, the model validation gave consistent and similar results to those obtained in the literature. Following this model was presented as a structural model and then common faults that may occur during the actual operation of a reverse osmosis were defined. A set of seven faults; three in sensors, two in actuators and two in internal parameters that can lead to malfunctioning of the RO membrane were considered.

Detectability analysis of seven faults in the design resulted in that all faults were detectable by the set of RRA designed. Furthermore, by analyzing isolability it was determined that all faults were isolable.

As a future work we propose the use of nonlinear observers of certain variables of interest, which will allow for obtaining additional redundancy relations to detect and isolate other faults of interest.

ACKNOWLEDGMENT

The authors acknowledge the support of the National Innovation Program for Competitiveness and Productivity (Innovate Peru) entity that financed the-FINCyT-IA-2013 207 project, under which this article was developed.

REFERENCES

Ahmed, F. (2013) *Modified Spiegler-Kedem Model to Predict the Rejection and Flux of Nanofiltration Processes at High NaCl Concentrations***.** University of Ottawa. 3,59-66.

Alatiqi, I.M., Ghabris, A.H., Ebrahim, S. (1989). System identification and control of reverse osmosis desalination. *Desalination Journal*, 75 (1989) 119-140.

Blanke, M., Kinnaert, M., Lunze, J., Staroswiecki, M. (2006) *Diagnosis and Fault-Tolerant Control*. Springer. 109-175.

Garcia D. (2011) Monitoring and Fault Detection in a Reverse Osmosis Plant using Principal Component Analysis. *IEEE Conference on Decision and Control and European Control Conference.*1-6.

Gambier A.; Miksch T.; Badreddin E. (2009). A reverse osmosis Laboratory Plant for Experimenting with Fault-Tolerant Control. *American Control Conference Hyatt Regency Riverfront, St. Louis, MO, USA. Automatica*, 27, 1039-1042

Dessouky and Ettouney (2002) *Fundamentals of Water Desalination***.** Department of Chemical Engineering College of Engineering and Petroleum Kuwait University. Elsevier, 410-435.

Isermann R. (2005) *Fault-Diagnosis Systems. An introduction from Fault Detection to Fault Tolerance***.** Springer. 61-82.

Jiang A; Ding Q, Wang J. (2014) Mathematical modeling and simulation of SWRO Process based on simultaneous method. *Hindawi Publishing Corporation, Journal of Applied Mathematics* Volume 2014, Article ID 908569, 11 pages.

Khalaf T. (2008). Estimation of concentration polarization using the Combined Film Theory/Spiegler Kedem Model and Empirical Correlation. *The 1st Regional Conference of Eng. Sci. NUCEJ Spatial ISSUE* vol 11, No2, pp 322-328.

McFall Ch, Panagiotis D. (2007) Fault-Tolerant Control of a reverse osmosis desalination process *8th International IFAC Symposium on Dynamics and Control of Process Systems, México.*, 161 -166.

Palacin L.G. (2011) *New dynamic of reverse osmosis plants with fault simulation.*. Department of Systems Engineering and Automatic Control University of Valladolid, Spain.,127-132.

Pérez G.; Sotomayor J. (2014). Análisis de Redundancia para diagnóstico de fallas de una planta desalinizadora de agua de mar**.** *Congreso Latinoamericano de Control Automático, México.*, 1-6.

Senthilmurugan, S., Ahluwalia A., Gupta S. (2005) Modeling of a spiral wound module and estimation of model parameters using numerical techniques. *Desalination*. 269-286.

Modeling of a Variable-BVR Rotary Valve Free Piston Expander/Compressor

Sergei Gusev [*] **Andres Hernandez** [**] **Davide Ziviani** [*]
Martijn van den Broek [*]

[*] *Ghent University, Department of Flow, Heat and Combustion Mechanics, Graaf Karel de Goedelaan 5, Kortrijk, Belgium (e-mail: Sergei.Gusev@UGent.be, Davide.Ziviani@UGent.be, Martijn.vandenBroek@UGent.be).*
[**] *Ghent University, Department of Electrical Energy, Systems and Automation, Technologiepark 914 B-9052 Zwijnaarde, Belgium (e-mail: Andres.Hernandez@UGent.be)*

Abstract: The concept of a free-piston expansion/compression unit with a variable Built-in Volume Ratio (BVR) is proposed. This device has no crankshaft mechanism which provides a possibility to optimize the expansion process free of mechanical limitations. An additional degree of freedom is used, namely the rotation to control the in- and the outlet ports timing. Further, the operation in the expander mode will be described.
In most of the existing linear expanders/compressors, bouncing chambers or devices are used to reverse the piston movement at extreme positions. This approach is characterized by relatively high energy losses due to irreversibility of such a process. As an alternative, a fully controlled movement of the piston is proposed. This paper is focused on the control algorithm based on rules, which have been obtained and based on the insight in the system. Including the rotation timing, resulting in an optimal expansion process with an outlet pressure matching with the required one.

1. INTRODUCTION

Steady state operating volumetric compressors and expanders are widely studied machines. The challenge is to use such devices under strongly varying in- and outlet pressure and temperature conditions, especially at relatively high pressure ratios, which is frequently the case in waste heat recovery for non-stationary applications. For instance, the heat recovery from truck flue gases by means of an Organic Rankine Cycle (ORC) requires an expansion device with adjustable Built-in Volume Ratio (BVR). Fixed BVR machines cannot follow variations in the evaporating pressure caused by changes in both the flow rate and the temperature of the exhaust gases, resulting in a non-optimal operation of the system.

To overcome the lack of commercially available expanders, a novel variable-BVR expander has been proposed by the authors as an alternative to existing solutions, mainly conversion of compressors to expanders (Imran et al., 2016). Based on the results obtained during the tests performed on an ORC laboratory setup, the proposed idea intends to be a solution for challenges in expander technology established during these experiments.

The long-term objective is to develop a commercial unit meeting the requirements for a micro-scale ORC-system: inexpensive, scalable, flexible and efficient. The short-term objective, which has to be reached in the frame of the current project, is to develop a setup meant to validate the behavior of the main components and to demonstrate the feasibility to synchronize the linear and the rotational movement of a piston in order to achieve any required volume ratio. A proposal for a patent has been filed.

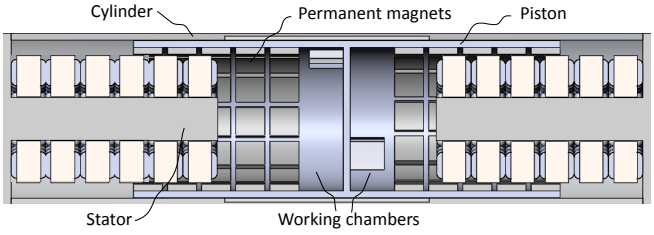


Fig. 1. Expander cutaway view

In this paper, the first approach towards the design of a control strategy for the proposed machine, is reported. The mathematical modeling and the numerical results, which led to the consideration of a controller design, are presented and discussed.

2. NOVEL VARIABLE-BVR EXPANDER

The mechanical design and the operation principle of the test prototype with an embedded linear generator (Fig. 1) is described in previous publications (Gusev et al., 2016). The piston rotates while moving, closing and opening the

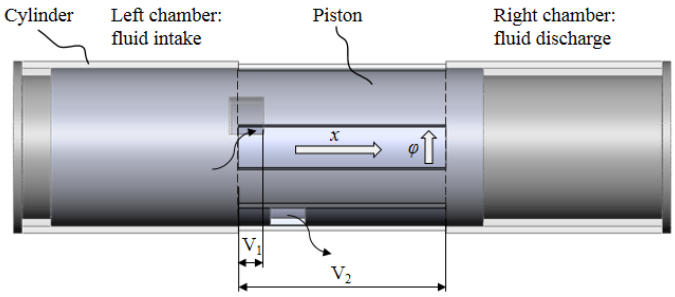


Fig. 2. Variable volume ratio

intake port at the required moment defined by the model. The built-in volumetric ratio can be expressed as:

$$BVR = \frac{V_2}{V_1} = \frac{1}{\chi}, \tag{1}$$

where V_1 is the volume at the moment of closing of the inlet port and V_2 is the volume of the working chamber after the expansion. The piston relative displacement χ corresponds with the volume V_1.

The $BVR = 10$ at $\chi = 0.1$ is shown in Fig. 2 when the leading-bottom corner of the piston skirt opening leaves the inlet opening of the cylinder. Since the rotation and translation of the piston have to be synchronized, the model is focused on the accurate definition and control of the piston position in both dimensions by means of rules, dictated by the thermodynamic model. The control coefficients, obtained from this model trigger the control sequence accelerating or decelerating the piston. The feedback signal is provided by the translation and the rotation encoders.

Since at the beginning of the prototype design, leakages are unavoidable and the discharge of expensive or toxic working fluids into the atmosphere is unacceptable, air is chosen as a working fluid.

In order to avoid complications with an embedded design, it was decided to separate translation and rotation movement and to use standard components. Several industrial linear motors operating in a brake mode are compared, its characteristics provided by the manufacturers are used as input for the model.

The chosen configuration is based on a standard linear motor consisting of a moving primary coil section and a secondary magnetic section (Fig. 2). In the adjusted design, two moving magnetic secondary sections are placed back-to-back on a linear guide system and two primary coil sections are fastened outside (Fig. 4). Such a configuration allows to equalize electromagnetic attraction forces and to reduce drastically the friction losses in the linear guiding system. Moreover, the maximal static force is twice the size, compared to a single-coil design for almost the same setup size and it is 6.2 kN in total. The inlet pressure applied to the piston can vary from 0.6 to 1.6 MPa. Other major setup parameters are summarized in Table 1.

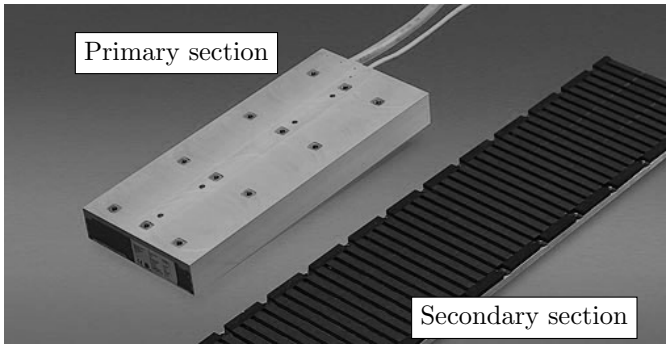


Fig. 3. Standard linear motor

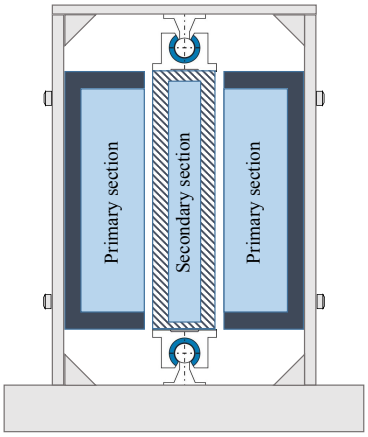


Fig. 4. Adjusted design

Table 1. Main parameters of the designed test setup

Bore (m)	Stroke (m)	Frequency (Hz)	Moving mass (kg)
0.08	0.23 - 0.31	5 - 10	30

3. THE EQUATION OF MOTION

3.1 Translation

The model of the free-piston expander is based on the same approach as (Mikalsen and Roskilly, 2008). The dynamics of the piston translation is dictated by Newton's second law:

$$F_{p,cyl} - F_{p,dis} - F_{fr} - F_{el} = m_p \frac{d^2x}{dt^2}, \tag{2}$$

where $F_{p,cyl}$ and $F_{p,dis}$ are gas forces in opposing working chambers, the friction force F_{fr}, the electromagnetic force F_{el} and the piston mass m_p.

The gas forces are applied to the same central element and defined by the pressure profile of an expansion process (Fig. 5) which is simulated using the expander hybrid gray box model designed in previous studies. The CoolProp library connected to Python is used (Bell et al., 2014) in order to calculate thermodynamical parameters.

The working fluid entering the expander is cooled down since the expander wall temperature is typically between the in- and the outlet temperature of the working medium. At the end of the expansion, the heat flux reverses. This heat transfer from or to the expander walls is taken

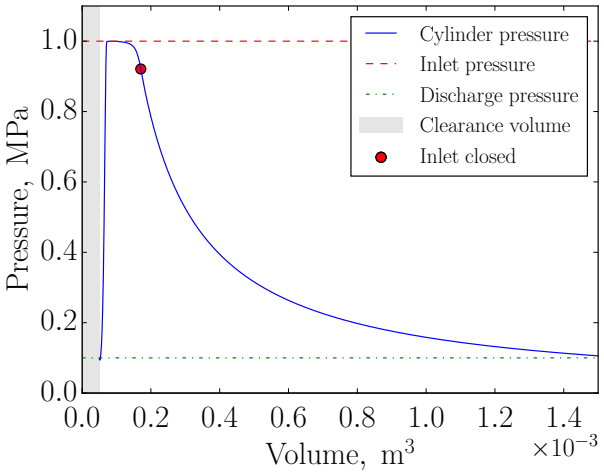


Fig. 5. pV-diagram

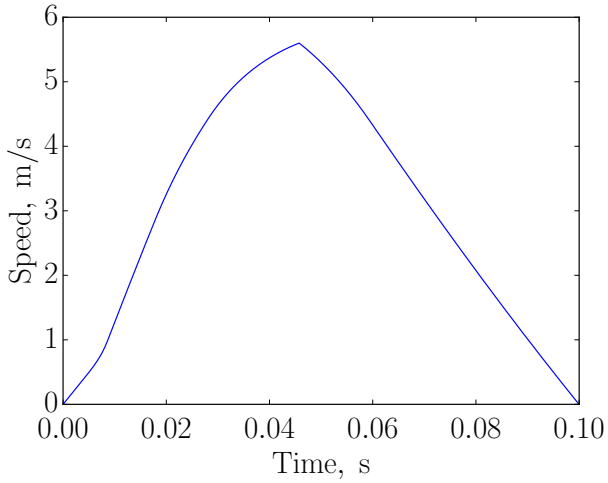


Fig. 6. Linear velocity profile

into account. The friction force F_{fr} is dependent on the speed of the piston. The correlation for dynamic friction behaviors of pneumatic cylinders obtained in (Tran and Yanada, 2013) is used. The heat produced by the friction is also incorporated into the model.

The translation speed changes under the applied electromagnetic force F_{el}, which is constant and acting alternately accelerating and decelerating the piston, in order to stop the piston at its extreme left and right positions, so all kinetic energy is absorbed and transformed into electricity. The approach is similar to (Petrichenko et al., 2015). The resulting equation can be written as follows:

$$F_{el}(t) = \frac{\pi D^2}{4}\left(p_{cyl}(t) - p_{dis}\right) - F_{fr}\left(\frac{dx}{dt}\right) - m_p\frac{d^2x}{dt^2}\,, \quad (3)$$

where D - is the piston diameter, p_{cyl} and p_{dis} are pressures in the working chamber and the discharge port respectively. The Equation 3 is used to control the drive of the linear motor, the position sensor is used for feedback of the piston position. The rotation is synchronized with the translation in order to obtain the required discharge pressure at the end of the expansion (Fig. 6).

3.2 Rotation

A servo motor attached to the expander on the opposite side of the linear generator side rotates the piston with an average frequency of 1/2 the frequency of the translation since there are two inlet and two outlet ports in each working chamber. The general equation for torque balance at the motor shaft can be written for the prototype as follows:

$$T_{el} = T_j + T_{fric}\,, \quad (4)$$

where T_{el} electromagnetic torque, T_j - inertia torque and T_{fric} - friction torque. The moment of inertia defining the inertia torque is the sum of moments of inertia of all components in the rotation train: the servo motor, the piston and shafts. These can be calculated using the sizes of the components used. These components rotate about the same axis and are have cylindric shape. The moment of inertia of a hollow cylinder is:

$$J = \frac{1}{8}m(D_{out}^2 + D_{in}^2)\,, \quad (5)$$

where m is the mass of the rotating component, D_{out} - the outer diameter, D_{in} - the inner one which is equal to zero for shafts and the servo motor rotor. The shaft mass and the diameter is relatively small and therefore can be neglected.

For the calculations of the friction torque, the same correlation as for the linear motion can be used (Tran and Yanada, 2013) with adjustments for the rotational motion by substitution of the peripheral speed of the piston instead of the linear one.

The resulting equation, used for the servo motor control can be written as follows:

$$T_{el}(t) = \frac{d^2\varphi}{dt^2}\left[\frac{1}{8}\left(m_{rot}D_{rot}^2 + m_{cyl}\left(D_{cyl,out}^2 + D_{cyl,in}^2\right)\right) + \right.$$
$$\left. + \frac{D_{cyl}}{2}F_{fric}\left(\frac{\varphi(t)}{dt}\right)\right], \quad (6)$$

where D_{rot} - is the servo motor rotor diameter, $D_{cyl,out}$ and $D_{cyl,in}$ are the outer and the inner diameter of the piston.

There is no influence op working pressures on the rotation since the in- and the outlet ports are placed axisymmetrically and therefore the pressure induced forces are compensated.

4. CONTROL STRATEGY

The control strategy here proposed corresponds to an algorithm designed based on the mechanical insight, where a set of rules are proposed to achieve the desired performance.

4.1 Intake and discharge.

The working medium enters the expander through a rectangular port formed by the openings in the housing and the skirts. The mass flow rate is dependent on the port area S, the pressure difference between the inlet pressure p_{su}

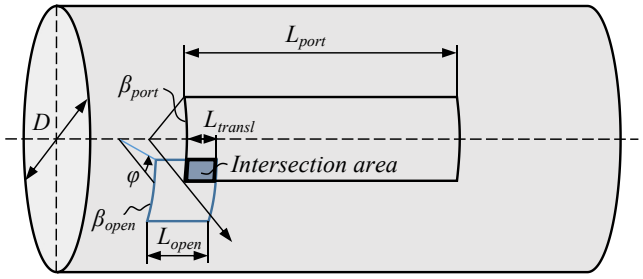


Fig. 7. Schematic of the port geometry.

Table 2. Inlet port and the skirt opening sizes

L_{open} (m)	L_{port} (m)	β_{open} (rad)	β_{port} (rad)
0.02	0.23	$\pi/12$	$\pi/12$

and the pressure in the cylinder p_{cyl}. The intersection area is changing according to the rotation and the translation of the piston (Fig. 7).

The position of the lower left corner of the cylinder wall opening is chosen as the reference point. The start of the intake process corresponds with the position of the piston when the lead-top corner of the skirt opening is at the reference point. A simplified algorithm defining the overlap of two rectangular openings is used to calculate the intersection area (Eq. 7):

$$\begin{aligned} S(t) = (&max(0, min(L_{transl}(t), L_{port}) \\ &- max(0, L_{transl}(t) - L_{open}))\times \\ &\times (max(0, min(\varphi(t), \beta_{port}) \\ &- max(0, \phi(t) - \beta_{open})) \cdot \frac{\pi D}{2} \cdot n\,, \end{aligned} \tag{7}$$

where n is the number of ports. There are two axisymmetric ports used in current configuration. The inlet port and the skirt opening sizes are shown in Table 2.

A similar model is applied to estimate the mass flow rate during the discharge process. The corresponding pressures p_{cyl} and p_{dis} are used.

4.2 Optimization criteria

Filling factor. The stroke length is defined in previous simulations and is kept constant. The piston movement profile under the applied forces defines the frequency of the machine, which must be maximized for a higher mass flow rate of the working medium. However, the higher the piston speed during the intake phase, the more difficult to maximize the inlet area for a higher filling factor. The definition of a filling factor is introduced by (Lemort et al., 2009) and means the measured flow rate divided by the displacement. Ideally, the density of the working fluid at the end of the intake is equal to the one at the inlet port. The actual density of the working fluid in the cylinder divided by the ideal one gives the indication of the intake efficiency.

$$\phi_{ff} = \frac{\rho_{cyl}}{\rho_{su}} \tag{8}$$

Table 3. Adjustable control parameters for different inlet pressures.

Pressure, MPa	$k1$	$k2$	$k3$	$k4$	$k5$	$k6$
1.0	0.41	1.05	0.426	0.242	0.15	0.006
0.6	0.75	1.09	1.0	0.242	0.15	0.007

A non-optimized velocity profile is shown on Fig. 6. As it can be seen on pV-diagram (Fig. 10), the inlet port closes relatively late causing a significant pressure drop of about 100 kPa at the end of the intake. The filling factor in this case is about 0.89.

After the optimization, the filling factor rises above the unity, which, beside the optimization, is caused by cooling down of the working fluid during the intake process due to a lower temperature of the expander. A normalized filling factor can be applied by using the actual temperature in the working chamber instead of the inlet temperature.

Intake efficiency. Another optimization criterion can be the intake efficiency expressed as a ratio of surface areas of the actual and ideal pV-diagrams from the opening of the inlet until it closes. After the rotation and translation adjustment, the values of 0.96 - 0.97 are obtained.

The linear motion and the rotation of the piston have to be synchronized in order to achieve the required BVR and a maximal inlet area during the intake. The maximal inlet area is theoretically achievable only if the motion of the piston is defined by square waves of both the rotation and the translation, which is impossible in practice due to a certain mass and the moment of inertia characterizing the piston. It is possible to approach such an ideal movement profile by reducing the piston acceleration while the inlet is open for more accurate timing control by rotation. The rotation has to be also adjusted.

Fig. 8 and Fig. 9 show the control algorithm for the piston linear movement and the rotation respectively. The adjustable parameters $k1$ - $k6$ allowing the pressure ratio of 1.0 and 0.6 MPa are shown in Table 3.

4.3 Translation control

Intake: $F_{em} = F_{nom} \cdot k1$ - the electromagnetic force is limited by the factor $k1$ during the inlet phase.

$p_{cyl} < p_{dis} \cdot k2$ - the piston is forced to move until the inlet port opens and the cylinder pressure starts to increase until a certain value defined by $k2$.

$F_{em} = -F_{del_p} + F_{fric}$ - the piston speed is kept constant as long as the inlet is open ($\varphi > \beta_{port} + \beta_{open}$). All forces are compensated by the electromagnetic one.

Acceleration: While the piston speed is lower then v_{max}, the nominal force (F_{nom}) is applied.

Brake: The electromagnetic force is adjusted so the piston reaches its extreme right position with $v = 0$ (Fig. 11). A PI-action can be applied at the end of the stroke in order to compensate a linear positioning error. It is important to avoid an overshoot since it means a mechanical impact of the piston on the stator.

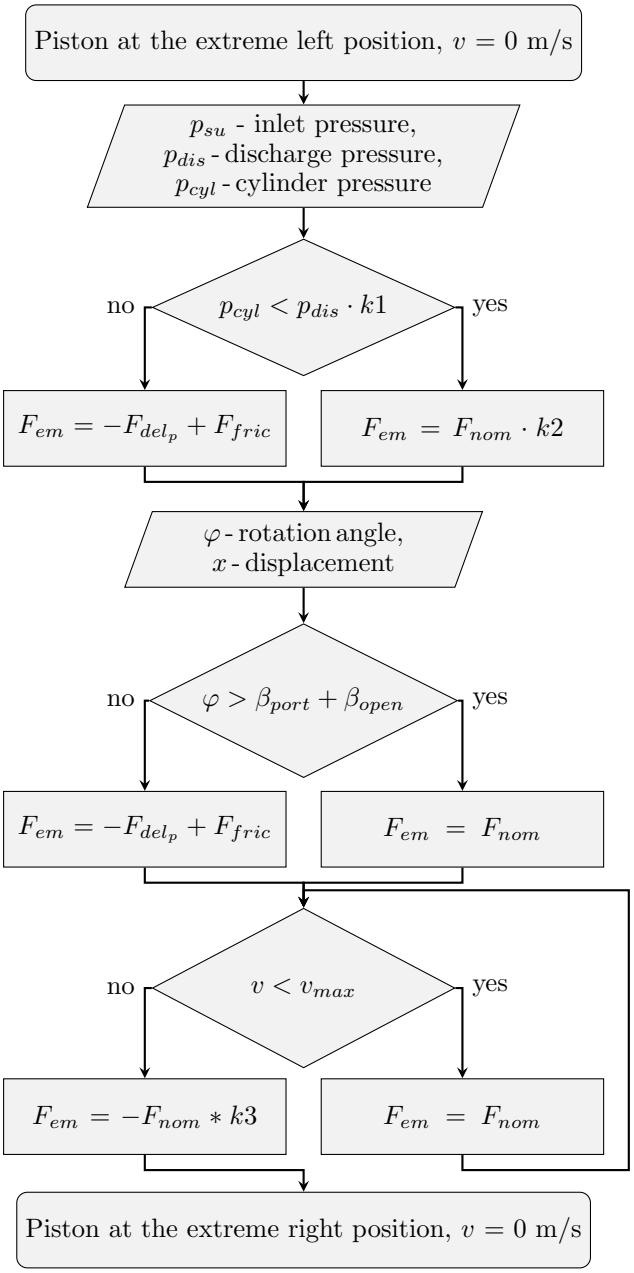


Fig. 8. Piston linear movement algorithm

4.4 Rotation control

Intake: While the piston starts to move away from its extreme left position, the rotation velocity is ω_0. By applying a decelerating torque of $-T_{nom}$ at the moment defined by $k4$, the rotation decreases, ideally down to zero ($k5 = 0$), when the skirt opening is aligned with the inlet opening. The resulting speed can be adjusted by the coefficient $k5$ if the rotation should not be completely stopped but just reduced to allow a higher piston response. This is an open loop control since a high positioning accuracy is not required.

The piston travels to the right without rotation. When the displacement reaches a certain value defined by $k6$, the rotation is accelerated with a torque of T_{nom} in order to close the inlet port when it is dictated by the thermodynamic model.

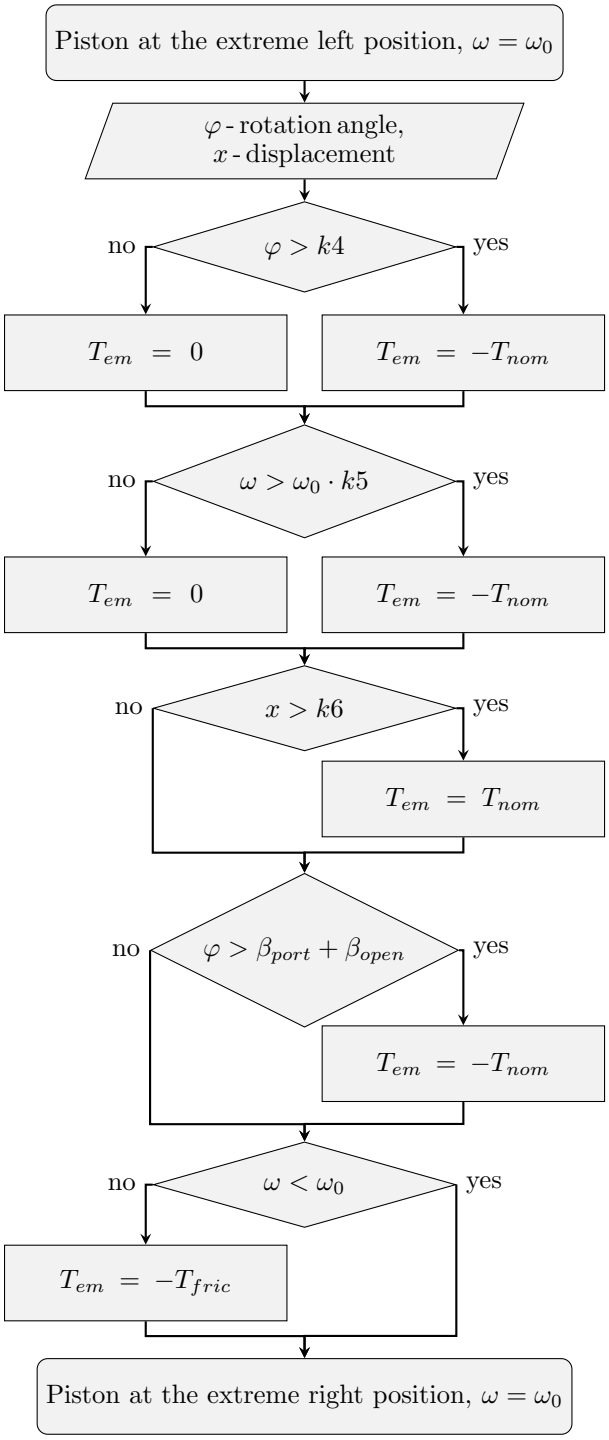


Fig. 9. Piston rotation algorithm

Deceleration After the inlet is closed ($\varphi > \beta_{port} + \beta_{open}$), the piston rotation speed needs to be reduced until it reaches ω_0, then the piston rotates with a constant speed. The torque applied from the servo motor is equal to the one caused by friction. At the end of the stroke, a PI-action can be applied for a better accuracy.

The resulting velocity profile vs. time is shown on Fig.12. It can be seen that the higher the inlet pressure, the faster the piston reaches its maximal translation speed, so a lower brake force is needed to decelerate it until the extreme right position.

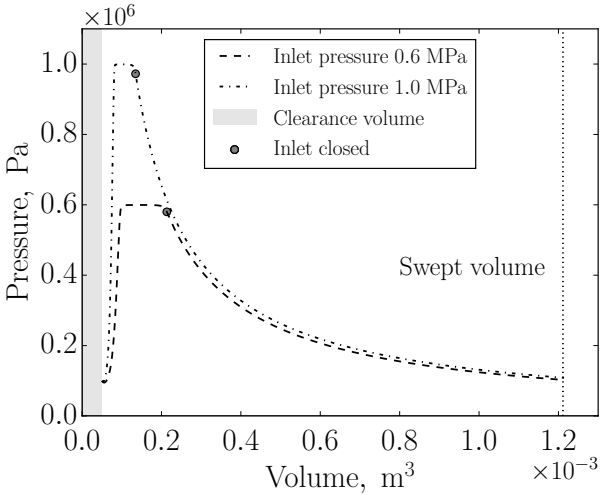


Fig. 10. Piston velocity vs. displacement

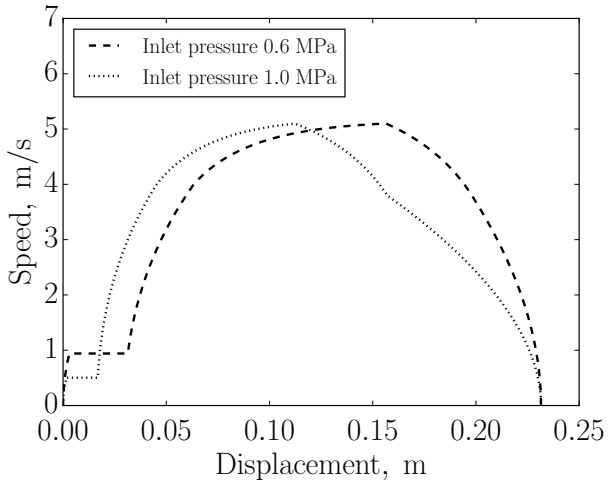


Fig. 11. Piston velocity vs. displacement

Two inlet pressures 1.0 MPa and 0.6 MPa are compared and the configuring factors are identified (Fig. 10).

By performing such simulations within the expected inlet pressure range with a certain step, a matrix of these parameters can be obtained and used in real time to adjust the piston movement under varying inlet pressure.

4.5 Parameters description

Variables: Inlet pressure

Constants: Inlet temperature, outlet pressure, expander geometry, zero piston velocity at the end of expansion phase.

Adjustable parameters: k1, k2, k3, k4, k5, k6.

Efficiency indicators: frequency, filling factor, intake efficiency, power output: to maximize.

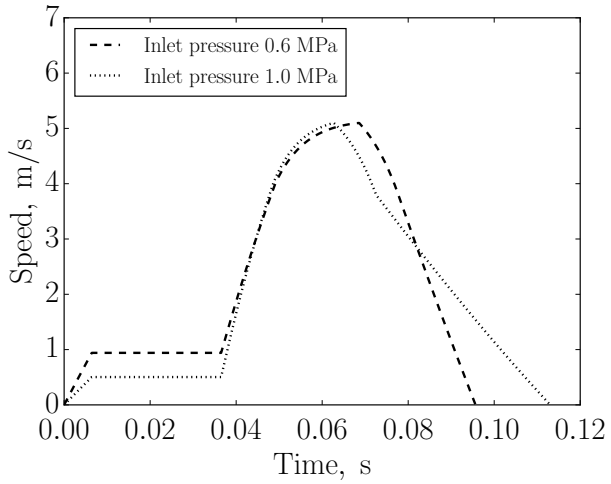


Fig. 12. Piston velocity vs. displacement

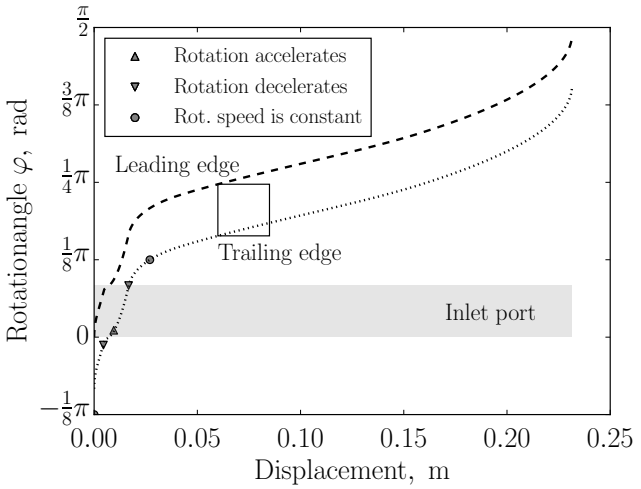


Fig. 13. Piston velocity vs. displacement

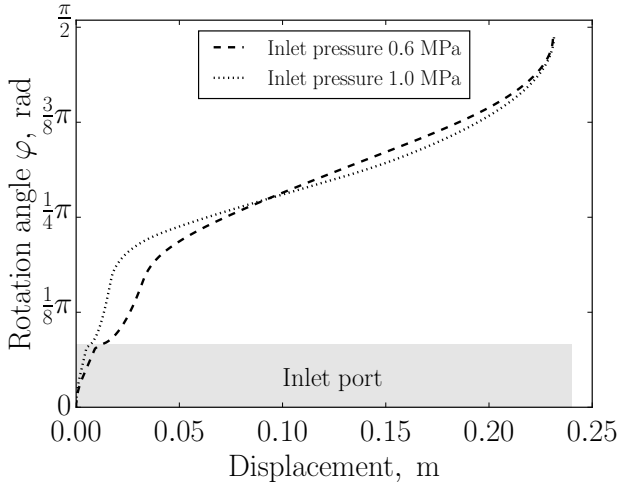


Fig. 14. Piston velocity vs. displacement

5. RESULTS AND DISCUSSIONS

The developed model is based on parameters of the selected linear motor and the servomotor and allows to synchronize the rotation with the linear movement of the expander. By decreasing the linear speed during the intake phase, the pressure loss at the inlet can be reduced. However, reduced translation speed means lower frequency and the mass flow rate through the expander, so the optimum has to be found (Fig. 12).

The rotation of the piston can be increased or decreased when necessary for a better fit of the inlet. A moderate rotating acceleration/deceleration is achievable in combination with the piston deceleration during the intake. The final deceleration must be adjusted so the leading edge of the skirt opening reaches the outlet port at the end of the stroke. In this case this angle is $\pi/2$ (Fig. 13). Two rotation/translation profiles for different inlet pressures are shown on Fig. 14

The shape of the inlet port needs to be adjusted to "follow" an optimal intake profile in order to keep the inlet intersection area around its maximum during the intake phase. Otherwise higher pressure ratios will require high dynamics from electromagnetic train or will cause relatively high intake losses.

The presented model is focused on the inlet control and therefore simplified by setting the outlet pressure as constant.

6. CONCLUSIONS

The dynamics of the system depend on the linear generator used. Industrial linear motors are characterized by large moving masses. It is possible to reduce the weight of the piston if magnets are directly attached to it. A high dynamics of the piston movement is necessary to increase the resulting frequency and the volumetric flow rate of the machine. A relatively high static force is required to keep the piston under control at its extreme positions. The use of a position encoder ensures the high accuracy of the inlet timing, which is crucial for the system efficiency.

The proposed system contains no bouncing devices such as gas- or mechanical springs, which are typically used in free piston machines. Instead, the piston movement is fully controlled, so its velocity becomes zero at both extreme positions. A higher system efficiency is expected since mechanical wear or thermodynamic irreversibilities are avoided.

REFERENCES

Bell, I.H., Wronski, J., Quoilin, S., and Lemort, V. (2014). Pure and pseudo-pure fluid thermophysical property evaluation and the open-source thermophysical property library coolprop. *Industrial & engineering chemistry research*, 53(6), 2498–2508.

Gusev, S., Ziviani, D., De Viaene, J., Derammelaere, S., and van den Broek, M. (2016). Modelling and preliminary design of a variable-bvr rotary valve expander with an integrated linear generator. In *Proceedings of the 17th International Refrigeration and Air Conditioning Conference at Purdue*.

Imran, M., Usman, M., Park, B.S., and Lee, D.H. (2016). Volumetric expanders for low-grade and waste heat recovery applications. *Renewable and Sustainable Energy Reviews*, 57, 1090–1109.

Lemort, V., Quoilin, S., Cuevas, C., and Lebrun, J. (2009). Testing and modeling a scroll expander integrated into an organic rankine cycle. *Applied Thermal Engineering.*, 29, 3094–3102.

Mikalsen, R. and Roskilly, A. (2008). The design and simulation of a two-stroke free-piston compression ignition engine for electrical power generation. *Applied Thermal Engineering*, 28(56), 589–600.

Petrichenko, D., Tatarnikov, A., and Papkin, I. (2015). Approach to electromagnetic control of the extreme positions of a piston in a free piston generator. *Modern Applied Science*, 9(1), 119–128.

Tran, X.B. and Yanada, H. (2013). Dynamic friction behaviors of pneumatic cylinders. *Intelligent Control and Automation*, 4(2).

Appendix A. NOMENCLATURE

A.1 Latin characters

D	diameter	(m)
F	force	(N)
L	length	(m)
m	mass	(kg)
n	number of ports	(-)
p	pressure	(Pa)
S	intersection area	(m^2)
t	time	(s)
V	volume	(m^3)
x	displacement	(m)

A.2 Greek characters

β	port angle	(rad)
Δ	difference	(-)
υ	speed	(m/s)
φ	angle of rotation	(rad)
ϕ	filling factor	(-)
χ	relative displacement	(-)

A.3 Subscript

cyl	cylinder
dis	discharge
el	electromagnetic
fr	friction
max	maximal
min	minimal
open	opening
port	port
rot	rotation
su	supply

Models for Planning and Supervisory Control for the feeding raw material in cement production [1]

Edgar CHACÓN R. [*,***] **Juan CARDILLO A.** [**]

[*] *Dpto. de Computación, Escuela de Ingeniería de Sistemas, Facultad de Ingeniería, Universidad de Los Andes, Mérida Venezuela, echacon@ula.ve*

[**] *Dpto. Sistemas de Control, Escuela de Ingeniería de Sistemas, Facultad de Ingeniería, Universidad de Los Andes, Mérida Venezuela, ijuan@ula.ve*

[***] *Unidad Académica de Ingeniería de Sistemas, Universidad Católica de Cuenca, Cuenca Ecuador*

Abstract:
The clinker production process is mainly affected by the quality of the raw material. The chemical composition required for the formation of clinker is giving by the mixing of different limestone sources, clay, and iron ore. The optimal combination may be determined in principle by linear programming techniques, but the variation of the quality of limestone mines is high, so it is necessary to adjust the mix online, by using transport mechanisms which are discrete and sharing other resources. The feed to the mill is divided into two processes : crushing and forming stacks (pre- homogenization) and the mixture preparation online in the mill for forming flour (homogenization).
The goal is to design mechanisms that allow the proper proportions of minerals to the furnace inlet through programming online resources for pre–homogenization : the limestone quarries, trucks, crusher and transport mechanism of the raw material to deposit. Supervisory control techniques are used, and models are constructed as discrete event systems to ensure that the mixture is as homogeneous as possible over time. A particular architecture (Holonic architecture), it is used in the solution; which allows an easy and effective implementation of an on-line supervisor. Supervisor uses the material existence and cost knowledge. A reference is made to an application that is a Discrete Event System model interpreter for the implementation of the supervisor.

Keywords: Discrete Event Systems, Integrated Automation, Cement industry, Planning, Supervisory Control.

1. INTRODUCCIÓN

The cement industry is critical in the development of contemporary society to be basic raw material in the construction industry . The central element in the manufacture of cement is the clinker production, obtained by calcining a mixture mainly of: Calcium Oxide (CaO), silicon dioxide (SiO_2), Aluminum Oxide (Al_20_3), and iron oxide (Fe_20_3) found in limestone, clay and the iron ore. The process for Portland cement production is shown in Figure 1, and it is divided into the following stages:

(1) Conveying limestone; it is transported to warehouses in the cement plant and placed in cells having the same quality.
(2) Crushing and pre-homogenizing, which reduces the size of the limestone rocks that can be used by the raw mill and at this stage a first mixture of stone from different sources is performed to obtain a limestone (CaO) pre - homogenizing. The mixture is made to comply with predetermined percentages of limestone.
(3) Homogenizing. It makes a mixture of limestone (pre-homogenizing) with iron ore and clay to achieve the ideal blend, and in some cases corrective limestone is added. A stage of homogenization process is the raw milling , where minerals are reduced to a diameter of millimeters, so that the mixture is given particle level. Then, this milled mixture is carried deposits where homogenization of the mixture is complete, and then fed to the furnace. The product obtained is the raw meal, which feeds the oven.
(4) Clinkering, consisting in transforming the mixture (raw meal) in clinker by a cooking process; and finally,
(5) Cement production. By grinding the clinker and adding other products like gypsum, limestone, pozzolan, among others, that give texture and resistance

[1] Thanks to the Senescyt of Ecuador who partially funded the project and Cementos Guapán in Azogues, Ecuador for the information provided.

to use. The product obtained is distributed in bags (50 kg), or in bulk.

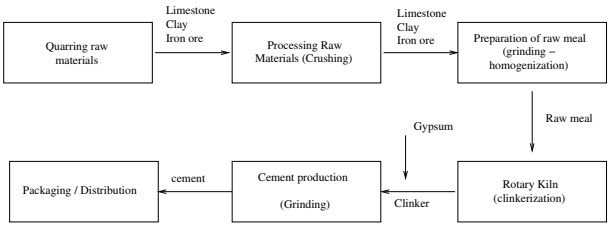


Fig. 1. General Process flow for the Production of cement. Modified from Huntzinger and Eatmon [2009]

When having the homogenized mixture, the mixture quality will determine its performance in the clinkering process. The oven has a continuous operation, so the input product must maintain a constant flow. The flow is determined by the oven capacity product quality at its input.

Input limestone comes from different quarries and has a high variability in their chemical composition and moisture levels , not only for its origin from different quarries, but also by the geological formation of the quarries. Limestone has an economic value according to their origin. The issue is to maintain an ideal choice for a flow that is determined by the furnace capacity to process raw meal mix at the lowest cost. The cost includes the cost of raw material and processing cost in the oven. The cost of processing furnace varies according to the quality of the raw meal.

An ideal mixing quality for the oven, which is feasible with available feedstock is established. It seeks to get that quality over time constantly mixing raw material from different quarries through dynamic allocation of resources and routes for forming pre-homogenizing limestone crushing corrective clay and preparation of corrective limestone, in amounts sufficient to maintain the constant flow of feeding to the furnace. The second step is feeding the mill uses a shared resource that is the transport system inputs, it goes from the tank pre-homogenized until the hoppers at the entrance to the mill.

Planning, programming, monitoring and even control production processes, can be obtained from models that reflect the behavior of the system in each of the above situations. Everything based on the basic objective is to produce, and which focuses on a request from a customer that accrues on, either in a delivery in warehouse or a production order in production or both. In most cases, when planning in a production process it is assumed for convenience a normal operation behavior; if required to do planning online, then you need to know the state of the process in order to generate an acceptable plan and, this is nothing more than take the product model (product route) and map on the set of available process equipment, incorporating physical constraints, considerations interconnection delays, which sets the schedule of when an order should start and completion date. Similarly, it requires a model for supervisory control schemes, whether coordination between units or supervision in each unit (resource). It is necessary to establish a planning mechanism for determining the amount of principal limestone, clay and limestone corrective to be produced; the online selection of the limestone quarries that is based on the quality obtained in the crushing process; supervisory control mechanism associated with the coordination of the stages.

The paper is organized as follows: the first section presents roughly the methodology to be used (This has been proven in commercial applications for companies in Merida, Venezuela and Medellin, Colombia) for the behavioral model of the process, in particular the model of the product routing. In the second section, the functional units and the associated IDEF diagrams are defined. In a third section the product route and Petri nets models associated with it and the functional units are shown, and as the model of behavior of the process is determined. A fourth unit shows how the pattern of behavior found is the model used for planning, programming, monitoring the process. Finally results and future work are presented. The results were found using a software tool developed by Janus Systems Merida.

2. PRODUCTION SYSTEMS MODELING USING DISCRETE EVENT SYSTEMS TECHNIQUES

To obtain a useful model production we require that serves to identify the model; in our case the models are used for: Planning, Programming, Supervision, Coordination, Monitoring of the production process. So far the useful descriptions thresholds used to detect conditions of the system; or the occurrence of events that can initiate, continue, enable, stop a process. This leads us naturally to the use of a description based on discrete event system, regardless of the nature or form of processing that has each of the entities making up the production process. A Dynamic Discrete Event System is a system whose dynamics is defined by changes in discrete variables, and this dynamic is driven by the occurrence of events. Within the formal techniques for the representation of Discrete Event Systems to find those based on Finite State Automata Ramadge and Wonham [1989], Wonham [2014] and Petri nets DAVID and ALLA [2001], David and Alla [2005]. Discrete Event Systems allow modeling the behavior of systems at different levels of the plant, from the plant floor allowing applications for PLC programming Uzam et al. [1996], Fabian and Hellgren [1998], Zhou and Twiss [1998]; at the supervisory level, in the description and supervisory control of batch processes Andreu et al. [1994, 1995], Viswanathan et al. [1998], Tittus and Åkesson [1999b,a], Tittus and Lennartson [1999], supervision of hybrid systems Antsaklis et al. [1998], Lemmon et al. [1999]; for levels of management, modeling and implementation of business processes van der Aalst [1998], van Der Aalst and Van Hee [2004], van Der Aalst et al. [2003a,b]; and integration between different levels Vernadat [2014], ISA-95 [2000], Chacón et al. [2008].

As we see, there are many authors who use DES as natural models to describe the production process (logical part), it is necessary to establish a production process configuration (physical part) so it can be executed. Figure 2 shows the procedure for establishing a configuration and run production activities, which is similar to the procedure proposed in Covanich and McFarlane [2009]. From the point of view of modeling, we see that is the model of behavior (route product) which is the generator of the

entire sequence shown in Figure 2, and therefore of the Planning, Programming, and Execution.

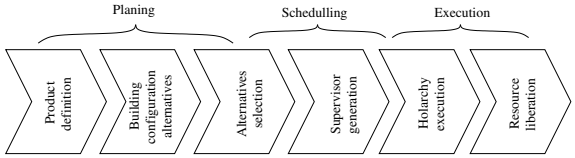


Fig. 2. Steps to establish a configuration

2.1 Product Model

A product model indicates the sequence of steps necessary to obtain a product , as shown in Figure 3. At each step, the competencies (skills) necessary for their implementation, as well as inputs (raw materials and services) and outputs for stage specified. In addition, information about the time needed to perform the step.

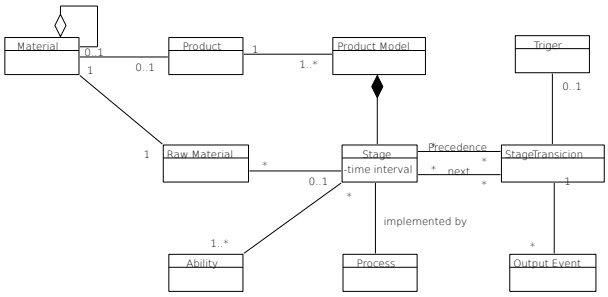


Fig. 3. Product Model, Production Model in UML

2.2 Description of the behavior of the Functional Unit

A functional unit is defined as responsible for carrying out a required skill, either in a business process or a production process. To describe the behavior of the functional unit, we rely on the deployment architecture shown in Figure 4. In our case, we only focus on the skills required in the production process. Functional Units generate services for a product. The relationship between the model and the functional units is obtained by the skills required by the product model and provided by the Functional Unit.

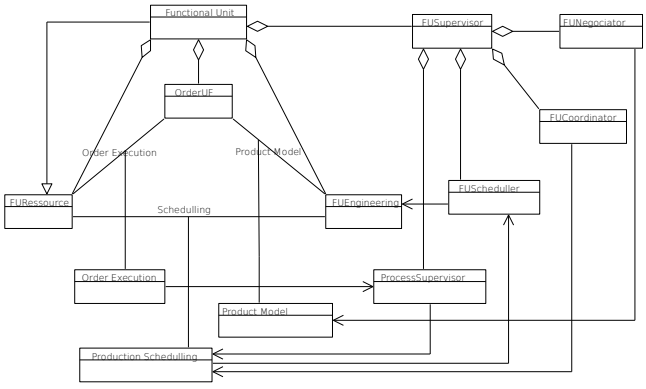


Fig. 4. Functional Unit model in UML

As shown, the Functional Unit has a core containing the holarchies management, consisting of a set of behavioral models that describes the functionality of the entities belonging to the unit. This set of structured models can be grouped to form holarchies . A holarchy shows a model in principle overall (useful) of the functional unit to accomplish a target.

This core set of universal behavior patterns in the UF depending on the purpose of production; considers aspects such as: normal condition , degradation of the operational condition and the functional abnormality between the components of the unit as well as the condition and the levels of the raw material. Models built in this way can be used *to track* the processes occurring in the entities (unit, plant and equipment).Supervisors can also be generated (coordination and supervisory functions) and drivers for regulation.

Functional Unit (**FU**), as a whole, is responsible for evaluating the feasibility of an activity and the execution of that activity (programming, implementation); as well as achieve coordination with other FU for the implementation of activities through cooperative interactions (runtime). The dynamic behavior of the FU can be described by Discrete Event Systems (DES), in our case, we use hierarchical Petri nets.

2.3 Modelling by Petri Nets

Petri Nets can describe the different processes that occur in a production system in an easy way. We use the same model for planning and programming activities and to monitor the actual process. The network locations represent us process states, or the resources used, as shown in Figure 5.

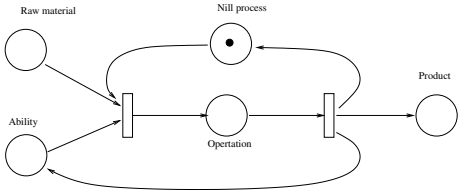


Fig. 5. Petri net for overall description of a process

In Johnsson [1997] the concept of interpreted Petri nets to assign meanings to places and transitions, so they represent the production process is used. Places, transitions, arcs and tokens have a semantics that can describe in sufficient detail the production process. In the description of the process model, a place represents: a needed input; if it is associated with resource indicates capabilities (in the case of equipment / functional units), or abilities (human resource) y and the amount of effort necessary to accomplish the task, the arcs define the amount of product consumed or generated into the process or the quantity of resource capability used. The network shown in Figure 5 corresponds to a product model and it will be used for planning. On the other hands, Timed Petri Nets allow duration or dates associated models, representing a production process. Transitions are associated with the occurrence of the events on the plant floor, and its implementation these events are captured by the installed technology. An event can be associated with the arrival of a production order, the completion of a task.

2.4 Behavior model for a production unit

The behavior of an elementary process can be expressed by a Discrete Event Systems, and the composition of elementary models based on interactions associated with transitions, allows us to obtain the overall behavior of a system. This methodology has been validated by Janus Sistemas in several applications. In Chacón et al. [2008] it is described the methodology to generate behavioral models of the production processes to planning, monitoring and supervise the process. In this paper we present only part of the methodology that allows us to model behavior of the first stage of the process of obtaining clinker (pre-homogenization) in order to the construction of the model consists of:

(1) Product description routes, such as establishing an acceptable configuration, represented in IDEF (IDEF).
(2) Description of functional units, their skills (abilities) and their interactions. Description of the states of the unit depending on regions and operation modes, and its representation in IDEF.
(3) Building product model, and its representation by Petri Nets.
(4) Construction of models of functional units and their representation using Petri Nets.
(5) Obtaining behavioral models using the model projection path product on equipment (Functional Units).
(6) Using the model of behavior in the previous step, get the model to planning, monitoring, and supervision a production order.
(7) Generate a plan and schedule for the production order and its monitoring mechanisms.

As shown, the pattern of behavior is nothing more than a holarchy models. These models are represented in Petri nets and lead to the global model of the production process, which is used to plan, schedule, monitor and follow a production order. The application used interprets these types of models efficiently. This approach to implementation of automation can be used interchangeably if the company is hierarchical, heterarchical or holonic.

3. CASE STUDY: PRE - HOMOGENIZATION STAGE IN CLINKER PRODUCTION

3.1 Modeling the whole production process

The first step in building models is to achieve global flows of information and products. The IDEF0 model in Figure 6 shows the information and products flows for the Clinker production process.

Thus, the clinker production process is described in the first 4 major stages of the 5 shown in section 1: Quarries, Crushing – Pre-homogenisation, homogenisation, and Clinker cooking (several limestone). Transportation between quarries and crushing is given by trucks. Interconnection between pre - homogenisation and mill is given by a belt (limestone, corrective limestone, clay and iron ore). Interconnections should be multiplexed to allow the transport of different products. Between the homogenizer and the furnace only the raw flour is transported.

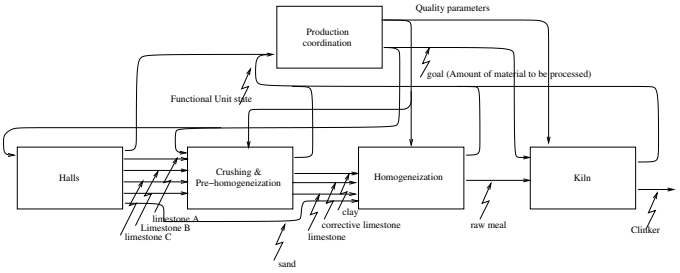


Fig. 6. General model (IDEF0) for the activities in the Clinker manufacturing process

3.2 Modeling the pre-homogenization process

The main raw material of the process comes from accumulations of limestone from different quarries and have a high variability in their chemical composition and moisture levels as it was said in introduction. The first step is the conditioning of the raw material in order to be transported and processed by the mill. Preparing raw material consists of three separate threads that are conditioning main limestone, limestone corrective preparation and conditioning of clay; these processes use the same resources: crusher and trucks. All these elements are stored in piles for subsequent transport to the raw mill, along with iron ore (ferrous sand).

The feeding process the raw mill, recovering the pre-homogenized limestone, corrective limestone, clay and sand corrective ferrous from the formed piles. The entire process eliminate variability in two stages, in the first, homogenization of the main product, and in the second phase, a completion is performed adding minerals that are not present in the limestone.

To determine the sources of raw material, the system takes into account: availability, quality and cost of raw materials stored in order to determine the optimal combination. The result gives tons of raw material from each source to use. In general, the expected quality of the raw meal is given in table 1.

Material	Min	Max
CaO	65	68
SiO_2	20	23
Al_20_3	4	6
Fe_2O_3	2	4
MgO	1	5

Table 1. Range of the main components

The properties of the raw materials are given in the table 2.

	Sou 1		Sou. 2		Sou. 3	
Material	Min	Max	Min	Max	Min	Max
SiO_2	8	25	x	x	x	x
Al_20_3	1	4,5	x	x	x	x
Fe_2O_3	1	4	x	x	x	x
CaO	40	43	45	x	x	x
MgO	0,3	1	x	x	x	x

Table 2. Range of the main components by quarry

A IDEF description associated with the physical structure of the plant, corresponding to the steps of Figure 6, is

given in Figure 7. The figure shows the information and product flows shown for the two processes analyzed. It can be realized that pre - homogenization part has a feeding system that is discret and, the the second part (homogenization) has a feeding systems that is by batch. The output of the whole process should be continuous, and this is achieved by the use of a temporary storage system between stages.

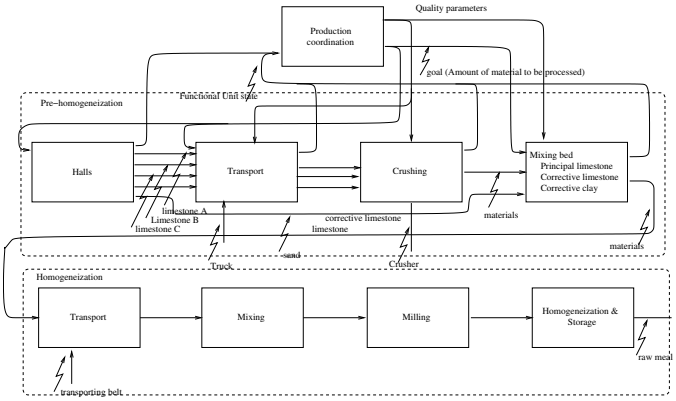


Fig. 7. Pre - homogenisation and Homogenization processes in the production of Clinker

3.3 Functional units in the first part: Crushing process and Pre-homogenization processes

The Functional Units in the pre - homogenisation are: Quarries, Transport Unit, Grinding Unit and Storage Unit. The process should produce three products: the main limestone, corrective limestone and, clay corrective. These products are three of the four inputs to the raw mill. The main resources that are used at this stage are the crusher and trucks. This stage consists of three processes: the transport of materials , mixing crushing of materials and creating layered stacks . The three processes are given separately and are the result of a plan that specifies the amount of material that has been produced and the quality expected as the IDEF shown in Figure 8.

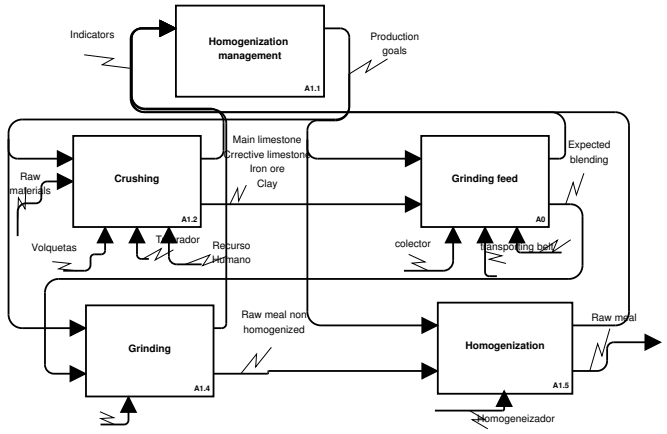


Fig. 8. Products and information flows in the process of homogenization

The crusher is the primary constraint and is used to form the stacks of main limestone, limestone and corrective corrective clay. The amount of each of the materials is determined by the optimization system by using figures in the table 3. The crusher works 8 hours/day, 5 days/week and must process at least the amount of product that will be consumed by the furnace daily taking into account the performance of the raw material in the clinkering process. The amount of production is known by a weighbridge system. The information and products flows are shown in the IDEF Figure 9

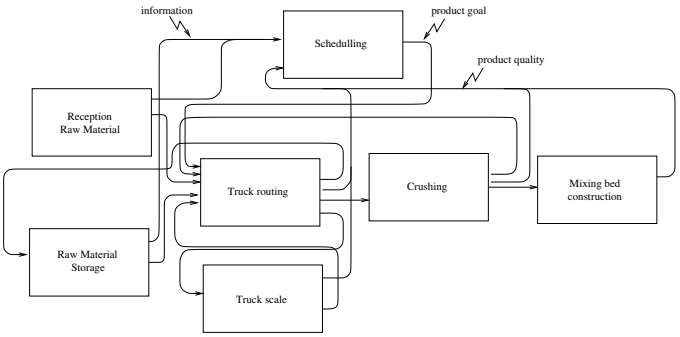


Fig. 9. IDEF0 for homogenisation process

Preparation of the main limestone The objective in the first step is to mix the various limestone quarries for a pre-homogenised limestone with a percentage of CaO over 70 % (Titration 100).

This process gets a product that, in addition to calcium oxide, has silicon, aluminum, iron and magnesium with the values given in Table 3. The product is stored in stacks of layers of material.

	SiO_2 (%)	Al_20_3 (%)	Fe_2O_3 (%)	CaO
Min	7	1	1	70
Max	10	4.5	4	75

Table 3. Expected range of chemical components of the limestone mixing bed

The properties of the materials, according to their origin, are given in the table 4.

Cantera	SiO_2 (%)	Al_20_3 (%)	Fe_2O_3 (%)	CaO (%)
A	13.1	1.7	0.8	45.2
B	6.9	0.5	0.2	50.7
C	2.5	0.7	2.8	45.5
Arcilla	43.3	27.4	11.5	0.5

Table 4. Chemical composition of limestone in each quarry

The estimated amount of $\%CaO$ in the limestone pre - homogenized is done by the following calculation:

$$\%CaO = \frac{\%CaO_1 \times M_1 + \%CaO_2 \times M_2}{M_1 + M_2} \quad (1)$$

Moisture and the presence of clay affect the grinding process, so that when selecting the sources of limestone, humidity is a new restriction on the crusher. Deficiencies of other chemicals are resolved in the process of feeding raw flour mill.

For the mixing bed in formation, the expected amount of CaO is calculated for each truck using the following equation:

$$\%CaO_{ac_k} = \frac{\%CaO_{ac_{k-1}} Mass_{ac_{k-1}} + \%CaO_k Mass_k}{Mass_{ac_{k-1}} + Mass_k} \quad (2)$$

It has an on-line analyzer for percent of CaO pre- homogenized, and similarly another analyzer for feeding to the mill.

Preparation of corrective clay Clay provides the silicon component necessary for the preparation of the clinker, but due to the amount of moisture that is, it must be mixed with limestone so it can be crushed. The result is a clay with a high percentage of limestone.

Preparation of corrective limestone Corrective limestone is limestone with a high percentage of CaO. It will be used in the raw mill to ensure the required percentages of calcium carbonate. If the pre-homogenization process was efficient, the amount of corrective limestone must tend to zero.

4. BEHAVIORAL MODELS OF FUNCTIONAL UNITS AND PRODUCT IN PETRI NETS

The flow of information and products crushing activity shown in Figure 10. Management should plan crushing activities according to the needs of pre-homogenized Limestone, corrective Limestone and corrective Clay. Then, the management system makes an vehicles assignment, to define routes according to the on-line calculation of the product quality that has being obtained.

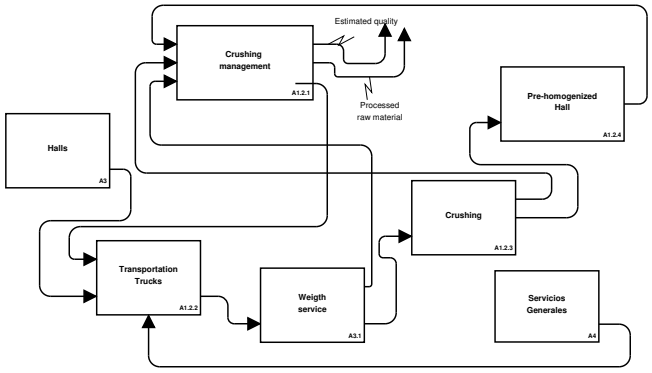


Fig. 10. Flow of information and products in the grinding process

Products and raw materials are materials and are described as defined by ISA-88. Table 5 presents this information.

Material	Material class
Limestone_A	Limestone
Limestone_B	Limestone
⋮	
Clay	Clay
Corrective limestone	Crushed limestone
Production limestone	Crushed limestone

Table 5. Materials

For a stage, which is complex, it has a tool that tracks each stage. Using a holonic structure, an interpreter models follow the patterns of behavior of the stage to monitor and supervise the process.

Figure 11 shows the behavioral model in Petri Nets for the production process of the main limestone.

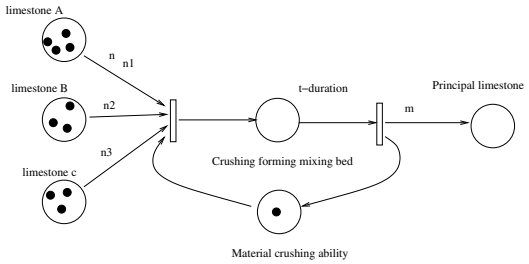


Fig. 11. Product Model (Behavior of the production process)

The behavior model production process limestone is shown in Figure 12. We can see that this is just the instantiation (in detail) of the product model on the equipment, in order to have the physical process, including capabilities, restrictions, etc.

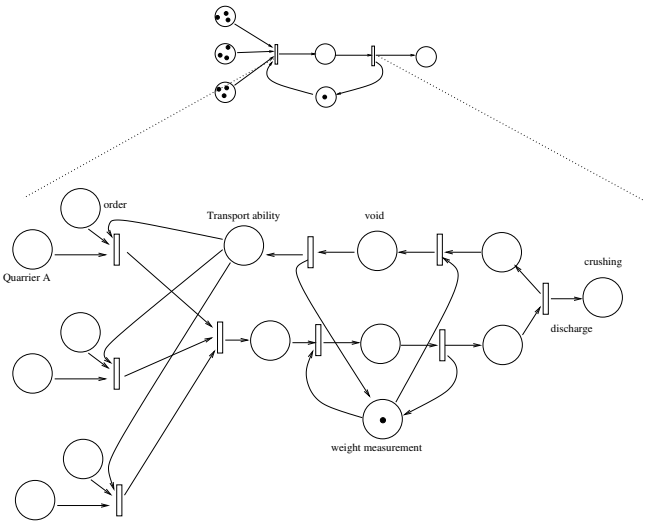


Fig. 12. Production process that implements the steps of production Product Model

Resources for Production The table 6 contains information of physical infrastructure. These elements are considered as functional units with abilities to perform the tasks of planning and monitoring. The components of the equipment hierarchy are considered as a functional units, which are able to build internal plans, track activities, monitor the implementation of activities and execute activities.

5. GENERATING A PRODUCTION PLAN, AND SUPERVISORY CONTROL SYSTEM: FENIX

For planning, a network is constructed from network behavior Figure 11. The information contained in the network allows to validate the feasibility of the implementation of a process. This requires a transformation of the initial network in a network that includes supplies and equipment using information from existing products in inventory and equipment that are available.

In case there are several production orders that use the same input and the same product, the initial network build

Equipment	Father Equipment	Ability	is a FU
Cement Plant		Cement production	yes
Quarry	Cement Plant	Limestone storage	yes
Quarry Limestone_A	Quarry	Limestone_A storage	yes
. . .	. . .	. . .	. . .
Crushing system	Cement plant	Material crushing	yes
Crusher	Crushing system	Crushing	yes
Transportation belt	Crushing system	transport crushed material	no
Mixing bed feeder	Crushing system	Mixing bed feeder	no
Quarry transport	Cement plant	Material transport	yes
Truck_1	Quarry transport	Truck material transport	yes
. . .	. . .	. . .	. . .

Table 6. Physical infrastructure

a global network validating the viability of the overall process according to steps (schedule of resources).

Thus, an implementation of the expected behavior of the system for the crusher & Pre-homogenization is shown in Figure 13. The crusher is a resource that can not be shared simultaneously , and this means that only one product can be over a period of time.

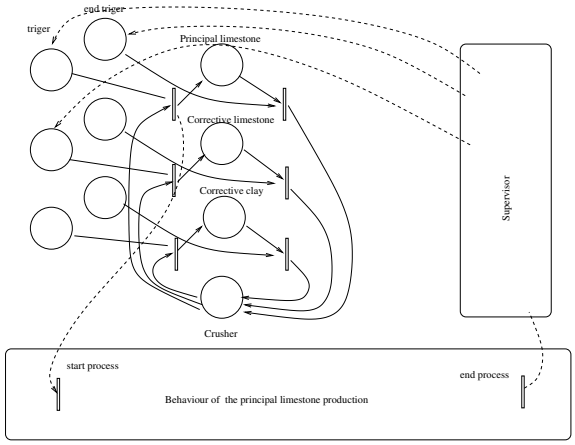


Fig. 13. Behavior of the crushing system

A scheme for tracking an order in the grinding process and the status of the order is shown in Figure 14.

This is feasible in this manner due to the availability of an application capable of interpreting these models and will run equipment assignments based on the quality of the batteries in training. The variability of the batteries minimizes ensuring product quality at the entrance of the mill.

6. CONCLUSIONS

The implementation of a monitoring scheme based online behavioral models ensures quality objectives raised, decreasing allocation tasks truck when performing this automatically. The same monitoring system allows integration with the administrative processes of the organi-

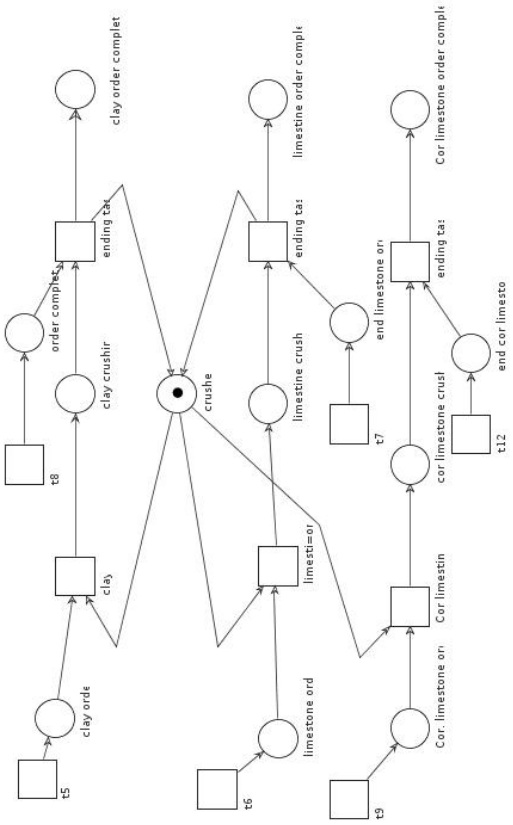


Fig. 14. Dynamic allocation of equipment according to the type of material to be produced

zation maintaining current existence of materials, use of equipment, hours worked, etc.

The modeling process is relatively easy, and the possibility of interpretation models for application reduces development time supervisors, eliminates the possibility of errors in coding, and allows you to incorporate new models of behavior when changes such as occur add a new quarry, changes in the quality of material or new specifications for the stacks of material.

The technology used for the implementation of supervisors is inexpensive and using PLC (programmable control devices) ensures the reliability of the instrumentation on the plant floor.

REFERENCES

Andreu, D., Pascal, J., Pingaud, H., and Valette, R. (1994). Batch process modelling using petri nets. In *Systems, Man, and Cybernetics, 1994. Humans, Information and Technology., 1994 IEEE International Conference on*, volume 1, 314–319. IEEE.

Andreu, D., Pascal, J., and Valette, R. (1995). Interaction of discrete and continuous parts of a batch process control system. In *ADEDOPS Workshop, Imperial College, Londo.*

Antsaklis, P., Koutsoukos, X., and Zaytoon, J. (1998). On hybrid control of complex systems: A survey. *Journal européen des systèmes automatisés*, 32(9–10), 1023–1045.

Chacón, E., Albarrán, J.C., Chacón, R., Rojas, O., and Gutiérrez, D. (2008). Metodología para la automati-

zación integrada de procesos de producción basada en el enfoque holónico. In *XIII CLCA/VI CAC*.

Chacón, E., Besembel, I., Rivero, M., and Cardillo, J. (2008). *Advances in Robotics, Automation and Control*, chapter The holonic production unit: an approach for an architecture of embedded production process, 301–314. ITech_ISA.

Covanich, W. and McFarlane, D. (2009). Assessing ease of reconfiguration of conventional and holonic manufacturing systems: Approach and case study. *Engineering Applications of Artificial Intelligence*, 22, 1015 – 1024.

DAVID, R. and ALLA, H. (2001). On hybrid petri nets. *Discrete Event Dynamic Systems: Theory and Applications*, 11, 9–40.

David, R. and Alla, H. (2005). *Discrete, Continuous, and Hybrid Petri Nets*. Springer.

Fabian, M. and Hellgren, A. (1998). Plc-based implementation of supervisory control for discrete event systems. In *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, volume 3, 3305–3310. IEEE.

Huntzinger, D.N. and Eatmon, T.D. (2009). A life-cycle assessment of portland cement manufacturing: comparing the traditional process with alternative technologies. *Journal of Cleaner Production*, 17(7), 668–675.

IDEF (????). Integrated definition models. http://www.idef.com/.

ISA-95 (2000). Ansi/isa–s95.00.01–2000, enterprise-control system integration. part 1: Models and terminology. Technical report, ISA.

Johnsson, C. (1997). Recipe-based batch control using high-level grafchart. Technical report, Department of Automatic Control, Lund Institute of Technology.

Lemmon, M., He, K., and Markovsky, I. (1999). Supervisory hybrid systems. *Control Systems, IEEE*, 19(4), 42–55.

Ramadge, P.J. and Wonham, W.M. (1989). The control of discrete event systems. *Proceedings of the IEEE*, 77(1), 81–98.

Tittus, M. and Åkesson, K. (1999a). Deadlock avoidance in batch processes. In *Proc. of 14th IFAC World Congress, Beijing, PR China*, 397–402. Citeseer.

Tittus, M. and Åkesson, K. (1999b). Petri net models in batch control. *Mathematical and Computer Modelling of Dynamical Systems*, 5(2), 113–132.

Tittus, M. and Lennartson, B. (1999). Hierarchical supervisory control for batch processes. *Control Systems Technology, IEEE Transactions on*, 7(5), 542–554.

Uzam, M., Jones, A., and Ajlouni, N. (1996). Conversion of petri net controllers for manufacturing systems into ladder logic diagrams. In *Emerging Technologies and Factory Automation, 1996. EFTA'96. Proceedings., 1996 IEEE Conference on*, volume 2, 649–655. IEEE.

van Der Aalst, W. and Van Hee, K.M. (2004). *Workflow management: models, methods, and systems*. MIT Press.

van der Aalst, W.M. (1998). The application of petri nets to workflow management. *Journal of circuits, systems, and computers*, 8(1), 21–66.

van Der Aalst, W.M., Ter Hofstede, A.H., Kiepuszewski, B., and Barros, A.P. (2003a). Workflow patterns. *Distributed and parallel databases*, 14(1), 5–51.

van Der Aalst, W.M., Ter Hofstede, A.H., and Weske, M. (2003b). Business process management: A survey. In *Business process management*, 1–12. Springer.

Vernadat, F. (2014). Enterprise modeling in the context of enterprise engineering: State of the art and outlook. *International Journal of Production Management and Engineering*, 2(2), 57–73.

Viswanathan, S., Johnsson, C., Srinivasan, R., Venkatasubramanian, V., and Ärzen, K.E. (1998). Automating operating procedure synthesis for batch processes: Part i. knowledge representation and planning framework. *Computers & chemical engineering*, 22(11), 1673–1685.

Wonham, W. (2014). *SUPERVISORY CONTROL OF DISCRETE-EVENT SYSTEMS*. University of Toronto.

Zhou, M. and Twiss, E. (1998). Design of industrial automated systems via relay ladder logic programming and petri nets. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 28(1), 137–150.

Modified PI control for the Stabilization and Control of a class of High-order System with Delay

M. A. Hernández–Pérez [**] **B. del Muro–Cuéllar** [*]
M. Velasco–Villa [*,⋆] **D. F. Novella–Rodríguez** [*]
R. A. Garrido–Moctezuma [**]

[*] *Escuela Superior de Ingeniería Mecánica y Eléctrica, Unidad Culhuacan, Instituto Politécnico Nacional, 04430, México D.F. (e-mail: mahp_hernandez@hotmail.com, bdelmuro@yahoo.com, velasco@cinvestav.mx).*
[**] *CINVESTAV-IPN, Av. IPN, No. 2508, Col. San Pedro Zacatenco, 07300, México D.F. (e-mail:garrido@ctrl.cinvestav.mx)*

Abstract: In this work it is addressed the problem of the stabilization and control of a specific class of high-order unstable linear systems with time delay. The system under consideration has one unstable pole, q stable poles and a minimum phase zero. Sufficient conditions to guarantee the stability of the closed loop system by a modified PI scheme are provided. The proposed strategy consist of a modified version of the traditional PI controller, which include a first order filter and conduces to the, here called, PI_f controller. This new scheme allows the improvement of the existing results when a traditional PI is used for high-order systems with time delay. The proposed result is illustrated by its application to a numerical example: control of a Continuously Stirred Tank Reactor (CSTR) linear model.

Keywords: Unstable system, minimum phase zero, time-delay, modified PI control.

1. INTRODUCTION

Time delays often arise in control systems, either from delays in the process itself or from delays during the processing of sensed signals. Industrial processes often presents time delays introduced by the finite time that material takes to flow through pipes. In the same way time delay could be produced due to heat and mass transfer in chemical industries, heavy computations and hardware restrictions of computational systems, high inertia in systems with heavy machinery and communications delays in space craft and remote operation.

Typical examples of systems exhibiting time delays are chemical processes Richard (2003), transportation systems, communication and power systems Franklin et al. (1995); Wang et al. (1999). In some others cases, delays are introduced by sensors and actuators devices Xian et al. (2005), Liu et al. (2005). In most cases this phenomenon is one of the main causes of instability and poor performance and it produces, in general, unwanted behaviors in dynamical systems. So, it is necessary to take special attention to the stability analysis and controller design to handle systems with time delays. Thus, there exist great motivation for the study of effects causing a time delay in dynamic systems Hu and Lin (2001), Trentelman et al. (2001) and Shamsuzzoha et al. (2009). In addition, the problem becomes more complicated when the system not only has a time delay but also it is unstable. For this reason, the interest of dealing with unstable processes containing a delay term has been growing in the control community Lee et al. (2010), Sipahi et al. (2011) and Gu et al. (2003) recently.

The control problem of time delay systems has been studied by different perspectives. The simplest approach consists on the approximation of the delay operator by means of Taylor or Pade series expansions which leads to a nonminimum- phase models with rational transfer function representation Munz et al. (2009). Another approach is to compensate the effect of time delays by removing the exponential term from the characteristic equation of the process. This technique was introduced by Smith (1957) and it is well-known as Smith Predictor (SP). This technique does not have a stabilization step which restricts its application to open-loop stable plants. To deal with this disadvantage, some modifications of the original SP structure have been proposed (for instance Palmor (1996), Seshagiri et al. (2007), Kawnish and Choudhury (2012)).

Another solution to control delayed systems is to use a Proportional-Integral- Derivative (PID) controllers. PID controller is widely used in the control of industrial processes due to its simple structure in many practical processes (Silva et al. (2005)). Also, they are frequently used to stabilize unstable time-delay processes. However, stability conditions for such processes are a very challenging topic. Huang and Chen (1997), use root locus diagrams to study the stabilizability problem of unstable delay pro-

⋆ M. Velasco-Villa is on sabbatical stay supported by Conacyt (No. 260936) from Mechatronic Section of the Electrical Engineering Department.

cesses using simple controllers and show that a normalized time delay should be less than 1 for P/PI controllers, while it should be less than 2 for the case of PD controller.

Silva et al. (2005), investigated the complete set of stabilizing PID controllers based on the Hermite-Biehler theorem for quasi-polynomials. This approach requires a lot of mathematical processing, in case the system has large dimension. Moreover, this approach does not provide an explicit characterization of the stabilizing PID parameter in terms of the maximal time delay. Hwang and Hwang (2004) applied the D-partition method to characterize the stability domain in the space of system and controller parameters. Thus, the stability boundary is reduced to a transcendental equation, and the whole stability domain is drawn in a two-dimensional plane by sweeping the remaining parameters. However, this result only provides a sufficient condition regarding the size of the time delay for stabilization of first-order unstable processes.

In Xiang et al. (2007), a frequency approach for the design of P/PI controllers focused on the specific class of second order systems with an unstable pole was introduced. With a similar approach, in Lee et al. (2010), a generalization for high order systems with an unstable pole is given. In both cases, the stability results are obtained in terms of the upper bound of the time-delay, however they do not address the problem of systems with a minimum phase zero.

In this work, a modified version of the traditional PI control scheme is proposed . This new control strategy will be denoted henceforth as a PI_f controller. This new scheme, besides of keeping the basic properties of a conventional PI controller such as disturbance rejection and reference tracking of step type signals, also allows the stabilization and control of the same family of systems considered in Lee et al. (2010), but with the advantage of the delays supported by this new scheme are larger than those supported by the traditional PI. Additionally, one of the main advantages of the PI_f is that it can deal with unstable systems which include a transmission zero, which is an interesting feature from the control point of view. Sufficient conditions to stabilize high-order unstable systems with time delay and a minimum phase zero are presented in this work. The stabilization conditions are expressed in terms of the maximum allowable time–delay magnitude. Also it is provided a procedure for determining the parameter ranges of the stabilizing controller. The proposed control scheme is illustrated by a numerical example to control the temperature in an unstable linear model of a Continuously Stirred Tank Reactor (CSTR).

The rest of the work is organized as follows: Section 2 presents the problem statement. Section 3 addresses the proposed control strategy, establishing the sufficient condition for the existence of a stabilizing control structure. In Section 4, the results are applied to a numerical example. Finally in Section 5 the conclusions are presented.

2. PROBLEM STATEMENT

Consider the following class of single input single output (SISO) linear time invariant systems (LTI) with delay in the input-output path given by,

$$\frac{Y(s)}{U(s)} = G(s)e^{-\tau s} = \frac{\alpha(s+\beta)}{(s-\gamma)\prod_{m=1}^{q}(s+\delta_m)}e^{-\tau s} \qquad (1)$$

with γ, δ_m and $\beta > 0$. Note that the system has an unstable pole, q stable poles, a zero in the left half complex plane s and time delay $\tau > 0$. The objective is to provide sufficient conditions to stabilize this class of systems by a PI-modified (PI_f) illustrated in Fig. 1.

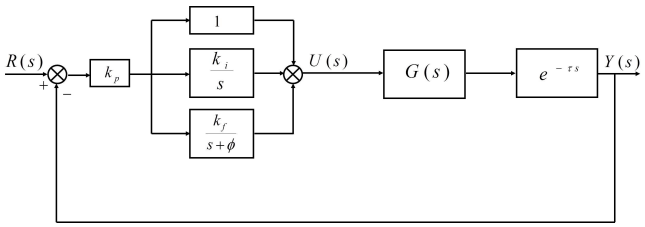


Fig. 1. Proposed control strategy PI_f

The PI_f control represented in Fig. 1 is given by,

$$H(s) = k_p\left(1 + \frac{k_i}{s} + \frac{k_f}{s+\phi}\right). \qquad (2)$$

The proposed PI_f control consists of a modification to the conventional structure of PI controller by adding a first order filter. In this approach is proven that a proposed PI_f controller not only improves the stability conditions reported in the literature using the traditional PI, but also the PI_f keeps the basic properties of a PI controller; disturbance rejection and reference tracking of step type signals.

3. PROPOSED CONTROL STRATEGY

This section present sufficient conditions to stabilize the close-loop system (1)-(2). Note that this control strategy produces an open-loop transfer function $Q(s)$ represented by,

$$Q(s) = H(s)G(s)e^{-\tau s}. \qquad (3)$$

To design a control strategy, the following theorem is stated.

Theorem 1. Consider the class of high-order time-delayed system with one unstable pole and a zero in the left half complex plane give by (1). There exists a PI_f controller given by (2) such that the corresponding closed–loop system is stable if,

$$\tau < \frac{1}{\gamma} - \sum_{m=1}^{q}\frac{1}{\delta_m} + \sqrt{\frac{1}{\gamma^2} + \sum_{m=1}^{q}\frac{1}{{\delta_m}^2}}. \qquad (4)$$

Proof. Suppose that condition (4) is satisfied. Due to the freedom selecting the parameter ϕ of the PI_f controller given by (2), it is possible to obtain a cancellation of the zero of the system (1) by $\phi = \beta$. Therefore, the open-loop response is obtained as,

$$Q(j\omega) = k_p\alpha\frac{\cancel{\left((j\omega)^2 + (k_f + k_i + \phi)j\omega + k_i\phi\right)(j\omega+\beta)}}{j\omega\,(j\omega-\gamma)\cancel{(j\omega+\phi)}\prod_{m=1}^{q}(j\omega+\delta_m)}e^{-j\omega\tau}. \quad (5)$$

In order to facilitate the analysis, it is possible to consider the following change of variables for k_p, k_i and k_f such that,

$$\begin{cases} \bar{k_p} = k_p(k_f + k_i + \phi) \\ \bar{k_f} = \frac{1}{k_f+k_i+\phi} \\ \bar{k_i} = \frac{k_i\phi}{k_f+k_i+\phi}. \end{cases} \quad (6)$$

From (6), the open-loop response (5) can be rewriten as,

$$Q(j\omega) = \bar{k_p}\bar{k_f}\alpha\frac{\left((j\omega)^2 + (\frac{1}{k_f})j\omega + \frac{\bar{k_i}}{k_f}\right)}{j\omega\,(j\omega-\gamma)\prod_{m=1}^{q}(j\omega+\delta_m)}e^{-j\omega\tau}. \quad (7)$$

To simplify the analysis, let us consider $k_i = 0$. Then, the phase and magnitude expression in the frequency domain of (7) are given by,

$$\angle Q(j\omega) = -\left(\pi - \arctan\left(\frac{\omega}{\gamma}\right)\right) - \omega\tau + \arctan\left(\bar{k_f}\omega\right) - \sum_{m=1}^{q}\arctan\left(\frac{\omega}{\delta_m}\right) \quad (8)$$

$$M_{Q(j\omega)} = \bar{k_p}\alpha\sqrt{\frac{1+\bar{k_f}^2\omega^2}{(\omega^2+\gamma^2)\prod_{m=1}^{q}\left(\omega^2+{\delta_m}^2\right)}}. \quad (9)$$

Taking into account the Nyquist stability criterion, system (7) will be stable if and only if $N + P = 0$, where P the number of poles in the right half complex plane and N the number of rotations to the point $(-1, 0j)$. In this case, as $P = 1$, it is required to have a counterclockwise rotation to the point $(-1; 0)$ in the Nyquist diagram.

Therefore, since the phase expression has an initial value of $-\pi$ and assuming that condition (4) is satisfied, to have a counterclockwise direction in the Nyquist diagram the magnitude expression $M_{Q(j\omega)}$ must be an strictly decreasing function of ω and the phase expression $\angle Q(j\omega)$ must be an increasing function of ω. From this fact, it is possible to prove based on (4) that,

$$\frac{d}{d\omega}\left(\frac{M_Q^2}{\bar{k_p}^2\alpha^2}\right)\bigg|_{\omega=0} < 0. \quad (10)$$

and

$$\frac{d}{d\omega}\left(\angle Q(j\omega)\right)\bigg|_{\omega=0} > 0 \quad (11)$$

Consequently, if (4) is satisfied, there exist gains $\bar{k_p}$, $\bar{k_f}$ and $\bar{k_i}$ such that the system (1) is stable in the closed-loop.

Therefore, from the decreasing and increasing properties stated in (10) and (11), and after some computations, the set of stabilizing $\bar{k_f}$ values are given by,

$$\tau - \frac{1}{\gamma} + \sum_{m=1}^{q}\frac{1}{\delta_m} < \bar{k_f} < \sqrt{\frac{1}{\gamma^2} + \sum_{m=1}^{q}\left(\frac{1}{\delta_m^2}\right)}. \quad (12)$$

When the integral part $\bar{k_i}$ is considered to be different from zero, the phase expression $\angle Q(j\omega)$ and magnitude expression $M_{Q(j\omega)}$ are represented by (13) and (14), respectively

$$\angle Q(j\omega) = -\left(\pi - \arctan\left(\frac{\omega}{a}\right)\right) - \omega\tau + \arctan\left(\bar{k_f}\omega - \frac{\bar{k_i}}{\omega}\right) - \sum_{m=1}^{q}\arctan\left(\frac{\omega}{b_m}\right) \quad (13)$$

$$M_{Q(j\omega)} = \bar{k_p}\alpha\sqrt{\frac{1+\left(\bar{k_f}\omega - \frac{\bar{k_i}}{\omega}\right)^2}{(\omega^2+a^2)\prod_{m=1}^{q}\left(\omega^2+{b_m}^2\right)}}. \quad (14)$$

Taking into account $\bar{k_i} = 0$ the magnitude expressions (14) and (9) are equivalents. In the same way, it can also be noted that (13) is equivalent to (8) when $\bar{k_i} = 0$. Considering this fact, it is possible to follow the principle of argument continuity for $\bar{k_i}$, this is, it is always possible to choose a small enough $\bar{k_i}$ gain such that conditions (10) and (11) are fulfilled. Thus, once an stabilizing $\bar{k_f}$ gain is found, it is always possible to choose a small enough $\bar{k_i}$ gain such that the Nyquist stability criterion is satisfied and the system (7) can be stabilized by a PI_f controller.

From this fact, if the phase and magnitude condition are satisfied, then there exists a adequate $k_p > 0$ such that the Nyquist diagram will start in the correct position and with a counterclockwise rotation, then the Nyquist stability criterion will be satisfied. The range of $\bar{k_p}$ values is given by,

$$\bar{k_p}(\omega_{c_1}) < \bar{k_p} < \bar{k_p}(\omega_{c_2})$$

with $\omega_{c_1} < \omega_{c_2}$ being the first two phase crossover frequencies solutions of,

$$\arctan\left(\frac{\omega_{c_i}}{\gamma}\right) - \omega_c\tau + \arctan\left(\bar{k_f}\omega_{c_i} - \frac{\bar{k_i}}{\omega_{c_i}}\right) - \sum_{m=1}^{q}\arctan\left(\frac{\omega_{c_i}}{\delta_m}\right) = 0. \quad (15)$$

where $\omega_{c_{i=1,2}}$ are crossover frequencies and $\bar{k_p}(\omega_{c_i})$ are given by

$$\bar{k_p}(\omega_{c_i}) = \frac{1}{\alpha_1}\sqrt{\frac{\left(\omega_{c_i}^2+\gamma^2\right)\prod_{m=1}^{q}\left(\omega_{c_i}^2+{\delta_m}^2\right)}{1+\left(\bar{k_f}\omega_{c_i} - \frac{\bar{k_i}}{\omega_{c_i}}\right)^2}}. \quad (16)$$

Thus, if condition (4) is is fulfilled, there always exist $\bar{k_f}$, $\bar{k_i}$ and $\bar{k_p}$ gains which are able to stabilize systems of the class (1) in close-loop by using a PI_f control structure.

Remark 2. It is worth noting that the proposed PI_f controller, in this work, not only maintains the basic properties of a convencional PI controller regarding disturbance rejection and reference tracking of step type signals. Besides, it presents two advantages over P/PI controller; as the stability condition is improved adding the term $\sqrt{\frac{1}{\gamma^2}+\sum_{m=1}^{q}\frac{1}{\delta_m{}^2}}$, i.e., the PI_f allows stabilizing systems with larger delays than the ones allowed by the conventional P/PI controller. The second advantage is that the PI_f control allows state conditions to stabilize systems with a zero which have not been addressed in the literature.

4. NUMERICAL EVALUATION

4.1 Control of CSTR by modified-PI controller

The following example has been taken from Bequette (2003). It is considered the chemical process of a unstable continuos stirred tank reactor. A general description of a CSTR is shows in Fig. 2.

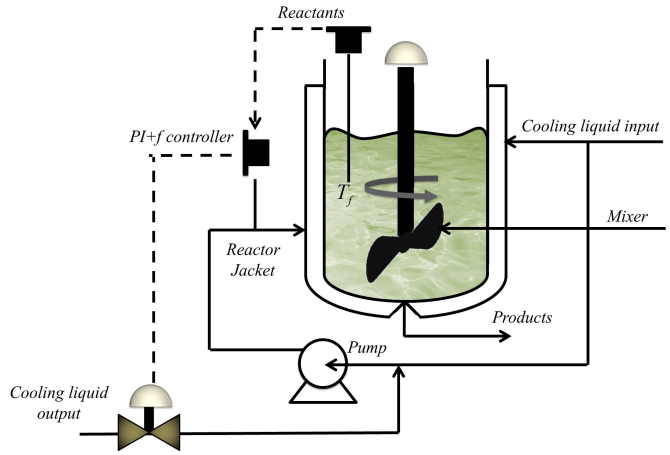


Fig. 2. Unstable continuously stirred tank reactor (CSTR).

The purpose of measuring the temperature in a reactant A is to manipulate the speed of the chemical reaction and, thus, to improve the selectivity of the reaction system. To remove the heat of the reaction, the reactor is surrounded by a jacket (coolant jacket) through which flows a coolant. Often, the heat transfer fluid is pumped through agitation nozzles that make the fluid circulate through the jacket at high velocity as you can see in Fig. 2.

From Bequette (2003) system parameters are listed in Table 1.

Differential equations of the process shown in Fig. 2 are characterized as follows:

(1) The balance of the mass on component A is,

$$V\frac{dC_A}{dt} = FC_{Af} - FC_A - V_{r_A}$$

with

$$r_A = k_0 e^{\frac{-\Delta E}{RT}} C_A$$

Table 1. CSTR parameters.

Description	Variables	Values	Unit
Volume of the reactor	V	85	ft^3
Activation energy	$-\Delta E$	32,400	$Btu/lbmol$
Heat transfer coefficient	U	75	$btu/hr^\circ F$
Heat of Reaction	$-\Delta H$	39,000	$Btu/lbmolPO$
Heat transfer area	A	88	ft^2
Frequency factor	k_0	$16.96x10^{12}$	hr^{-1}
Ideal gas constant	R	1.987	$Btu/lbmol^\circ F$
Volume cooling liquid in the chamber	V_j	21.25	ft^3
Heat capacity on reactor with density of the reagent	pc_p	53.25	$Btu/ft^3\,^\circ F$
Density of the cooling liquid with heat capacity of the cooling liquid	$p_j c_{pj}$	55.6	$Btu/ft^3\,^\circ F$

where r_A is the rate of reaction per unit volume, k_0 is the frequency factor, ΔE is the activation energy and R is the ideal gas constant, such that,

$$\frac{dC_A}{dt} = \frac{F}{V}\left(C_{Af} - C_A\right) - k_0 e^{\frac{-\Delta E}{RT}} C_A. \tag{17}$$

(2) Energy balance in the reactor,

$$\frac{dT}{dt} = \frac{F}{V}\left(T_f - T\right) + \frac{-\Delta H}{RT} k_0 e^{\left(\frac{-\Delta E}{RT}\right)} C_A - \frac{UA}{VpC_p}\left(T - T_j\right). \tag{18}$$

(3) Energy balance in the cooling liquid,

$$\frac{dT_j}{dt} = \frac{F_{jf}}{V_j}\left(T_{jf} - T_j\right) + \frac{UA}{V_j p_j C_{pj}}\left(T - T_j\right). \tag{19}$$

To get a linear representation of the CSTR model, the jacket temperature T_{jf} is considered as the manipulated (input) variable and the temperature of the CSTR T_j, as the controlled (output) variable. A tangent approximation will be obtained by considering the operating point (detailed descriptions of these parameters can be found in Bequette (2003)),

$$(C_{Af}^o, C_A^o, F^o, F_{jf}^o, T^o, T_j^o, T_f^o, T_{jf}^o) = (0.132, 0.066, 340, 28.75, 101.1, 55, 60, 50).$$

where C_A^o is the concentration of the product A, F^o is the concentration of the product A, F_{jf}^o is the flow cooling liquid (reactor jacket), T^o is the temperature in the Reactor, T_j^o is the temperature in the cooling liquid (reactor jacket), T_f^o is the Input temperature product A and T_{jf}^o is the cooling liquid temperature at the input-liquid.

The linear representation around the considered operating point it is obtained as,

$$\frac{T_j(s)}{T_{jf}(s)} = \frac{0.8714s + 6.963}{s^2 + 2.848s - 1.132}. \tag{20}$$

Notice that the linear system (20) is of the form given in equation (1) with $\tau = 0$ since it has an unstable pole, a stable pole and a zero . Using high quality sensors to determine the temperature (T_j) in CSRT reactors usually means high costs. This disadvantage in the sensor generates a dead-time in the measurement of the temperature, which creates a problem for the control strategy design, that increases its difficulty when the system is unstable. For that reason, it is possible to consider this effect by

adding an adequate time-lag to equation (20), that for this kind of reactor, the time–delay in the concentration measurement is approximately 0.3 hr. this produces,

$$\frac{T_j(s)}{T_{jf}(s)} = \frac{0.8714s + 6.963}{s^2 + 2.848s - 1.132} e^{-0.3s}. \qquad (21)$$

Taking into account equation (1), the system parameters (21) are $\gamma = 0.35$, $\delta = -3.20$, $\beta = 7.97$ and $\tau = 0.3$. From the stability condition shown in Theorem 1 and assuming that $m = 1$,

$$\tau = 0.3 < \frac{1}{\gamma} - \frac{1}{\delta} + \sqrt{\frac{1}{\gamma^2} + \frac{1}{\delta^2}} = 5.3$$

therefore, system (21) can be stabilized by a PI_f controller. To get the controller, the first step considers the free pole ϕ given by the PI_f which is located on the same position as zero β in order to cancel this dynamic. Then, $\phi = 7.97$. From equation (12), the range of values for the stabilizing gain $\bar{k_f}$ is $-2.21 < \bar{k_f} < 2.84$.

Considering $\bar{k_f} = 0.1176$, the gain $\bar{k_i}$ must be small enough to satisfy the conditions (10) and (11). For this case $\bar{k_i} = 0.022$.

From (15) and (16) the range of stabilizing gain $\bar{k_p}$ is given by $1.37 < \bar{k_p} < 14.06$. Considering $\bar{k_p} = 2$, the PI_f control is represented as:

$$PI_f = 0.23\left(1 + \frac{0.023}{s} + \frac{0.51}{s + 7.97}\right).$$

In order to illustrate that the Nyquist stability criterion has been satisfied, Fig. 3 shows the existence of counter-clockwise rotation to the point $(-1, 0j)$ in the Nyquist diagram.

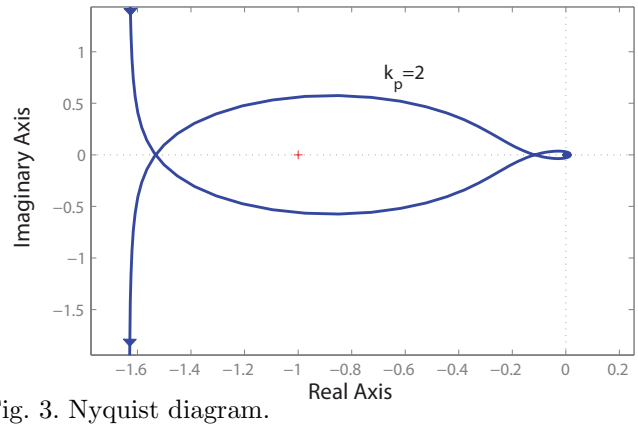


Fig. 3. Nyquist diagram.

Fig. 4 illustrates the control strategy performance regarding the reactor concentration when an unitary input reference is considered. A continues line shows the nominal system performance while a dotted line represents the system performance with a uncertainty of 28% in the time delay such that $\tau = 1.8$. In addition, the PI_f controller proves its effective disturbance rejection for step-type inputs with fast recovery when a disturbance occurs at $100hrs$. Finally, the control signal is illustrated in Fig. 5

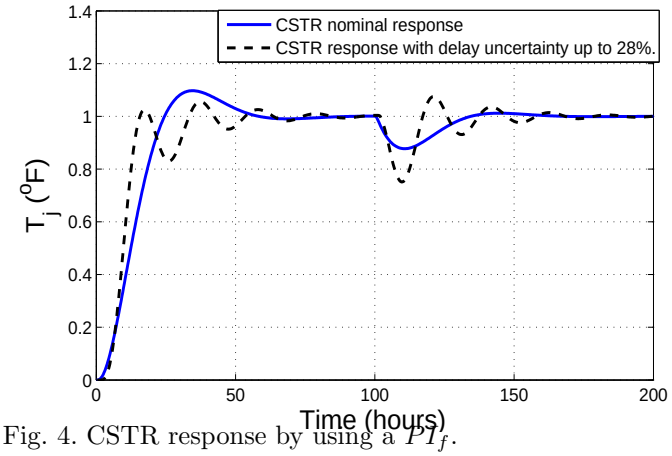


Fig. 4. CSTR response by using a PI_f.

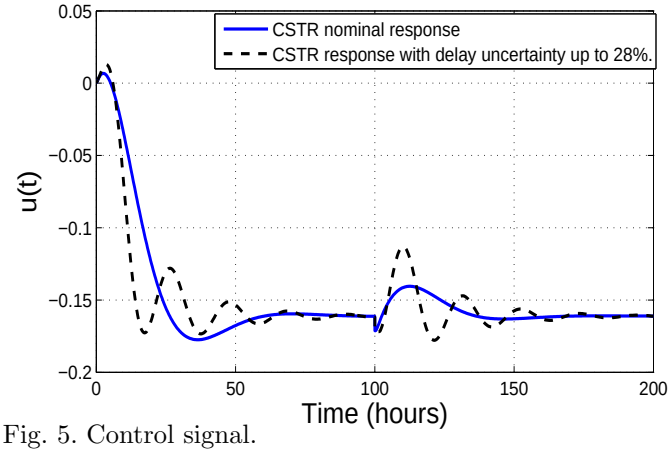


Fig. 5. Control signal.

5. CONCLUSION

This paper presents a sufficient condition for the stabilization of a high order unstable linear systme with time delay and one "stable" zero by using a modified-PI controller (PI_f). Additionally, the procedure to determine the parameters for the stabilizing gains k_p, k_i and k_f are given in order to provide an accurate PI_f controller tuning. The proposed scheme not only maintains the basic properties of a convencional PI controller, but also, the stability condition is improved allowing to stabilize systems with larger delays than the ones considered by the a conventional PI controller. Finally, an unstable continuously stirred tank reactor was used to verify the performance of the proposed strategy using a numerical simulation.

REFERENCES

Bequette, B.W. (2003). *Process Control. Modelling, Design and Simulation.* Prentice-Hall Internacional.

Franklin, G.F., Powell, .D., and Emami-Naeini, A. (1995). Feedback control of dynamic systems. *Addison Wesley Publishing Company, 3er edition.*

Gu, K., Kharitonov, V.L., and Chen, J. (2003). *Stability of Time-Delay Systems.* Birkhäuser.

Hu, T. and Lin, Z. (2001). *Control Systems with Actuator Saturation: Analysis and Design.* Birkhauser, Boston, MA.

Huang, H. and Chen, J. (1997). On stabilizing a time delayed unstable process. *Journal of the Taiwan Institute of Chemical Engineers*, 28(4), 289–299.

Hwang, C. and Hwang, J.H. (2004). Stabilisation of first-order plus dead-time unstable processes using pid controllers,. *Proceedings Control Theory and Applications*, 151(1), 89–94.

Kawnish, K. and Choudhury, M.S. (2012). A novel dead time compensator for stable processes with long dead times. *Journal of Process Control*, 22(3), 612–625.

Lee, S.C., Wang, Q.G., and Xiang, C. (2010). Stabilization of all-pole unstable delay processes by simple controllers. *Journal of Process Control*, 20(2), 235 – 239.

Liu, T., Zhang, W., and Gu, D. (2005). Analytical design of two-degree-of-freedom control scheme for open-loop unstable processes with time delay. *Journal of Process Control*, 15, 559–572.

Munz, Ebenbauer, C., Haag, T., and Allgower, F. (2009). Stability analysis of time-delay systems with incommensurate delays using positive polynomials. *IEEE Trans. Autom. Control*, 54(5), 1019 – 1024.

Palmor, Z.J. (1996). Time-delay compensation smith predictor and its modifications. *The control handbook*, 224–237.

Richard, J.P. (2003). Time-Delay Systems:An Overview of Some Recent Advances and Open Problems. *Automatica*, 39, 1667–1694.

Seshagiri, R.A., Rao, V.S.R., and Chidambaram, M. (2007). Simple analytical design of modified smith predictor with improved performance for unstable first-order plus time delay (fodtp) processes. *Ind. Eng. Chem. Res*, 46(13), 4561–4571.

Shamsuzzoha, M., Jongpal, J., and Moonyong, L. (2009). Improved analytical pid controller design for the second order unstable process with time delay. *17th European Symposium on Computer Aided Process Engineering*, 8, 1107–118.

Silva, G.J., Datta, A., and Bhattacharyya, S.P. (2005). *PID controllers for time-delay systems.* Springer, London.

Sipahi, R., Niculescu, S.I., Abdallah, C., Michiels, W., and Gu, K. (2011). Stability and stabilization of systems with time delay. *Control Systems, IEEE*, 31(1), 38–65.

Smith, O.J.M. (1957). Closer control of loops with dead-time. *Chem. Eng. Prog.*, 53(5), 217–219.

Trentelman, L., Stoorvogel, A., and Hautus, M. (2001). *Control Theory for Linear Systems.* Springer-Verlag, 1 edition.

Wang, Q.G., Lee, T.H., and Tan, K.K. (1999). Finite spectrum assignment for time- delay systems. *Lecture Notes in Control and Information Science*, (239).

Xian, L., Yong-Sheng, Y., Quing-Guo, W., and Wei-Xing, Z. (2005). A double two-degree-of-freedom control scheme for improved control of unstable delay processes. *Journal of Process Control*, 15, 605–614.

Xiang, C., Wang, Q., Lu, X., Nguyen, L., and Lee, T. (2007). Stabilization of second-order unstable delay system by simple controllers. *Journal of Process Control*, 17(8), 675–682.

Navigation Assistance System for the Visually Impaired People Using the Modified Fictitious Force Algorithm

Sánchez B. David. Ortega H. Christian.
Sotomayor O. Nelson, Chávez G. Danilo

Escuela Politécnica Nacional
Departamento de Automatización y Control Industrial, Ladrón de Guevara E11-253,
Quito, Ecuador

(e-mail: david.sanchez@epn.edu.ec, christian.ortega01@epn.edu.ec,
nelson.sotomayor@epn.edu.ec, danilo.chavez@epn.edu.ec)

Abstract: This work is focused on the implementation of a prototype that assists, complementary to the white cane, in the navigation of blind people by detecting obstacles with artificial vision. The navigation system was developed based on the Kinect sensor coupled to a helmet, which in addition to providing a color image of the environment, it is also capable of delivering a depth image of the same; this information was processed in a computer and based on the modified fictitious force algorithm by Scaglia et al. (2009), an obstacle free route was determined, which is transmitted to the user through audible messages; furthermore a complementary vibration system was implemented and mounted on a vest, which will serve as backup in order to alert the person about near obstacles. The prototype also has an identification and decoding system for QR codes that helps for a better orientation of the visually impaired person within the environment in which they manage themselves.

Keywords: Visual disability, vision, fictitious force, depth, navigation.

1. INTRODUCTION

Nowadays people deprived partial or totally of the sense of sight are limited in their mobilization for the development of several important aspects of their lives. Mobilizing involves a risk because of the obstacles that stand in their way, therefore several devices have been developed that help these people in their navigation; especially in places where there are suspended objects, as these cannot be detected by the white cane. Unfortunately, access to these devices is limited due to their high cost and lack of information about the existence of them.

INEC (2011) underline that the limited visual ability of individuals influences their lives in several important aspects such as social, academic and occupational. Navigation for these people either in work or in a domestic environment implies a big challenge because of the many obstacles that threaten their physical integrity, and the inadequacy of the cities for the blind; so, most of them should be guided by a walking stick, a guide dog, a relative or a friend.

Through this prototype and using the Microsoft Kinect tool, an assistance system for the blind was developed to complement the white cane in order to allow them to safely navigate through the detection and avoidance of obstacles, which they could collide with; especially objects that are not at floor level. In addition, more information on the environment that surrounds the visually impaired person is provided by detecting certain visual messages that could be located at specific points of importance, such as an information desk or an emergency exit.

The work is divided as stated below: Section 2 presents the main characteristics of the used sensor, section 3 describes the implemented prototype, section 4 shows the acquisition and processing of the delivered image by the Kinect sensor to set the initial navigation parameters, section 5 details how the modified fictitious force algorithm was implemented using artificial vision for obstacle avoidance, section 6 explains about the QR codes detection and their functionality within navigation, section 7 is about the generation and emission of audible instructions, while in section 8 system operation and interaction vibration button are explained. In Section 9 experimental results for people with and without visual impairment are shown.

2. MICROSOFT KINECT SENSOR

The Microsoft Kinect sensor for windows was used in the color image acquisition, which is used to find defined pattern matching within the entire image that determine the existence of a two-dimensional bar code (QR Code); as well as the depth image, which delivers depth information to digitized objects within the image and based on this, perform the necessary calculations of angles and required distances for the development of the implemented algorithm.

The Fig. 1, show the most important technical specifications about the Kinect for Windows for example, the minimum or maximum distance visible into the range the navigation or the horizontal and vertical field of view, some of the most important characteristics used into the algorithm.

Feature	Kinect for Windows 1	Kinect for Windows 2
Color Camera	640 x 480 @30 fps	1920 x 1080 @30 fps
Depth Camera	320 x 240	512 x 424
Max Depth Distance	~4.5 M	~4.5 M
Min Depth Distance	40 cm in near mode	50 cm
Horizontal Field of View	57 degrees	70 degrees
Vertical Field of View	43 degrees	60 degrees
Tilt Motor	yes	no
Skeleton Joints Defined	20 joints	26 joints
Full Skeletons Tracked	2	6
USB Standard	2.0	3.0
Supported OS	Win 7, Win 8	Win 8
Price	$299	TBD

Fig. 1. Technical Specifications.

2.1 Depth Sensor

Display a three-dimensional image and provide depth values are two of the main features of the sensor and are precisely those that differentiate it from other devices.

Kean et al. (2011) describe that the sensor has two depth ranges. The default range and near range, which is only available on the Kinect for Windows sensor, as shown in Fig. 2.

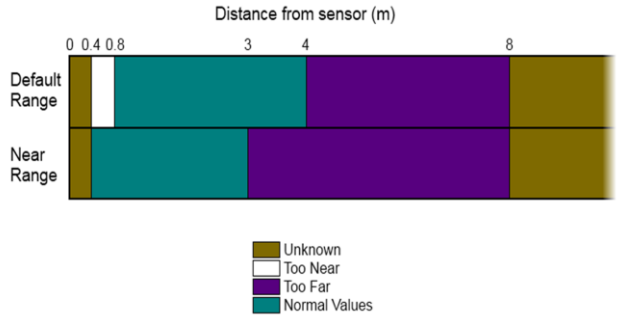


Fig. 2. Microsoft (2014) shows Kinect sensor depth ranges.

2.2 Limitations

Kinect sensor has several limitations that make that depth of certain regions of the scene cannot be estimated or if estimated, the data reliability is not acceptable by Salvatore (2014).

One of the great problems is caused by the infrared projection points because when they impact on an object, they generate a shadow in another on a greater distance, as seen in Fig. 3. The result is that depth in the affected areas by these shadows cannot be determined. This manifests as zero value pixels ("black zones") in the depth image as underlined by Córdova (2013).

Besides sensor's own internal limitations, limitations also exist by external factors, in particular because of the objects' surface properties, such objects may include translucent objects, reflective objects, black color objects, concave objects or reflective cavities. A strong brightness change (sunlight) is another limitation since sunlight, also being composed of infrared rays, prevents the sensor's infrared beam itself to be detected by the IR camera.

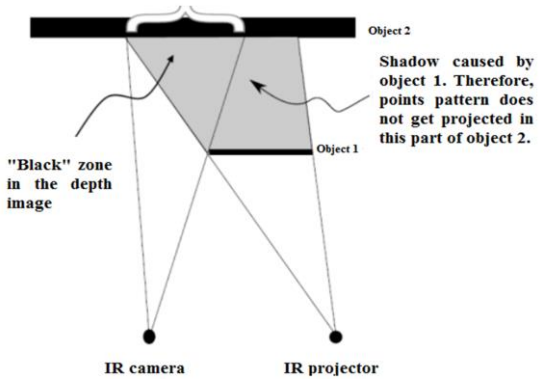


Fig. 3. Limitation due to projected shadows.

3. BLOCK DIAGRAM

The principal block diagram about the application is shown in the Fig. 4.

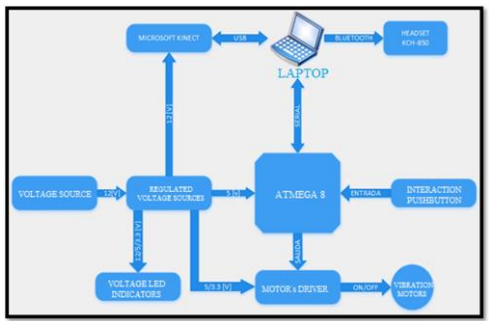


Fig. 4. Block Diagram.

The main processor in the application is the laptop , which receives depth data issued by the Kinect sensor that are necessary for the generation of repulsive forces of the obstacles and the attractive force of the user so that it can implement the algorithm modified fictitious force. All the mathematical development for the algorithm is implemented within the laptop reason why the bluetooth communication is used by a headset to communicate obstacle free route to the visually impaired person. Serial communication between the laptop and microcontroller is used to activate the vibration motors located in the vest, in addition, this communication is used to transmit user requirements through interaction button. All electronic elements are fed according to the technical specifications of the manufacturer.

4. IMAGE ACQUISITION AND PROCESSING OF THE KINECT SENSOR

The overall program structure is based on the acquisition and processing of both, color and depth image, which are delivered by the Kinect sensor. Data is processed in the LabVIEW 2013 Software, which through serial

communication transmits information to a microcontroller in order to enable or disable vibration motors; as well as receive and transmit the actions performed by the user via an interaction button. This button enables or disables the voice instructions issued by the developed system or issues the system's use instructions.

Kinect sensor initialization was performed using the toolkit 'Microsoft Kinect API v1.0' and its own LabVIEW library, where the inclination angle, that should have the sensor based on the user's height, is calculated.

In order to obtain the maximum depth value that the Kinect sensor can visualize a triangle between points ABC is formed as shown in Fig. 5, it is also used to estimate a user's height, which corresponds to a leg of the triangle in mention. Furthermore, it is established through testing and error that the proper distance as the second leg should be approximately 2 m, the calculation of an ideal depth (*D_IDEAL*) is performed using the Pythagorean Theorem.

This value of ideal depth is not applicative since it shows the ground as an obstacle to avoid when that's not the truth, thus it was decided to find a desired depth value (*D_WISHED*) that will show the obstacles to be avoided limited to a minimum height (h '). Based on several tests and because it is a complementary system to the white cane, it is determined that the minimum height of the obstacle to be avoided is 35% of the approximate user's height value, which is calculated by the system. To find the desired depth value, a similarity between the A'BC' and ABC is established achieving to eliminate the safety depth measure (D_SEC), the resulting expression is shown in (1)

$$DMÁX = D\_WISHED = D\_IDEAL\left(\frac{h - h'}{h}\right) \quad (1)$$

Based on (1) it is determined that desired depth is 65% of the ideal depth. Establishing this depth as the maximum threshold value (THR_MÁX) that the Kinect sensor can visualize in its environment.

Kean et al. (2011) underline that the minimum established threshold in the Kinect sensor is specified by its limited permissible depth acquisition, which is 45cm in near mode, therefore the value of THR_MIN is equal to this distance.

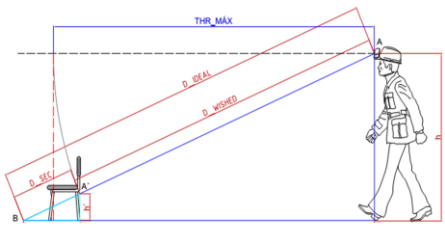


Fig. 5. Getting the maximum depth of vision.

6. ALGORITHM IMPLEMENTATION OF FICTITIOUS MODIFIED FORCE BY ARTIFICIAL VISION

The modified fictitious force algorithm will determine an obstacle free route where the visually impaired person can navigate without the possibility of getting impacted by any objects around them. To fulfill this purpose, there are required the following input data to the system: The amount of digitized objects in the sensor's image processing and the distances that these are from the person.

For the algorithm implementation it is necessary to establish a coordinate axes system that is considered for the mathematical and geometric algorithm development, as well as generate the repulsive force emitted by the objects in the environment. It is also necessary to establish an attractive force that is always attached to the visually impaired person within their navigation and finally, these forces will relate with a vector sum, generating a final vector containing the rotation direction necessary to avoid the impact of the person with the obstacles, as Ribeiro (2005) indicate.

5.1 Coordinate axes System.

The established coordinate axes system will remain invariant during the development of the algorithm throughout the mathematical processing of the navigation system and it is attached to the visually impaired person as shown in Fig. 6.

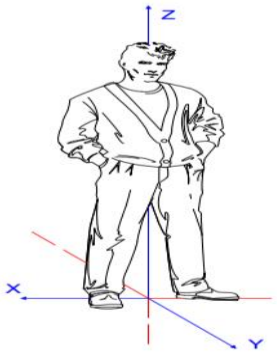


Fig. 6. Coordinate axes System.

5.2 Repulsive Force Generation.

In order to generate a repulsive force emitted by the obstacles, force shall be understood as a vector and therefore it has characteristics such as a direction (incidence angle) and a module, which is inversely proportional to the distance between the person and the obstacle's center of mass. The mathematical expression showing the magnitude calculation for the repulsive force $|\overrightarrow{Fr}|$ is (2).

$$|\overrightarrow{Fr}| = \begin{cases} 0, & si\ Pmed > Pmáx \\ \frac{Pmáx - Pmed}{Pmáx - Pmín}, & si\ Pmed < Pmáx \end{cases} \quad (2)$$

Where:

Pmáx = Maximum vision depth set in the Kinect sensor.

Pmed = Measured depth between the obstacle and the visually impaired person.

Pmín = Minimum vision depth set in the Kinect sensor.

As it can be seen in (3), the repulsive force magnitude is null when objects are outside the range of vision, as well as an increase in module is seen as the obstacle gets closer to the person. Finally, magnitude is greater than 1 when the distance to the obstacle is less than the depth of investment forces which warrants immediate evasion action.

$$|\overrightarrow{Fr}| = \begin{cases} 0, & si\ Pmed > Pmáx \\ (0,1], & si\ Pinv \leq Pmed < Pmáx \\ > 1, & si\ Pmed < Pinv \end{cases} \quad (3)$$

Where:

Pinv = Depth set for the investment of the final evasion force.

The angle theta (θ) shown in Fig. 7, is defined by (4), mathematical expression that relates the data of horizontal vision angle of the Kinect sensor, the missing value of 61° to complete the reference to the *x* positive axis and the *x* coordinate of each object's center of mass. An example of the above is shown in Fig. 8. It should be noted that the value of 0.0890625 is the relation between pixel and maximum horizontal vision angle of the Kinect sensor.

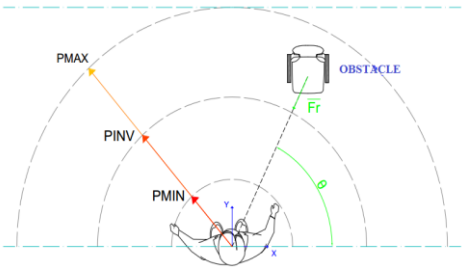


Fig. 7. Generation of a fictitious repulsive force from the obstacle to the visually impaired person.

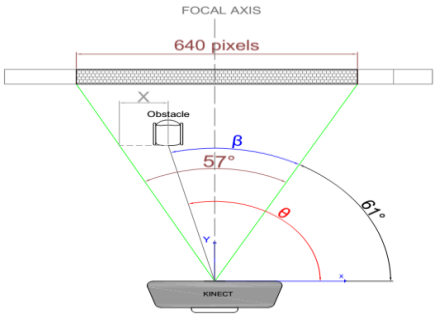


Fig. 8. Repulsion angle generated by the obstacle.

$$\theta = \big((640 - x)(0.0890625)\big) + 61 \quad (4)$$

Where:

x = Horizontal position of the digitized obstacle's center in the navigation environment.

5.3 Attractive Force Generation.

An attractive force from mobile robots algorithms is one that drives the robot to reach a certain point in its trajectory called 'goal', as mentioned by Ribeiro (2005).

Adapting this definition for the visually impaired person, such force is mentioned as a vector that is attached to the user, with a direction in the *y* positive axis and with a constant magnitude equal to 1. This force is responsible for generating the movement towards in the person's trajectory. Fig. 9 shows the attraction force ($\overrightarrow{Fa}$) that the user generates on their route.

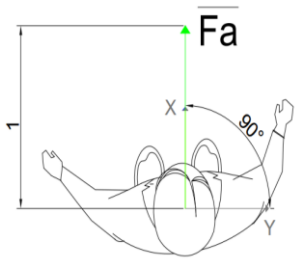


Fig. 9. Attractive force generated in the user.

5.4 Final Rotation Force.

The attraction and repulsion forces obtained above are related in order to calculate a specific evasion angle, which represent the direction of an obstacle free route within the digitized environment by the Kinect sensor.

For obtaining this angle, vector addition is applied by the polygon method, in this case vectors are all fictitious forces generated by both the obstacles and the user, thus a final resultant force ($\overrightarrow{Ff}$) is obtained with a module and angle value, referred to the *x* positive axis.

Based on the critical depths established as the Kinect sensor vision, the following consideration can be seen: if the obstacle is within the region between the investment depth forces ($Pinv$) and the Kinect's minimum vision depth ($Pmín$), the vector's magnitude will have a greater value than the unit. To calculate the final repulsive force magnitude when two or more objects are digitized by the Kinect sensor in the environment, (2) is used, with the difference that the measured depth ($Pmed$) is set as the average of all obstacles distances calculated from the visually impaired person, also leaving this magnitude value escaladed as shown in (3) and Fig. 10.

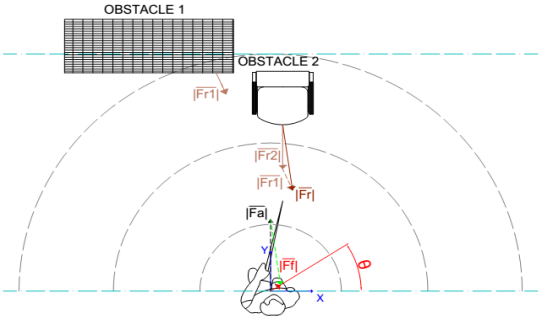


Fig. 10. Final force obtaining in an environment with the presence of two obstacles, vector sum by the polygon method for repulsive vectors and subsequent vector sum by the polygon method with the attraction force.

6. TWO-DIMENSIONAL BARCODE IDENTIFICATION

Two-dimensional barcodes identification, including the so-called QR codes, has its use in this work when it comes to suppress the Kinect sensor limitations. The designed algorithm for QR codes detection is comprised in 4 stages, which allow to optimize code reading and reproduction of the same in the user's navigation. The first stage will be responsible for Kinect's RGB image processing into a grayscale image. The second analyzes and locates search patterns (small squares in 3 of the 4 corners of a QR code) across the vision range. The third stage is responsible for decoding the hidden message within the limited area by the search patterns as in Fig. 11. Lastly, there is the stage of the audible messages play, which has a memory system that prevents repetitive playback of a same visualized code more than once within a time interval.

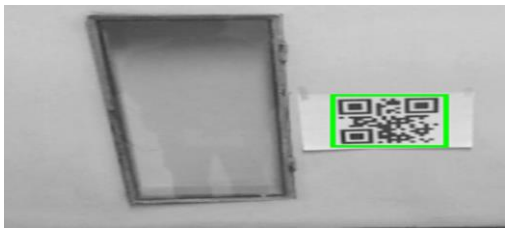


Fig. 11. QR code reading.

7. TRANSMISSION OF INSTRUCTIONS AND VEST.

7. 1 Audible messages generation as control instructions.

Audible messages are the control actions that are delivered to the visually impaired people, which will be emitted by the main processor (computer) via a bluetooth headset.

As shown in Fig. 12, control actions will be issued to the visually impaired person according to the region where the final rotational force is oriented. If the evasion angle is from 85° to 95°, an instruction to keep forward without altering their direction will be emitted. If the angle is between 260° and 280°, it means that the obstacle is in front of the user and they are forced to stop in their trajectory, then they must evaluate the obstacles existence with a head movement so they can know which route to be heading.

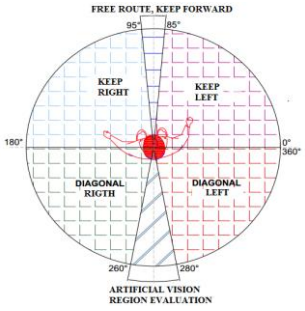


Fig. 12. Evasion angle interpretation for control instructions.

7.2. Switching and vibration interaction system

This system consists of vibration motors and a button, both placed in the user's vest. Motors' objective shown in Fig. 13 is to give an emergency notice, supported by the sense of touch against a possible collision with objects, while the purpose of the button is to transmit the system any desired requirement by the user.

A button that generates an external interrupt to the microcontroller was used, so it allows the user to have an interaction with the navigation system. Its main features are: enable or disable navigation instructions through an instantaneous pulse; as well as emitting the use or interpretation instructions of each of the audible messages that the person will hear through a long pulse (approx. 4 s).

Fig. 13. Vibration motors and interaction button location.

8. PROTOTYPE DESCRIPTION

The prototype consists of six main parts:

- A white industrial helmet with all necessary adaptations to incorporate the Kinect sensor.
- Sensor Kinect, which is responsible for depth image acquisition.
- Mini Headset KCH-850, is the device by which the user will hear all audible messages issued by the system.
- Control, communication and power supply regulation module; which is mainly the system responsible of communication between the computer and the microcontroller Atmega8.
- Power supply, which is a LiPo battery in charge of supplying power to the entire system.
- Vest with all fastenings and necessary compartments to incorporate all electronic, vibration, pulsation and power systems.

Due to the location of the Kinect sensor (helmet), the system focuses on the avoidance of obstacles that are above the waist, reaching a reliable navigation with 60cm high objects and 1.8m away, which is enough because the project is a complementary system, not a substitute to the white cane.

9. EXPERIMENTAL RESULTS

Tests on the prototype with visually impaired people were performed using the system with and without the white cane; as the prototype is a complementary system.

9.1 Obstacle detection Ranges.

These tests were performed in order to validate the calibrated distances as a range of maximum and minimum depth necessary for obstacles visualization within a given space as shown in Fig. 7. The minimum depth that the sensor can visualize is 45 [cm] due to Kinect's self-limitation, the maximum depth will be set by the algorithm that calculates the inclination and the sensor's maximum depth based on the user's approximate height. Table 1 shows a sampling for different heights and depth ranges established by the Kinect sensor.

Table 1. Maximum and minimum depth ranges

User's approximate height [m]	Minimum Depth			Maximum Depth		
	Calculated [cm]	Real [cm]	Error [%]	Calculated [m]	Real [m]	Error [%]
1.55	45	46.2	2.66	1.64	1.66	1.21
1.66	45	45.8	1.77	1.68	1.70	1.19
1.74	45	46.5	3.33	1.72	1.71	0.58
1.78	45	46.0	2.22	1.74	1.74	0
1.82	45	46.2	2.66	1.75	1.77	1.14

9.2 Obstacles detection with a minimum established height.

These tests were developed in order to validate the navigation system's ability to detect obstacles that have as minimum height the established relation between the user's height and the maximum vision depth that is set in the Kinect sensor. Table 2 shows the obtained results for this design condition verification.

Table2. Obstacle detection with minimum height

User's approximate height [m]	Obstacle's minimum height [m]	Samples	Desk table 80 [cm]		Trash can 60 [cm]	
			Detected	Error [%]	Detected	Error [%]
1.55	0.54	5	5	0	5	0
1.66	0.58	5	5	0	5	0
1.74	0.60	5	5	0	2	60
1.78	0.62	5	5	0	1	80
1.82	0.64	5	5	0	0	100

9.3 Results with visually impaired people

These tests were performed by 6 people (4 women and 2 men) of different ages in the range of 30 to 56 years old and with three different types of visual impairment:

- Low vision.
- Total blindness acquired.
- Total blindness of birth.

Tests with the white cane were performed by 3 people with total visual impairment, and these were held in different structured and unstructured environments both indoors and outdoors. Halls, a parking lot, a backyard and an auditorium were chosen.

System's operation was verified in narrow roads such as corridors, where the system tends to make the blind person moves in a oscillating way because evasion depends entirely on the interpretation of the instruction by the user and the inclination taken by them. Fig. 14 shows an example of a person with total acquired visual disability moving in an oscillating way.

The usefulness of vibration backup system was reflected, as when the system allowed the user to get too close to the walls (less than 1m), vibration alerted him to stop or lower his speed until another voice instruction is emitted by the system.

Fig. 14. Navigation of a visual impaired person in a narrow corridor.

The decrease in the user's speed significantly influenced in the results since the system could emit instructions that were more efficient, thus oscillating displacement in narrow aisles decreased.

People with a more developed sense of orientation as in the case of people with total blindness of birth, assimilated better the system's instructions and therefore their navigation was more efficient; however, like blind people with some level of acquired visual impairment, they decreased their speed when they didn't have their white cane's help, which helped to a better navigation.

10. CONCLUSIONS

- The implemented prototype is not intended to replace the white cane, but rather being a complementary system that assist the visually impaired people in their navigation, so they do not crash with obstacles that cannot be detected by the white cane and represent a potential danger, such as tall or hanging objects.
- The implementation of the modified fictitious force algorithm helps to despise supposed obstacles that the camera vision manifests in the user's trajectory, such as the ground, where any evasion action should be emitted.
- Detection and especially instruction for obstacles avoidance in the environment depend largely on the speed with which the user walks. The system has a higher efficiency when the person moves at a slower pace as it allows the system to end a voice instruction before moving on to the next.

- To cover with more accuracy and efficiency the environment that surrounds a blind person, it was determined that it is not sufficient to use a single sensor as the system's feedback since areas, where the presence of another sensor can contribute to the evasive action, are neglected.

- The vibration system incorporated into the vest has a great importance in the prototype's development, since along with the white cane, it manifests as the last safety resort to alert the user about the presence of a near obstacle.

REFERENCES

INEC. (2011). *2010 Census Results*. [Online]. Available: www.ecuadorencifras.gob.ec/resultados/.

Kean, S., Hall, J., and Perry, P. (2011). *Meet the Kinect*, New York: Apress.

Microsoft. (2014). *Coordinate Spaces*. *Microsoft*. [Online]. Available: https://msdn.microsoft.com/en-us/library/hh973078.aspx.

Córdova, F. (2013). *Theft/abandonment detection of indoors objects using depth cameras*. [Online]. Available: http://arantxa.ii.uam.es/~jms/pfcsteleco/lecturas/20121212FabricioACordovaLucero.pdf.

Scaglia, G. (2009). *Mobile robot navigation in unstructured environments using linear algebra*. Madrid: RIAI.

Ribeiro, M. Isabel (2005). *Obstacle Avoidance*. [Online]. Available:http:user.isr.ist.utl/~mir/pub/ObstacleAvoidance.pdf.

Salvatore, J., Osio, J., Morales, M (2014). *Object detection using Kinect sensor*. LACCEI.

Neural Control for Photovoltaic Panel Maximum Power Point Tracking

Martin J. Loza-Lopez * **T.B. Lopez-Garcia** *
Riemann Ruiz-Cruz ** **Edgar N. Sanchez** *

* *Centro de Investigación y Estudios Avanzados del IPN, Zapopan, Jal 45030 México, (e-mail: {mdloza,tblopez,sanchez}@ gdl.cinvestav.mx)*
** *ITESO University, Periferico Sur Gomez Morin 8585, Tlaquepaque, Jalisco, México C.P. 45604 ,(e-mail: riemannruiz@iteso.mx)*

Abstract: With the rise in the use of renewable energies, solar panels have proven to be reliable and have a favorable cost-benefit ratio, producing energy free of noise and air pollution. Solar panels are subject to considerable variations in working conditions due to changes in solar irradiation levels and temperature, that affect its semiconductor properties. To be able to profit as much as possible from this source of energy, control of the modules and perturbation rejection is very important to obtain the highest viable amount of electrical power. This work is concerned with the on-line identification and control of a photovoltaic system using neural networks. Having on-line identification and control allow the system to be more adaptable to changes in weather and other variations than with common off-line methods.

Keywords: Photovoltaic systems, solar energy, high order neural networks, Kalman filter, maximum power point tracking.

1. INTRODUCTION

This work is concerned with the application of recurrent high order neural networks to design a robust photovoltaic panel model and to control the output voltage of the panel (V_{PV}) to obtain the maximum power available.

It is important to understand the impact that different uncertainties and parameter variations have on the mathematical model of solar panels, since their parameters are so dependent on weather conditions. In the field of neural networks there are several studies that focus on the characterization and modeling of solar panels based on artificial intelligence, as can be seen in Hadjab et al. (2012). Another popular use of artificial neural networks is that of designing maximum power point trackers for solar photovoltaic (SPV) modules, as can be observed in Zhang and Bai (2005), Ramaprabha et al. (2011) and Alabedin et al. (2011). The methods applied in these studies are fuzzy logic controllers, genetic algorithms, and radial basis functions; and they are usually off-line methods.

In order to solve the problem of time-varying parameters and uncertainties, in this paper the on-line identification and control is proposed. The maximum power point (MPPT) is obtained by means of a searching algorithm; then the system is controlled to track the MPPT using a neural network to identify the model on-line. Then a controller which regulates the switching frequency of an insulated-gate bipolar transistor (IGBT) in the DC-DC buck converter, based on the identified model, is developed.

The advantage of this method is that the model is not greatly affected by the high frequency noise created by the IGBT and other perturbations, and so it is possible to reduce the use of filters. A conventional maximum power point tracker *perturb and observe* is used to find the output voltage of the solar panel (V_{PV}) necessary to have the maximum electrical power.

This paper is organized as follows. In section 2, mathematical preliminaries are given, including a review of photovoltaic systems, neural networks and the Kalman filter. In section 3, the neural control is presented, starting with the model of the DC-DC buck converter, and continuing with the identification and control design. In section 4, the results are validated using Simulink's Simscape Power Systems Blocks [1], and a comparison of the neural controller developed with a discrete sliding modes controller is shown. Finally, in section 5, the conclusions are presented.

2. MATHEMATICAL PRELIMINARIES

2.1 MPPT algorithm applied to photovoltaic systems

Photovoltaic systems use solar cells to capture solar energy and convert it into electricity. These systems are generally made from modified silicon and other semiconductor materials, they are usually long lasting (25 to 30 years); with the advance of technology there has been a rise in variety of manufacturers and models available, for a lower price.

Solar panels can me modeled using an equivalent circuit which consists of a current source I_{cc} (whose value in amperes depends on the irradiance at the moment of measurement), a diode for discharge, and two resistors; one of them represents losses due to bad connections (R_s)

[1] Simulink/Simscape Power Systems are trademarks of The MathWorks, Inc.

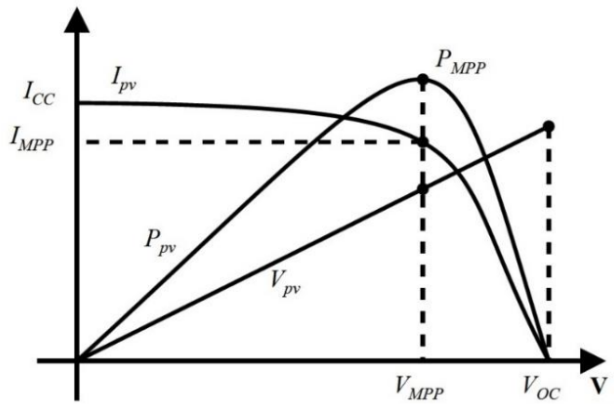


Fig. 1. VI graph of solar panel

and the other represents the leakage current from the capacitor (R_{sh}). The equation that defines the behavior of said equivalent model is (Cubas et al., 2014):

$$I_{PV} = I_{cc} - I_o(e^{\frac{q(V_{PV}+I_{PV}R_s)}{nkT}} - 1) - \frac{V_{PV} + I_{PV}R_s}{R_{sh}} \quad (1)$$

where k is the Boltzman constant, T is the absolute temperature in the photovoltaic panel, I_o is the inverse saturation current of the diode, q is the charge of the electron, n is the ideality factor of the diode.

From (1), it is clear that there exists a relationship between the voltage and the current in the photovoltaic panel. This relationship can be observed in Fig. 1, which shows the existence of a unique maximum power point, P_{MPP}, for each solar panel depending on the temperature and irradiance at the moment of measurement.

To be able to successfully follow the maximum power point of the solar panel, it is necessary to have a reference voltage which corresponds to that point, and to design a controller to track that reference voltage; this is commonly known as a maximum power point tracker (MPPT). In the literature, there are several algorithms that have been developed for this purpose, based on neural networks, incremental conductance, fuzzy logic, etc, (Esram and Chapman, 2007). In this paper, the MPPT algorithm used is known as the *perturb and observe* method which can be implemented in real-time, and it is one of the algorithms most commonly used for this purpose. This algorithm is based on the following criterion: the voltage of the solar panel is perturbed and if for this new value the power obtained has been incremented, then a change in that direction will be spurred; if on the contrary, the new power value has decreased, a new perturbation will be realized in the opposite direction.

The next step is to manipulate the voltage of the panel (V_{PV}) to track the voltage generated by the algorithm (V_{MPP}), this is accomplished by using a DC-DC Buck converter, discussed in section 3, which forces the electrical output power of the solar panel to reach the desired value.

2.2 Recurrent High Order Neural Networks (RHONN)

In the field of neural networks, usually k denotes a sampling step, where $k \in 0 \cup \mathbb{Z}^+$. Also considering the traditional definitions of $|\cdot|$ as the absolute value and $\|\cdot\|$ as an adequate norm for a vector or matrix. Considering a MIMO nonlinear system (Zhang and Bai, 2005):

$$x_{k+1} = F(x_k, u_k) \quad (2)$$

$$y_k = h(x_k) \quad (3)$$

where $x \in \mathbb{R}^n, u \in \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a nonlinear map. For (2), u is the input vector, it is chosen as a state feedback function of the state:

$$u_k = h(x_k)$$

Substituting this in (2) to obtain an unforced system:

$$x_{k+1} = F(x_k, h(x_k)) = \tilde{F}(x_k). \quad (4)$$

Defining a discrete-time recurrent high order neural network (Alanis et al., 2007):

$$\hat{x}^i_{k+1} = (w^i)^T z^i(\hat{x}_k, u_k), i = 1, ..., n \quad (5)$$

where $\hat{x}^i$ is the sate of the i-th neuron, n is the state dimension, w^i is the respective on-line adapted weight vector, and $z^i(\hat{x}_k, u_k)$ is given by:

$$z^i(\hat{x}_k, u_k) = [z^{i1}...z^{iL_i}]^T = [\prod_{j\in I_1} \psi_{ij}^{dij(1)} ... \prod_{j\in I_{L1}} \psi_{ij}^{dij(L_1)}]^T \quad (6)$$

where L_i is the respective number of high order connections, $I_1, I_2, ..., I_{L_i}$ is a collection of non-ordered subsets of $1, 2, ..., n$, and ψ_i is given by:

$$\psi_i = [S(\hat{x}_1)...S(\hat{x}_n)u_1...u_m]^T \quad (7)$$

where $S(\cdot)$ is defined as a logistic function.

Assuming that the system (2) is observable, it is approximated by the discrete time RHONN parallel representation (Rovithakis and Christodoulou, 2000) :

$$x^i_{k+1} = (w^{i*})^T z^i(x_k, u_k) + \epsilon^{zi} \quad (8)$$

where x^i is the i-th plant state, ϵ^{zi} is a bounded approximation error, which can be reduced by increasing the number of adjustable weights (Rovithakis and Christodoulou, 2000) .

Assuming that there exists an ideal weight vector w^{i*} such that the norm of the approximation error can be minimized on a compact set $\Omega^{zi} \subset \mathbb{R}^{L_i}$. The ideal weight vector w^{i*} is used only for analysis, assuming that it exists and is an unknown constant (Rovithakis and Christodoulou, 2000) . Defining the estimate of the weight as w_i and the estimation error as:

$$\tilde{w}^i_k = w^{i*} - w^i_k \quad (9)$$

Since w^{i*} is assumed to be a constant, the next expression is true:

$$\tilde{w}^i_{k+1} - \tilde{w}^i_k = w^i_k - w^i_{k+1} \quad (10)$$

2.3 Kalman Filter

The Kalman filter (KF) estimates the state of a linear system with additive state and output white noises (Feldkamp, L.A., Feldkamp, T.M., Prokhorov, 2001; Brown

and Hwang, 1997; Song and Grizzle, 1992) . For KF-based neural network training, the network weights become the states to be estimated. The error between the neural network output and the measured plant output is considered to be additive white noise. Since the neural network mapping is non linear, an extended Kalman filter (EKF) is applied (Sanchez, E. N., Alanis A.Y., 2004) . The goal of the training is to find the optimal weight values that minimize the prediction errors. In this paper, a EKF-based training algorithm is used, described by:

$$\begin{aligned} w_{k+1}^i &= w_k^i + \eta_i K_k^i e_k, i = 1, ..., n \\ K_k^i &= P_k^i H_k^i M_k^i \\ P_{k+1}^i &= P_k^i - K_k^i (H^i)^T P_k^i + Q^i \end{aligned} \tag{11}$$

with:

$$\begin{aligned} M_k^i &= [R^i + (H^i)^T P_k^i H_k^i]^{-1} \\ e_k &= y_k - \hat{y}_k \end{aligned}$$

where $e_k \in \mathbb{R}^p$ is the observation error and $P_k^i \in \mathbb{R}^{L_i \times L_i}$ is the weight estimation error covariance matrix at step k, $w^i \in \mathbb{R}^{L_i}$ is the weight vector, L_i is the respective number of neural network weights, $\hat{y} \in \mathbb{R}^p$ is the neural network output, $y \in \mathbb{R}^p$ is the plant output, n is the number of states, $K^i \in \mathbb{R}^{L_i \times p}$ is the Kalman gain matrix, $Q^i \in \mathbb{R}^{L_i \times L_i}$ is the NN weight estimation noise covariance matrix, $R^i \in \mathbb{R}^{p \times p}$ is the error noise covariance, and finally, $H^i \in \mathbb{R}^{L_i \times p}$ is a matrix, in which each entry is the derivative of the i-th neural output with respect to ij-th NN weight, given as:

$$H_k^{ij} = \left[\frac{\partial \hat{y}_k}{\partial w_k^{ij}} \right]^T \tag{12}$$

where $j = 1, ..., L_i$ and $i = 1, ..., n$. Usually, P^i and Q^i are initialized as diagonal matrices.

The use of EKF algorithms allows for an accurate parameter identification performed on-line. On-line identification using recurrent neural networks using the Kalman filter has been analyzed in different works such as Straub and Schroder (1996) and in Rajesh et al. (2010) where the EKF training algorithm is compared with the maximum likelihood estimation (MLE) and the mean square error algorithms for neural network modeling of a nonlinear system and it was found that EKF is the fastest to converge and has good performance compared to the other algorithms. A similar scheme as the one presented in this paper for identifying a recurrent high order neural network can be seen in Antonio-Toledo et al. (2015), where it is used with neural inverse optimal control for trajectory tracking of a three-phase induction motor. The same strategy for identification is implemented in Ruiz et al. (2012) to control a doubly fed induction generator (DFIG) with block control technique.

3. NEURAL CONTROL DESIGN

3.1 Buck Converter model

The buck converter circuit used has a capacitor at the connection point with the solar panel, as can be seen in Fig. 2, to be able to take V_{PV} as a state.

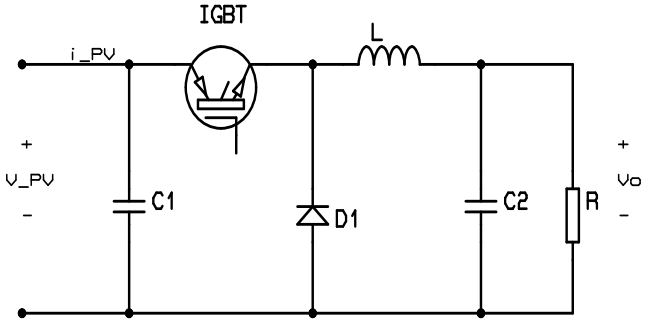


Fig. 2. Buck Model

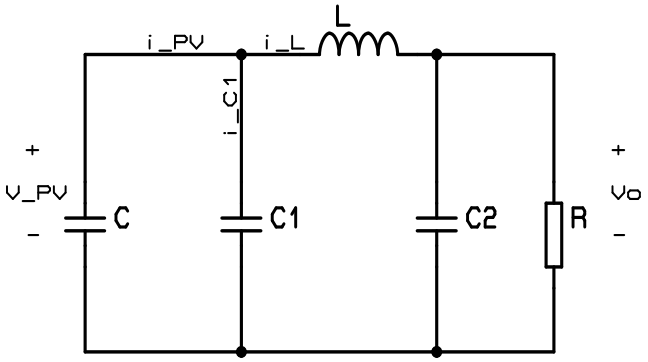


Fig. 3. Buck equivalent circuit with u=1

Three states are taken into account in the model, which are: the voltage given by the solar panel V_{PV}, the voltage at the output load resistor V_o and the current flowing through the inductor i_L.

$$\begin{aligned} x_1 &= V_{PV} \\ x_2 &= V_o \\ x_3 &= i_L \end{aligned}$$

Two models are obtained depending on the state of the IGBT control input, u; afterwards, these are combined into a single state space model, which will be used for the identification.

When the IGBT is in conduction mode ($u = 1$), the equivalent circuit can be seen in Fig. 3. The corresponding state space is defined as:

$$\begin{aligned} \dot{x}_1 &= -\frac{x_3}{C_1} + \frac{i_{PV}}{C_1} \\ \dot{x}_2 &= -\frac{x_2}{C_2 R} + \frac{x_3}{C_2} \\ \dot{x}_3 &= \frac{x_1}{L} - \frac{x_2}{L} \end{aligned} \tag{13}$$

When the IGBT is in non-conduction mode ($u = 0$), the equivalent circuit can be seen in Fig. 4. This way, the state space is defined as:

$$\begin{aligned} \dot{x}_1 &= -\frac{i_{PV}}{C_1} \\ \dot{x}_2 &= -\frac{x_2}{RC_2} + \frac{x_3}{C_2} \\ \dot{x}_3 &= -\frac{x_2}{L} \end{aligned} \tag{14}$$

From the state spaces (13) and (14), a new state space model can be obtained:

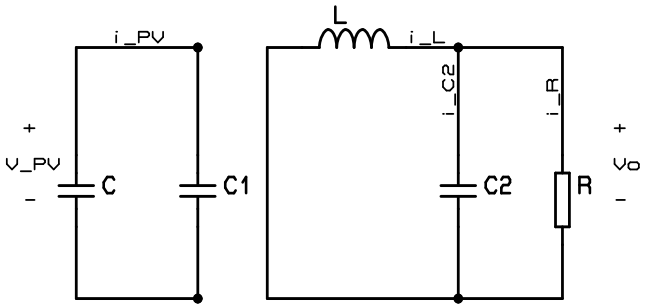


Fig. 4. Buck equivalent circuit with u=0

$$\begin{aligned} \dot{x}_1 &= \frac{i_{PV}}{C_1} + \alpha_{13} x_3 u \\ \dot{x}_2 &= \alpha_{22} x_2 + \alpha_{23} x_3 \\ \dot{x}_3 &= \alpha_{32} x_2 + \alpha_{31} x_1 u \end{aligned} \tag{15}$$

where,

$$\alpha_{13} = -\frac{1}{C_1}, \quad \alpha_{22} = -\frac{1}{RC_2}, \quad \alpha_{23} = -\frac{1}{C_2}, \quad \alpha_{32} = -\frac{1}{L}, \quad \alpha_{31} = \frac{1}{L}.$$

The model (15), can also be written as:

$$\dot{x} = f(x) + g(x)u, \tag{16}$$

where,

$$f(x) = \begin{bmatrix} \frac{i_{PV}}{C_1} \\ \alpha_{22} x_2 + \alpha_{23} x_3 \\ \alpha_{32} x_2 \end{bmatrix}, \quad g(x) = \begin{bmatrix} \alpha_{13} x_3 \\ 0 \\ \alpha_{31} x_1 \end{bmatrix}.$$

In order to obtain a discrete-time model, the Euler discretization method is used:

$$\begin{aligned} x^1_{k+1} &= x^1_k + \alpha^1_k + \alpha^2 x^3_k u_k \\ x^2_{k+1} &= x^2_k + \alpha^3 x^2_k + \alpha^4 x^3_k \\ x^3_{k+1} &= x^3_k + \alpha^3 x^2_k + \alpha^6 x^1_k u_k \\ y(k) &= x^1_k \end{aligned} \tag{17}$$

where,

$$\alpha^1_k = \frac{t_s i_{PVk}}{C_1}, \quad \alpha^2 = -\frac{t_s}{C_1}, \quad \alpha^3 = -\frac{t_s}{RC_2}, \quad \alpha^4 = \frac{t_s}{C_2}, \quad \alpha^5 = -\frac{t_s}{L}, \quad \alpha^6 = \frac{t_s}{L}.$$

3.2 Identification

Based on the structure of (17), the RHONN proposed for the DC-DC Buck converter is as follows:

$$\begin{aligned} \hat{x}^1_{k+1} &= w^{11}_k \hat{x}^1_k + w^{12}_k S(\hat{x}^3_k) + w^{13}_k S(i_{PVk}) + 0.01 u_k \\ \hat{x}^2_{k+1} &= w^{21}_k \hat{x}^2_k \\ \hat{x}^3_{k+1} &= w^{31}_k \hat{x}^3_k + w^{32}_k \hat{x}^2_k + w^{33}_k u_k \\ \hat{y}_k &= \hat{x}^1_k \end{aligned} \tag{18}$$

where, $S(\cdot)$ is a logistic function, as was seen in the mathematical preliminaries. The second and third equations in (18) correspond to the internal dynamics of the system. The second equation describes the dynamics of the voltage at the load resistor of the buck converter; due to the nature of the converter, this voltage will always be lower than V_{PV}. The third equation represents the dynamics of the current through the inductor. The weight vectors are updated online using the extended Kalman filter (EKF), the estimation error is defined by:

$$\tilde{x_k} = x_k - \hat{x}_k \tag{19}$$

It is worth to note that the states need to be measurable.

3.3 Control Design

The control is based on the identification described in the previous subsection, its objective is that the voltage at the output of the solar panel reaches the trajectory given by the MPPT, then the tracking error is defined as

$$e_k = \hat{x}^1_k - x^{1ref}_k. \tag{20}$$

The dynamic error is obtained as follows:

$$\begin{aligned} e_{k+1} &= \hat{x}^1_{k+1} - x^{1ref}_{k+1} \\ &= w^{11}_k \hat{x}^1_k + w^{12}_k S(\hat{x}^3_k) + w^{13}_k S(i_{PVk}) \\ &\quad + 0.01 u_k - x^{1ref}_{k+1}. \end{aligned} \tag{21}$$

The desired dynamic error is $e_{k+1} = -k_1 e_k$, which implies a control law as:

$$u_{eq} = \frac{-1}{0.01}\left(k_1 e_k - x^{1ref}_{k+1} + w^{11}_k \hat{x}^1_k + w^{12}_k S(\hat{x}^3_k) + w^{13}_k S(i_{PVk})\right) \tag{22}$$

where, $0 < k_1 \leq 1$ is a control design constant to minimize the error asymptotically.

4. SIMULATION RESULTS

In order to test the performance of the proposed neural controller, a simulation is developed implementing the DC-DC Buck converter and the solar panel by means of Simscape Power Systems[2] blocks, which allows to consider the components dynamics for future real-time implementation.

For the realized test, the temperature in the cell is considered constant at 25 °C. At the beginning a 0 W/m^2 irradiance is applied, then at 1.5 seconds it is increased to 500 W/m^2, finally at 3 and 4.5 seconds the irradiance is changed to 1000 and 3000 W/m^2 respectively. At the beginning of the simulation the plant is left in open-loop to identify the states, then at 0.2 seconds the loop is closed and the controller starts to operate. In Fig. 5, the theoretical power level given by (1), using the photovoltaic panel model and applying the previously described irradiance changes, is shown in the red dot line; and the power obtained by the proposed MPPT is displayed in blue line. It can be seen that the convergence time of the controller and the tracking error are within acceptable boundaries.

In Fig. 6, the identification errors are shown. It can be seen that the estimated state with the biggest error is the

[2] Simscape Power Systems is a trademark of The MathWorks, Inc.

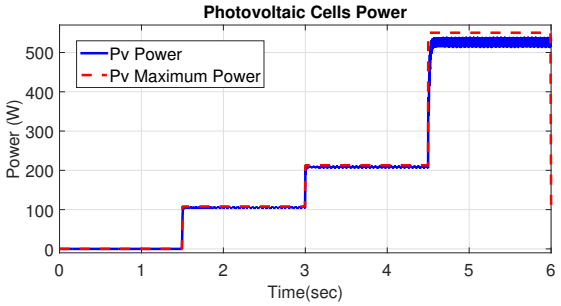


Fig. 5. Photovoltaic power at different irradiance values

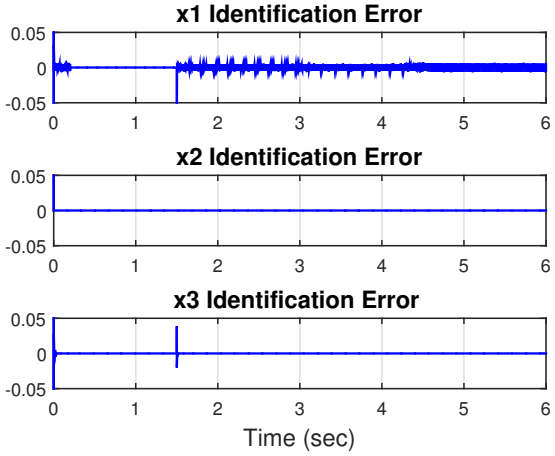


Fig. 6. Identification error

Table 1. Statistical comparison of controllers

	NN Controller		DSM Controller	
	Mean	**SD**	**Mean**	**SD**
0%	0.4267	**2.2182**	-0.1681	2.8135
10%	0.3131	2.2822	-0.1488	2.9877
20%	0.4472	2.3697	**-0.1284**	3.322
30%	0.4143	2.5501	-0.1388	3.5537
40%	0.4639	2.4802	-0.168	4.0019

voltage given by the solar panel, especially when irradiation levels go from $0\,\mathrm{W/m^2}$ to $500\,\mathrm{W/m^2}$, nevertheless the errors remain within adequate bounds for the entire simulation event though changes in irradiance are applied.

In order to compare the proposed neural algorithm with a same class but different controller, an additional simulation is performed using a discrete sliding mode controller (DSM) based on Khiari et al. (2004), with parametric changes in the DC-DC Buck converter components. Table 1, shows the mean and the standard deviation (SD) of the error with changes in capacitance and inductance values to different percentages. The best values for each statistical measure are emphasized in bold. It is evident that the error mean is lower when using the DSM controller but the SD increases when parametric changes exist, while the neural controller mean and SD remain mostly constant. The error in the mean of the neural controller can be attribute to the always existing identification errors.

5. CONCLUSION

This paper presents a novel application of the neural network on-line identification using the EKF as in Sanchez, E. N., Alanis A.Y. (2004), to achieve a photovoltaic panel MPP reference tracking. This method for identification provides robustness against parametric changes in the components. The results validate the performance of the proposed controller for irradiance changes compared to a DSM controller, proving its precision and high reaction velocity. The simulation was developed using Simscape Power Systems which includes the different component dynamics, establishing the basis for a real-time implementation.

REFERENCES

Alabedin, a.M.Z., El-Saadany, E.F., and Salama, M.M.a. (2011). Maximum power point tracking for Photovoltaic systems using fuzzy logic and artificial neural networks. *2011 IEEE Power and Energy Society General Meeting*, 1–9. doi:10.1109/PES.2011.6039690.

Alanis, A.Y., Sanchez, E.N., and Loukianov, A.G. (2007). Discrete-time backstepping induction motor control using a sensorless recurrent neural observer. In *Proceedings of the IEEE Conference on Decision and Control*, 6112–6117. doi:10.1109/CDC.2007.4434164.

Antonio-Toledo, M.E., Sanchez, E.N., and Loukianov, A.G. (2015). Real-time neural inverse optimal control for position trajectory tracking of an induction motor. In *System of Systems Engineering Conference (SoSE), 2015 10th*, 193–198. doi: 10.1109/SYSOSE.2015.7151923.

Brown, R.G. and Hwang, P.Y.C. (1997). *Introduction to Random Signals and Applied Kalman Filtering*. 4. doi: 10.1521/ijgp.2010.60.4.455.

Cubas, J., Pindado, S., and De Manuel, C. (2014). Explicit expressions for solar panel equivalent circuit parameters based on analytical formulation and the lambert W-function. *Energies*, 7(7), 4098–4115. doi: 10.3390/en7074098.

Esram, T. and Chapman, P.L. (2007). Comparison of Photovoltaic Array Maximum Power Point Tracking Techniques. *IEEE Transactions on Energy Conversion*, 22(2), 439–449. doi:10.1109/TEC.2006.874230.

Feldkamp, L.A., Feldkamp, T.M., Prokhorov, P. (2001). Neural Network training with the nprKF. In *International Joint Conference on Neural Networks'01*, 109–114.

Hadjab, M., Berrah, S., and Abid, H. (2012). Neural network for modeling solar panel. *INTERNATIONAL JOURNAL OF ENERGY*, 6(1), 9–16.

Khiari, B., Sellami, A., Andoulsi, R., M'Hiri, R., and Ksouri, M. (2004). Discrete control by sliding mode of a photovoltaic system. In *Control, Communications and Signal Processing, 2004. First International Symposium on*, 469–474. doi:10.1109/ISCCSP.2004.1296330.

Rajesh, M.V., Archana, R., Unnikrishnan, A., and Gopikakaumari, R. (2010). Comparative study on ekf training algorithm with em and mle for ann modeling of nonlinear systems. In *Proceedings of the 29th Chinese Control Conference*, 1407–1413.

Ramaprabha, R., Gothandaraman, V., Kanimozhi, K., Divya, R., and Mathur, B.L. (2011). Maximum power

point tracking using GA-optimized artificial neural network for Solar PV system. *2011 1st International Conference on Electrical Energy Systems*, (1), 264–268. doi: 10.1109/ICEES.2011.5725340.

Rovithakis, G.A. and Christodoulou, M.A. (2000). *Adaptive control with recurrent high order neural networks / theory and industrial applications.*

Ruiz, R., Sanchez, E.N., and Loukianov, A.G. (2012). Real-time neural block control for a doubly fed induction generator. In *World Automation Congress (WAC), 2012*, 1–6. Puerto Vallarta, Mexico.

Sanchez, E. N., Alanis A.Y., C.G. (2004). Recurrent neural networks trained with kalman filtering for discrete chaos recontruction. In *Asian-pacific workshop on chaos control and synchronization'04*. Melbourne, Australia.

Song, Y. and Grizzle, J.W. (1992). The Extended Kalman Filter as a Local Asymptotic Observer for Nonlinear Discrete-Time Systems. In *American Control Conference, 1992*, 3365–3369.

Straub, S. and Schroder, D. (1996). Identification of nonlinear dynamic systems with recurrent neural networks and kalman filter methods. In *Circuits and Systems, 1996. ISCAS '96., Connecting the World., 1996 IEEE International Symposium on*, volume 3, 341–344 vol.3. doi:10.1109/ISCAS.1996.541603.

Zhang, L. and Bai, Y.F. (2005). Genetic algorithm-trained radial basis function neural networks for modelling photovoltaic panels. *Engineering Applications of Artificial Intelligence*, 18(7), 833–844. doi: 10.1016/j.engappai.2005.02.004.

Non-Singular Predefined-Time Stable Manifolds

Juan Diego Sánchez-Torres * **Esteban Jiménez-Rodríguez** **
David Gómez-Gutiérrez ** **Alexander G. Loukianov** **

* *Department of Mathematics and Physics, ITESO, Periférico Sur Manuel Gómez Morín 8585 C.P. 45604, Tlaquepaque, Jalisco, México. (e-mail: dsanchez@iteso.mx).*
** *Department of Electrical Engineering, CINVESTAV-IPN Guadalajara, Av. del Bosque 1145 Col. El Bajío CP 45019, México. (e-mail:{ejimenezr,dgomez,louk}@gdl.cinvestav.mx).*

Abstract: In this paper it is introduced a class of non-singular manifolds with predefined-time stability. That is, for a given dynamical system with its trajectories constrained to this manifold it can be shown predefined-time stability to the origin. In addition, the function that defines the manifold and its derivative along the system trajectories are continuous, therefore no singularities are presented for the system evolution once the constrained motion starts. The problem of reaching the proposed manifold is solved by means of a continuous predefined-time stable controller. The proposal is applied to the predefined-time exact tracking of fully actuated and unperturbed mechanical systems. It is assumed the availability of the state and the desired trajectory as well as its two first derivatives. As an example, the proposed solution is applied over a two-link planar manipulator and numerical simulations are conducted to show its performance.

Keywords: Predefined-Time Stability, Sliding Mode Algorithms, Second Order Systems, Mechanical Systems.

1. INTRODUCTION

Several applications are characterized for requiring hard time response constraints. In order to deal with those requirements, various developments concerning to concept of *finite-time stability* have been carried out (see for example: Roxin (1966); Weiss and Infante (1967); Michel and Porter (1972); Haimo (1986); Utkin (1992); Bhat and Bernstein (2000); Moulay and Perruquetti (2005, 2006)). Nevertheless, usually this finite time is an unbounded function of the initial conditions of the system.

With the aim to eliminate this boundlessness, the notion of fixed-time stability have been studied in Andrieu et al. (2008); Cruz-Zavala et al. (2010); Polyakov (2012); Fraguela et al. (2012); Polyakov and Fridman (2014). Fixed-time stability represents a significant advantage over finite-time stability due to its desired feature of the convergence time, as a function of the initial conditions, is bounded. That makes the fixed-time stability a valuable feature in estimation and optimization problems.

For the most of the proposed fixed-time stable system, there are problems related with the convergence time. First, the bounds of the fixed stabilization time found by Lyapunov analysis constitute usually conservative estimations, i.e. they are much larger than the true fixed stabilization time (see for example Cruz-Zavala et al. (2011), where the upper bound estimation is approximately 100 times larger than the actual true fixed stabilization time). Second, and as consequence, it is often complicated to find a direct relationship between the tuning gains and the fixed stabilization time, making this time hard to tune.

To overcome the above, a class of first-order dynamical systems with the minimum upper bound of the fixed stabilization time equal to their only tuning gain has been studied (Sánchez-Torres et al., 2014; Sánchez-Torres et al., 2015). It is said that these systems exhibit the property of *predefined-time stability.*

In this sense, this paper introduces the concept of non-singular predefined-time stable manifolds. Similarly to Jiménez-Rodríguez et al. (2016), the proposed scheme allows to define second-order predefined-time stable systems as a nested application of first-order predefined-time stabilizing functions, with the difference that such function which defines the manifold and its derivative along the system trajectories are continuous, therefore no singularities are presented for the system evolution.

Finally, this idea is used to solve the problem of predefined-time exact tracking in fully actuated mechanical systems, assuming the availability of the state and the desired trajectory and its two first derivatives measurements.

In the following, Section 2 presents the mathematical preliminaries needed to introduce the proposed results. Section 3 exposes the main result of this paper, which is the non-singular predefined-time stable manifold design. Section 4 presents a non-singular second-order predefined-time tracking controller for fully actuated mechanical systems. Section 5 describes the model of a planar two-link manipulator, where the proposed controller is applied. The

simulation results of the example are shown in Section 6. Finally, Section 7 presents the conclusions of this paper.

2. PRELIMINARIES

2.1 Mathematical Preliminaries

Consider the system

$$\dot{x} = f(x;\rho) \tag{1}$$

where $x \in \mathbb{R}^n$ is the system state, $\rho \in \mathbb{R}^b$ represents the parameters of the system and $f : \mathbb{R}^n \to \mathbb{R}^n$ is a nonlinear function. The initial conditions of this system are $x(0) = x_0$.

Definition 2.1. (Polyakov, 2012) The origin of (1) is *globally finite-time stable* if it is globally asymptotically stable and any solution $x(t, x_0)$ of (1) reaches the equilibrium point at some finite time moment, i.e., $\forall t \geq T(x_0) : x(t, x_0) = 0$, where $T : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\}$.

Definition 2.2. (Polyakov, 2012) The origin of (1) is *fixed-time stable* if it is globally finite-time stable and the settling-time function is bounded, i.e. $\exists T_{\max} > 0 : \forall x_0 \in \mathbb{R}^n : T(x_0) \leq T_{\max}$.

Remark 2.1. Note that there are several choices for $T_{\max}$. For instance, if the settling-time function is bounded by T_{m}, it is also bounded by λT_{m} for all $\lambda \geq 1$. This motivates the following definition.

Definition 2.3. (Sánchez-Torres et al., 2014; Sánchez-Torres et al., 2015) Let $\mathcal{T}$ be the set of all the bounds of the settling time function for the system (1), i.e.,

$$\mathcal{T} = \{T_{\max} > 0 : \forall x_0 \in \mathbb{R}^n : T(x_0) \leq T_{\max}\}. \tag{2}$$

The minimum bound of the settling-time function T_f, is defined as:

$$T_f = \inf \mathcal{T} = \sup_{x_0 \in \mathbb{R}^n} T(x_0). \tag{3}$$

Remark 2.2. In a strict sense, the time T_f can be considered as the true fixed-time in which the system (1) stabilizes.

Definition 2.4. (Sánchez-Torres et al., 2014; Sánchez-Torres et al., 2015) For the case of fixed time stability when the time T_f defined in (3) can be tuned by a particular selection of the parameters ρ of the system (1), it is said that the origin of the system (1) is *predefined-time stable.*

Definition 2.5. Let $h \geq 0$. For $x \in \mathbb{R}$, define the function

$$\lfloor x \rceil^h = |x|^h \operatorname{sign}(x),$$

with $\operatorname{sign}(x) = 1$ for $x > 0$, $\operatorname{sign}(x) = -1$ for $x < 0$ and $\operatorname{sign}(0) \in [-1, 1]$.

Remark 2.3. For $x \in \mathbb{R}$, some properties of the function $\lfloor \cdot \rceil^h$ are:

(i) $\lfloor x \rceil^h$ is continuous for $h > 0$.
(ii) $\lfloor x \rceil^0 = \operatorname{sign}(x)$.
(iii) $\lfloor x \rceil^1 = \lfloor x \rceil = x$,
(iv) $\lfloor 0 \rceil^h = 0$ for $h > 0$.
(v) $\frac{d|x|^h}{dx} = h \lfloor x \rceil^{h-1}$ and $\frac{d\lfloor x \rceil^h}{dx} = h |x|^{h-1}$.
(vi) For $h_1, h_2 \in \mathbb{R}$, it follows:
- $|x|^{h_1} |x|^{h_2} = |x|^{h_1+h_2}$
- $\lfloor x \rceil^{h_1} |x|^{h_2} = |x|^{h_1} \lfloor x \rceil^{h_2} = \lfloor x \rceil^{h_1+h_2}$
- $\lfloor x \rceil^{h_1} \lfloor x \rceil^{h_2} = |x|^{h_1+h_2}$

(vii) For $h_1, h_2 > 0$, then $\left\lfloor \lfloor x \rceil^{h_1} \right\rceil^{h_2} = \lfloor x \rceil^{h_1 h_2}$.

Definition 2.6. (Sánchez-Torres et al., 2014; Sánchez-Torres et al., 2015) For $x \in \mathbb{R}$, the *predefined-time stabilizing function* is defined as:

$$\Phi_p(x; T_c) = \frac{1}{T_c p} \exp\left(|x|^p\right) \lfloor x \rceil^{1-p} \tag{4}$$

where $T_c > 0$ and $0 < p \leq 1$.

Remark 2.4. It can be checked, using Remark 2.3, that the derivative of the predefined-time stabilizing function (4) is given by

$$\frac{d\Phi_p(x; T_c)}{dx} = \frac{\exp\left(|x|^p\right)}{T_c p} \left[p + (1-p) \frac{1}{|x|^p} \right], \quad \forall x \neq 0 \tag{5}$$

To handle vector systems, the above definitions are extended.

Definition 2.7. Let $h \geq 0$, $T_c > 0$, $0 < p \leq 1$ and $v = \begin{bmatrix} v_1 & \cdots & v_k \end{bmatrix}^T \in \mathbb{R}^k$. Then, the functions $\operatorname{sign}(\cdot)$, $|\cdot|^h$, $\lfloor \cdot \rceil^h$ and $\Phi_p(\cdot; T_c)$ are extended component-wise, as follows:

(i) $\operatorname{sign}(v) = \begin{bmatrix} \operatorname{sign}(v_1) & \cdots & \operatorname{sign}(v_k) \end{bmatrix}^T$
(ii) $|v|^h = \begin{bmatrix} |v_1|^h & \cdots & |v_k|^h \end{bmatrix}^T$;
(iii) $\lfloor v \rceil^h = \begin{bmatrix} \lfloor v_1 \rceil^h & \cdots & \lfloor v_k \rceil^h \end{bmatrix}^T$;
(iv) $\Phi_p(v; T_c) = \begin{bmatrix} \Phi_p(v_1; T_c) & \cdots & \Phi_p(v_k; T_c) \end{bmatrix}^T$.

Definition 2.8. Let $v = \begin{bmatrix} v_1 & \cdots & v_k \end{bmatrix}^T \in \mathbb{R}^k$. Then $\operatorname{diag}(v)$ will denote the $k \times k$ matrix defined as

$$\operatorname{diag}(v) = \begin{bmatrix} v_1 & 0 & \cdots & 0 \\ 0 & v_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & v_k \end{bmatrix}.$$

Remark 2.5. The properties *(i)*, *(ii)*, *(iii)*, *(iv)* and *(vi)* of Remark 2.3 remain the same. For $v \in \mathbb{R}^k$, the derivatives of the functions $|\cdot|^h$, $\lfloor \cdot \rceil^h$ and $\Phi_p(\cdot; T_c)$ are:

$$\frac{\partial |v|^h}{\partial v} = \operatorname{diag}\left[h \lfloor v_1 \rceil^{h-1} \quad \ldots \quad h \lfloor v_k \rceil^{h-1} \right] = h \operatorname{diag} \lfloor v \rceil^{h-1},$$

$$\frac{\partial \lfloor v \rceil^h}{\partial v} = \operatorname{diag}\left[h |v_1|^{h-1} \quad \ldots \quad h |v_k|^{h-1} \right] = h \operatorname{diag} |v|^{h-1}$$

and

$$\frac{\partial \Phi_p(v; T_c)}{\partial v} = \operatorname{diag}\left[\frac{d\Phi_p(v_1; T_c)}{dv_1} \quad \ldots \quad \frac{d\Phi_p(v_k; T_c))}{dv_k} \right],$$

respectively.

Remark 2.6. It is important to note that if $k = 1$, all the extensions reduce to the scalar case considered by Definition 2.5, Remark 2.3 and Definition 2.6.

From the Definition 2.6 of the stabilizing function, the following Lemma presents a dynamical system with the predefined-time stability property.

Lemma 2.1. (Sánchez-Torres et al., 2014; Sánchez-Torres et al., 2015) The origin of the system

$$\dot{x} = -\Phi_r(x; T_c) \tag{6}$$

with $T_c > 0$, and $0 < r < 1$ is predefined-time stable with $T_f = T_c$. That is, $x(t) = 0$ for $t > T_c$ in spite of the $x(0)$ value.

2.2 Motivation

Consider the following second order system:

$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= -\alpha \text{sign}\left(x_2 + \beta \lfloor x \rceil^{\frac{1}{2}}\right)\end{aligned} \tag{7}$$

where $x_1, x_2, u \in \mathbb{R}$ and $\alpha, \beta > 0$. The initial conditions of this system are $x_1(0) = x_{1,0}$ and $x_2(0) = x_{2,0}$.

Let the variable $\sigma = x_2 + \lfloor x \rceil^{\frac{1}{2}}$ and its time derivative given by

$$\dot{\sigma} = -\alpha \text{sign}(\sigma) + \frac{1}{2}\beta x_2 |x_1|^{-\frac{1}{2}}. \tag{8}$$

For $\alpha > \beta/2$ a sliding motion on the manifold $\sigma = 0$ is obtained in finite time. This can verified by evaluating the dynamics of (8) when the sliding motion starts. Once the manifold $\sigma = 0$ is reached, the dynamics of (7) reduces to

$$\dot{x}_1 = -\beta \lfloor x \rceil^{\frac{1}{2}} \tag{9}$$

that is finite-time stable. Therefore, there is a time $T = T(x_{1,0}, x_{2,0})$ such that $\sigma = 0$ and $x_1 = 0$ for every time $t \geq T$, which implies that $x_2 = 0$ for $t \geq T$.

Remark 2.7. The exposed procedure is the main idea behind of the terminal sliding mode controllers (Venkataraman and Gulati, 1992) since the motion on $\sigma = 0$ is also finite time stable and, the nested high-order sliding mode controllers (Levant, 2003) since x_1 and its derivative x_2 are driven to zero in finite time and the system has a nested structure.

3. PREDEFINED-TIME STABLE NON-SINGULAR MANIFOLDS

Similarly to the nested approach presented given in (7)-(8), in order to obtain a similar second order system but with predefined-time stability, consider the double integrator system

$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= u\end{aligned} \tag{10}$$

where $x_1, x_2, u \in \mathbb{R}$.

As first attempt, from (4), the variable

$$\sigma = x_2 + \frac{1}{T_{c_1} p} \exp(|x_1|^p) \lfloor x \rceil^{1-p}. \tag{11}$$

with $T_{c_1} > 0$ can be used. However, note that the dynamics of (11) is given by

$$\dot{\sigma} = u + \frac{\exp\left(|x_1|^p\right)}{T_{c_1} p}\left[p + (1-p)\frac{1}{|x_1|^p}\right] x_2 \tag{12}$$

which has a singularity at $x_1 = 0$. Therefore, the variable σ in (11) is called a singular sliding variable.

Considering that drawback, it is desirable a variable which provides the same dynamics in $\sigma = 0$ than these presented in (11), but avoiding the singularity thus (12) exposes. With this aim, let the following variables:

$$\begin{aligned}\sigma_1 &= x_1\\ \sigma_2 &= (1-p)\lfloor x_2 \rceil^{\frac{1}{1-p}} + (1-p)\lfloor \Phi_p(x_1; T_{c_1}) \rceil^{\frac{1}{1-p}},\end{aligned} \tag{13}$$

where $\lfloor \Phi_p(x_1; T_{c_1}) \rceil^{\frac{1}{1-p}} = \left[\frac{1}{T_c p}\right]^{\frac{1}{1-p}} \exp\left(\frac{1}{1-p}|x_1|^p\right) \lfloor x_1 \rceil$ with $0 < p < \frac{1}{2}$.

Hence, the system (10) can be written as

$$\begin{aligned}\dot{\sigma}_1 &= -\left\lfloor \lfloor \Phi_p(\sigma_1; T_{c_1}) \rceil^{\frac{1}{1-p}} - \frac{1}{1-p}\sigma_2 \right\rceil^{1-p}\\ \dot{\sigma}_2 &= |x_2|^{\frac{p}{1-p}} u + \psi(\sigma_1) \lfloor x_2 \rceil,\end{aligned}$$

where $\psi(\sigma_1) = \left[\frac{1}{T_{c_1} p}\right]^{\frac{1}{1-p}} \exp\left(\frac{1}{1-p}|\sigma_1|^p\right)\left[p|\sigma_1|^p + (1-p)\right]$.

Remark 3.1. The variable σ_2 in (13) is based on the approach proposed in Feng et al. (2002). However, here it is not necessary to define fractional powers in terms of odd integers.

With $u = -|x_2|^{\frac{p}{p-1}}\left[\Phi_r(\sigma_2; T_{c_2}) + \psi(x_1)\lfloor x_2 \rceil\right]$, $T_{c_2} > 0$ and $0 < r < 1$, it yields

$$\begin{aligned}\dot{\sigma}_1 &= -\left\lfloor \lfloor \Phi_p(\sigma_1; T_{c_1}) \rceil^{\frac{1}{1-p}} - \frac{1}{1-p}\sigma_2 \right\rceil^{1-p}\\ \dot{\sigma}_2 &= -\Phi_r(\sigma_2; T_{c_2}).\end{aligned} \tag{14}$$

The stability analysis of the system (14) is an direct application of Lemma 2.1. For $t > T_{c_2}$, $\sigma_2 = 0$ and the system reduces to $\dot{\sigma}_1 = -\Phi_p(\sigma_1; T_{c_1})$. Then, for $t > T_{c_1} + T_{c_2}$, $(\sigma_1, \sigma_2) = (0, 0)$. Consequently, from (13), $(x_1, x_2) = (0, 0)$ for $t > T_{c_1} + T_{c_2}$.

Remark 3.2. From (14), it can be noted x_2 cannot be zero before $\sigma_2 = 0$. Besides, once $\sigma_2 = 0$, the control signal becomes

$$u_{\sigma_2 = 0} = -|x_2|^{\frac{p}{p-1}} \psi(x_1) \lfloor x_2 \rceil = -\psi(x_1) \lfloor x_2 \rceil^{\frac{2p-1}{p-1}},$$

which is continuous since $0 < p < \frac{1}{2}$. In addition, it can be observed that the term $|x_2|^{\frac{p}{p-1}}$ in the controller vanishes in predefined time T_{c_2}, avoiding a singularity at $x_2 = 0$.

4. PREDEFINED-TIME TRACKING CONTROLLER OF A CLASS OF MECHANICAL SYSTEMS

4.1 Problem Statement

A generic model of second-order, fully actuated mechanical systems of n degrees of freedom has the form

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + P(\dot{q}) + \gamma(q) = \tau, \tag{15}$$

where $q, \dot{q}, \ddot{q} \in \mathbb{R}^n$ are the position, velocity and acceleration vectors in joint space; $M(q) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ is the Coriolis and centrifugal effects matrix, $P(\dot{q}) \in \mathbb{R}^n$ is the damping effects vector, usually from viscous and/or Coulomb friction and $\gamma(q) \in \mathbb{R}^n$ is the gravity effects vector.

Defining the variables $x_1 = q$, $x_2 = \dot{q}$ and $u = \tau$, the mechanical model (15) can be rewritten in the following state-space form

$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= f(x_1, x_2) + B(x_1, x_2)u,\end{aligned} \tag{16}$$

where $f(x_1, x_2) = -M^{-1}(x_1)\left[C(x_1, x_2)x_2 + P(x_2) + \gamma(x_1)\right]$, $B(x_1, x_2) = M^{-1}(x_1)$ are continuous maps and the initial conditions are $x_1(0) = x_{1,0}$, $x_2(0) = x_{2,0}$.

Remark 4.1. The matrix function $M(x_1)$ is, in fact, invertible since $M(x_1) = M^T(x_1)$ is positive definite.

A common problem in mechanical systems control is to track a desired time-dependent trajectory described by the triplet $(q_d(t), \dot{q}_d(t), \ddot{q}_d(t))$ of desired position

$q_d(t) = [q_{d_1}(t) \ \cdots \ q_{d_n}(t)]^T \in \mathbb{R}^n$, velocity $\dot{q}_d(t) = [\dot{q}_{d_1}(t) \ \cdots \ \dot{q}_{d_n}(t)]^T \in \mathbb{R}^n$ and acceleration $\ddot{q}_d(t) = [\ddot{q}_{d_1}(t) \ \cdots \ \ddot{q}_{d_n}(t)]^T \in \mathbb{R}^n$, which are all assumed to be known.

To be consequent with the state space notation, the desired position and velocity vectors are redefined as $x_{1,d} = q_d$ and $x_{2,d} = \dot{q}_d = \dot{x}_{1,d}$, respectively. Then, defining the error variables as $e_1 = x_1 - x_{1,d}$ (position error) and $e_2 = x_2 - x_{2,d}$ (velocity error), the error dynamics are:

$$\begin{aligned} \dot{e}_1 &= e_2 \\ \dot{e}_2 &= f(x_1, x_2) + B(x_1, x_2)u - \ddot{x}_{1,d}, \end{aligned} \tag{17}$$

with initial conditions $e_1(0) = e_{1,0} = x_{1,0} - x_{1,d}(0)$, $e_2(0) = e_{2,0} = x_{2,0} - x_{2,d}(0)$.

The task is to design a state-feedback, second-order, predefined-time controller to track the desired trajectory. In other words, the error variables e_1 and e_2 are to be stabilized in predefined time with available measurements of x_1, x_2, $x_{1,d}$, $x_{2,d} = \dot{x}_{1,d}$ and $\ddot{x}_{1,d}$.

4.2 Controller Design

With basis on Definition 2.7, consider the non-singular transformation

$$\begin{aligned} s_1 &= e_1 \\ s_2 &= (1-p)\lfloor e_2 \rceil^{\frac{1}{1-p}} + (1-p)\lfloor \Phi_p(e_1; T_{c_1}) \rceil^{\frac{1}{1-p}}, \end{aligned} \tag{18}$$

with $0 < p < \frac{1}{2}$.

From (17), the dynamics of the system in the new coordinates (s_1, s_2) can be written as

$$\begin{aligned} \dot{s}_1 &= -\left\lfloor \Phi_p(s_1; T_{c_1}) \right\rceil^{\frac{1}{1-p}} - \frac{1}{1-p} s_2 \Bigg\rceil^{1-p} \\ \dot{s}_2 &= \operatorname{diag} |e_2|^{\frac{p}{1-p}} \left[f(x_1, x_2) + B(x_1, x_2)u - \ddot{x}_{1,d} \right] + \Psi(s_1) e_2, \end{aligned} \tag{19}$$

where $\Psi(s_1) = \operatorname{diag} |\Phi_p(s_1; T_{c_1})|^{\frac{p}{1-p}} \frac{\partial \Phi_p(s_1; T_{c_1})}{\partial s_1}$.

Hence, for the system (19) the following controller is proposed:

$$\begin{aligned} u = -B^{-1}(x_1, x_2) \Bigg[& f(x_1, x_2) - \ddot{x}_{1,d} + \\ & \operatorname{diag} |e_2|^{\frac{p}{p-1}} \left[\Psi(s_1) e_2 + \Phi_r(s_2; T_{c_2}) \right] \Bigg], \end{aligned} \tag{20}$$

with $0 < r < 1$ and $T_{c_2} > 0$.

Thus, the system (19) closed-loop with the controller (20) has the form

$$\begin{aligned} \dot{s}_1 &= -\left\lfloor \Phi_p(s_1; T_{c_1}) \right\rceil^{\frac{1}{1-p}} - \frac{1}{1-p} s_2 \Bigg\rceil^{1-p} \\ \dot{s}_2 &= -\Phi_r(\sigma_2; T_{c_2}). \end{aligned} \tag{21}$$

Taking into account the structure of the system (21), the following theorem states the tracking of the system (15).

Theorem 4.1. For the system (15), $q = q_d$ and $\dot{q} = \dot{q}_d$ for $t > T_{c_1} + T_{c_2}$.

Proof. The proof is similar to the stability analysis carried out in Section 3 and, hence, is omitted. ∎

5. EXAMPLE: TRAJECTORY TRACKING FOR A TWO-LINK MANIPULATOR

Consider a planar, two-link manipulator with revolute joints as the one exposed in Utkin et al. (2009) (see Fig. 1). The manipulator link lengths are L_1 and L_2, the link masses (concentrated in the end of each link) are M_1 and M_2. The manipulator is operated in the plane, such that the gravity acts along the z–axis.

Examining the geometry, it can be seen that the end-effector (the end of the second link, where the mass M_2 is concentrated) position (x_w, y_w) is given by

$$\begin{aligned} x_w &= L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) \\ y_w &= L_1 \sin(q_1) + L_2 \sin(q_1 + q_2), \end{aligned}$$

where q_1 and q_2 are the joint positions (angular positions).

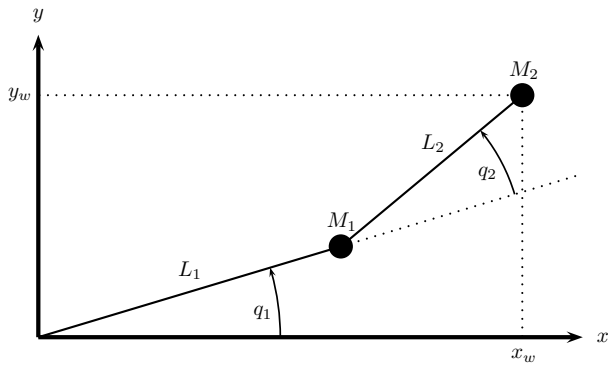


Figure 1. Two-link manipulator.

Applying the Euler-Lagrange equations, a model according to (15) is obtained, with

$$\begin{aligned} m_{11} &= L_1^2(M_1 + M_2) + 2(L_2^2 M_2 + L_1 L_1 M_2 \cos q_2) - L_2^2 M_2 \\ m_{12} &= m_{21} = L_2^2 M_2 + L_1 L_1 M_2 \cos q_2 \\ m_{22} &= L_2^2 M_2 \\ h &= L_1 L_2 M_2 \sin q_2 \\ c_{11} &= -h\dot{q}_2 \\ c_{12} &= -h(\dot{q}_1 + \dot{q}_2) \\ c_{21} &= h\dot{q}_1 \\ c_{22} &= 0, \end{aligned}$$

$$M(q) = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}, \quad C(q, \dot{q}) = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

$$P(\dot{q}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \gamma(q) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

The absence of gravity term is because the manipulator is operated in the plane, perpendicular to gravity. Note also that friction terms are neglected.

For this example, the end-effector of the manipulator is required to follow a circular trajectory of radius r_d and center in the origin. To solve this problem the controller exposed in Section 4 is applied.

6. SIMULATION RESULTS

The simulation results of the example in Section 5 are presented in this section. The two-link manipulator parameters used are shown in Table 1.

The simulations were conducted using the Euler integration method, with a fundamental step size of 1×10^{-4} s. The initial conditions for the two-link manipulator were selected as: $x_1(0) = \begin{bmatrix} -\frac{3\pi}{4} & -\frac{\pi}{4} \end{bmatrix}^T$ and $x_2(0) = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$. In addition, the controller gains were adjusted to: $T_{c_1} = 1$, $T_{c_2} = 0.5$, $p_1 = \frac{1}{3}$ and $p_2 = \frac{1}{2}$.

The desired circular trajectory in the joint coordinates is described by the equations $q_d(t) = x_{1,d}(t) = \begin{bmatrix} q_{d_1}(t) & q_{d_2}(t) \end{bmatrix} = \begin{bmatrix} \frac{\pi}{2}t - \pi & -\frac{\pi}{2} \end{bmatrix}$ and it corresponds to a circumference of radius 0.2828 m.

The following figures show the behavior of the proposed controller.

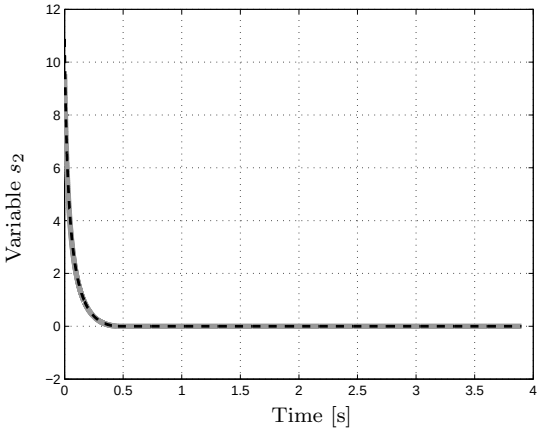


Figure 2. Variable s_2. First component (gray and solid) and second component (black and dashed). Note that $s_2(t) = 0$ for $t > T_{c_2} = 0.5$ s.

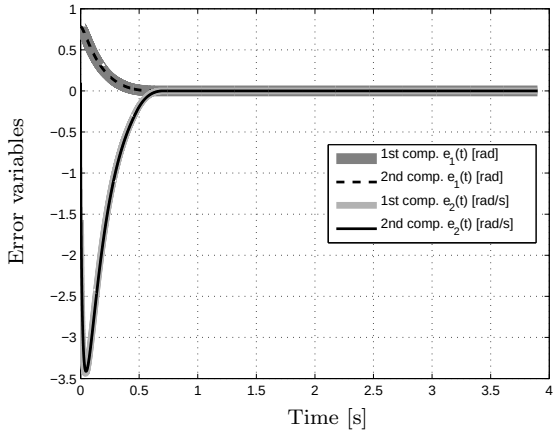


Figure 3. Error variables. First component of e_1 (dark gray and thick), second component of e_1 (black and dashed), first component of e_2 (light gray and solid) and second component of e_2 (black and solid).

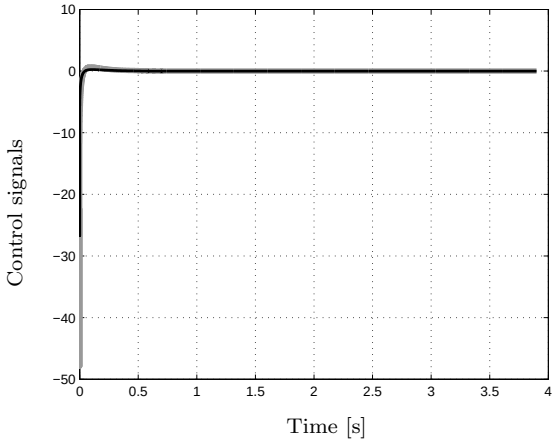


Figure 4. Control signal. First component (gray and solid) and second component (black and solid).

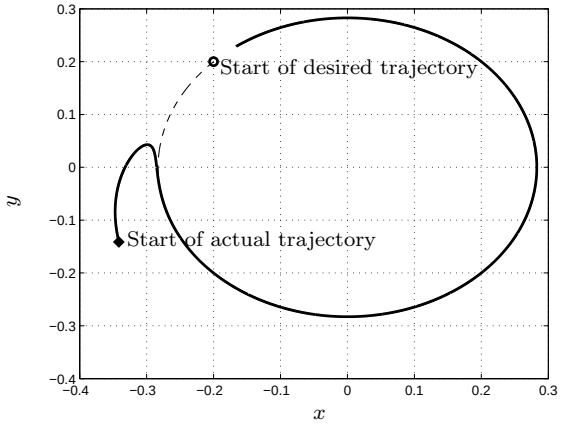


Figure 5. Actual trajectory (x_w, y_w) (black and solid) and desired trajectory $(x_{w,d}, y_{w,d})$ (black and dashed).

Note that $\sigma_2(t) = 0$ for $t \geq 0.47$ s $< T_{c_2} = 0.5$ s (Fig. 2). Once the error variables slide over the manifold $\sigma_2 = 0$, this motion is governed by the reduced order system

$$\dot{e}_1 = e_2 = -\Phi_{p_1}(e_1; T_{c_1}).$$

This imply that the error variables are exactly zero for $t > T_{c_1} + T_{c_2} = 1.5$ s. In fact, from Fig. 3, it can be seen that $e_1(t) = e_2(t) = 0$ for $t \geq 0.74$ s $< T_{c_1} + T_{c_2} = 1.5$ s. Fig. 4 shows the control signal (torque) versus time. Finally, from Fig. 5, it can be seen the reference tracking in rectangular coordinates.

Table 1. Parameters of the two-link manipulator model.

Parameter	Values	Unit
M_1	0.2	kg
M_2	0.2	kg
L_1	0.2	m
L_2	0.2	m

7. CONCLUSION

In this paper a class of non-singular manifolds with predefined-time stability was introduced. As a result, the trajectories of a given dynamical system constrained to this class of manifolds have predefined-time stability to the origin and, in addition, the function that defines the manifold and its derivative along the system trajectories are continuous, therefore no singularities are presented for the system evolution once the constrained motion starts. Besides, the problem of reaching the proposed manifold was solved by means of a continuous predefined-time stable controller.

The proposal was applied to the predefined-time exact tracking of fully actuated and unperturbed mechanical systems as an example, assuming the availability of the state and the desired trajectory as well as its two first derivatives. Furthermore, the resulting controller is applied over a two-link planar manipulator and numerical simulations are conducted to show its performance.

ACKNOWLEDGMENT

Esteban Jiménez acknowledges to CONACyT, México for the MSc scholarship number 426598.

REFERENCES

Andrieu, V., Praly, L., and Astolfi, A. (2008). Homogeneous approximation, recursive observer design, and output feedback. *SIAM Journal on Control and Optimization*, 47(4), 1814–1850.

Bhat, S. and Bernstein, D. (2000). Finite-time stability of continuous autonomous systems. *SIAM Journal on Control and Optimization*, 38(3), 751–766.

Cruz-Zavala, E., Moreno, J.A., and Fridman, L.M. (2011). Uniform robust exact differentiator. *IEEE Transactions on Automatic Control*, 56(11), 2727–2733.

Cruz-Zavala, E., Moreno, J., and Fridman, L. (2010). Uniform second-order sliding mode observer for mechanical systems. In *Variable Structure Systems (VSS), 2010 11th International Workshop on*, 14 –19.

Feng, Y., Yu, X., and Man, Z. (2002). Non-singular terminal sliding mode control of rigid manipulators. *Automatica*, 38(12), 2159 – 2167.

Fraguela, L., Angulo, M., Moreno, J., and Fridman, L. (2012). Design of a prescribed convergence time uniform robust exact observer in the presence of measurement noise. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 6615–6620.

Haimo, V. (1986). Finite time controllers. *SIAM Journal on Control and Optimization*, 24(4), 760–770.

Jiménez-Rodríguez, E., Sánchez-Torres, J.D., Gómez-Gutiérrez, D., and Loukianov, A.G. (2016). Predefined-time tracking of a class of mechanical systems. In *13th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), 2016*.

Levant, A. (2003). Higher-order sliding modes, differentiation and output-feedback control. *International Journal of Control*, 76(9-10), 924–941.

Michel, A. and Porter, D. (1972). Practical stability and finite-time stability of discontinuous systems. *IEEE Transactions on Circuit Theory*, 19(2), 123–129.

Moulay, E. and Perruquetti, W. (2005). Lyapunov-based approach for finite time stability and stabilization. In *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, CDC-ECC '05*, 4742–4747. doi: 10.1109/CDC.2005.1582911.

Moulay, E. and Perruquetti, W. (2006). Finite-time stability and stabilization: State of the art. In C. Edwards, E. Fossas Colet, and L. Fridman (eds.), *Advances in Variable Structure and Sliding Mode Control*, volume 334 of *Lecture Notes in Control and Information Science*, 23–41. Springer Berlin Heidelberg.

Polyakov, A. (2012). Nonlinear feedback design for fixed-time stabilization of linear control systems. *IEEE Transactions on Automatic Control*, 57(8), 2106–2110.

Polyakov, A. and Fridman, L. (2014). Stability notions and Lyapunov functions for sliding mode control systems. *Journal of the Franklin Institute*, 351(4), 1831 – 1865. Special Issue on 2010-2012 Advances in Variable Structure Systems and Sliding Mode Algorithms.

Roxin, E. (1966). On finite stability in control systems. *Rendiconti del Circolo Matematico di Palermo*, 15(3), 273–282.

Sánchez-Torres, J.D., Sanchez, E.N., and Loukianov, A.G. (2015). Predefined-time stability of dynamical systems with sliding modes. In *American Control Conference (ACC), 2015*, 5842–5846.

Sánchez-Torres, J.D., Sánchez, E.N., and Loukianov, A.G. (2014). A discontinuous recurrent neural network with predefined time convergence for solution of linear programming. In *IEEE Symposium on Swarm Intelligence (SIS)*, 9–12.

Utkin, V.I. (1992). *Sliding Modes in Control and Optimization*. Springer Verlag.

Utkin, V.I., Guldner, J., and Shi, J. (2009). *Sliding Mode Control in Electro-Mechanical Systems, Second Edition (Automation and Control Engineering)*. CRC Press, 2 edition.

Venkataraman, S.T. and Gulati, S. (1992). Control of nonlinear systems using terminal sliding modes. In *American Control Conference, 1992*, 891–893.

Weiss, L. and Infante, E. (1967). Finite time stability under perturbing forces and on product spaces. *IEEE Transactions on Automatic Control*, 12(1), 54–59.

Nonlinear Model Predictive Control of a Passenger Vehicle for Lane Changes Considering Vehicles in the Target Lane

Andrés F. Acosta * **Alejandro Márquez** ** **Jairo Espinosa** ***

* *Universidad Nacional de Colombia, Medellín, Colombia (e-mail: afacostag@unal.edu.co).*
** *Universidad Nacional de Colombia, Medellín, Colombia (e-mail: amarque@unal.edu.co)*
*** *Universidad Nacional de Colombia, Medellín, Colombia (e-mail: jespinov@unal.edu.co)*

Abstract: This article presents a Nonlinear Model Predictive Control (MPC) for controlling the lateral dynamics of a passenger vehicle, based on a planar Single Track Model. This MPC controller is aimed at tracking suitable reference values on its states in order to achieve a lane change manoeuvre. Simulation results are showed for an scenario with a single vehicle and another scenario with two vehicles in the target lane, for different initial conditions that validate the proposed nonlinear MPC.

Keywords: Model-based control, Nonlinear control, Optimal control, Predictive control, Automated guided vehicles, Automotive control, Vehicle dynamics.

1. INTRODUCTION

Road traffic modelling and control play an important role for improving safety. Particularly, it has been found that lane changes are an important cause of accidents (Sen et al., 2003). In the last decades, some advancements in technologies such as computer vision and instrumented vehicles allowed applications known as Advanced Driving Assistance Systems (ADAS), including Adaptive Cruise Control (ACC) (Bifulco et al., 2013), lane keeping systems and parking assistance systems. Moreover, modelling and control of vehicle dynamics can help to improve microscopic traffic models. For example, it has been found that existing lane-change models focus on the decision making and gap acceptance models, but little attention has been paid to modelling the execution of the manoeuvre and the heterogeneity of drivers and vehicles (Toledo, 2007; Zheng, 2014).

Models that describe the dynamic behaviour of vehicles based in their physical characteristics such as dimensions, power-to-weight ratio and inertial parameters, among others, and input variables including the steering angle, the position of the accelerator and the braking pedal, sometimes also known as submicroscopic models (Maerivoet and De Moor, 2005), are models that, in general, comprise a phenomenological and an empirical component. The former is derived from the fundamental laws of moment conservation while the later is mainly focused in the modelling of the interaction forces between the tires and the road surface, due to the complexity of the structure of the tires. Among the vehicle dynamic models, one of the most common because of its simplicity and representativeness of lateral dynamics is the Single Track Model (STM), which has been successful used in vehicle control systems, including steering and acceleration control, in applications such as reference trajectory generation and tracking (Gerdts et al., 2009; Falcone et al., 2007), obstacle avoidance (Park et al., 2009) and stabilization (Guvenc et al., 2009).

This article presents a Nonlinear Model Predictive Control (MPC) of a passenger vehicle, based on a planar Single Track Model (STM), aimed at tracking suitable reference values on one of its states in order to achieve a lane change maneuver. Furthermore, two simulation cases are presented: one without the influence of vehicles in the target lane, and another that includes two vehicles in the target lane. Additionally, different initial conditions are tested in order to demonstrate the effectiveness of the proposed MPC.

This article is organized, as follows: Section 2 describes the nonlinear Single Track Model that serves as a basis for the MPC controller design. Section 3 presents the nonlinear MPC problem with reference tracking for lane changes. Section 4 shows simulation results obtained for a single vehicle. In section 5, the presence of vehicles in the target lane is considered. Section 6 shows simulation results obtained with the inclusion of vehicles in the target lane. Finally, section 7 presents the conclusions.

2. NONLINEAR SINGLE TRACK MODEL

The nonlinear vehicle model considered in this work is obtained from the linear Single Track Model (STM) found in (Rajamani, 2011) by incorporating two additional states that correspond to the position of the vehicle in global coordinates (Falcone et al., 2007). An schematic of the single track model is showed in figure 1. The STM describes the dynamics of the vehicle under the lateral forces acting on

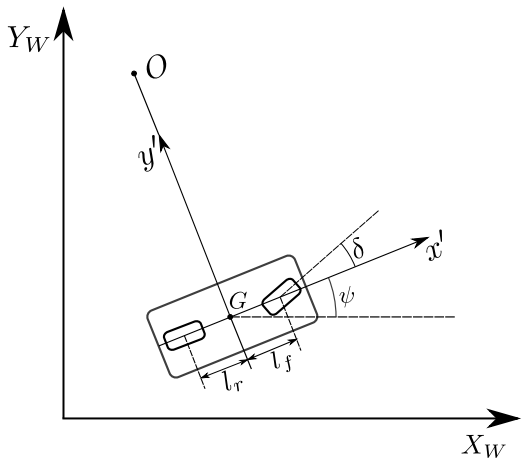


Fig. 1. The Single Track Model

it, considering a local reference frame whose coordinates are given by $[x', y']$.

This STM assumes that the longitudinal velocity of the vehicle is constant. Hence, the unidimensional input of the STM is the front wheel steering angle δ, and the states are given by $x = [y', \psi, \dot{y'}, \dot{\psi}, Y_W, X_W]^T$, being y' the lateral position of the vehicle to the instantaneous turning center O, ψ, its orientation angle and Y_W, X_W the position of the vehicle in global coordinates. The STM is described by the following dynamic equations:

$$\begin{aligned}
\ddot{y'} =& -\frac{2C_{\alpha f} + 2C_{\alpha r}}{m\dot{x'}}\dot{y'} \\
& - \left(\dot{x'} + \frac{2C_{\alpha f}l_f - 2C_{\alpha r}l_r}{m\dot{x'}}\right)\dot{\psi} + \frac{2C_{\alpha f}}{m}\delta \\
\ddot{\psi} =& -\frac{2l_fC_{\alpha f} - 2l_rC_{\alpha r}}{I_z\dot{x'}}\dot{y'} - \frac{2l_f^2C_{\alpha f} + 2l_r^2C_{\alpha r}}{I_z\dot{x'}}\dot{\psi} \\
& + \frac{2l_fC_{\alpha r}}{I_z}\delta \\
\dot{Y}_W =& \dot{x'}\sin(\psi) + \dot{y'}\cos(\psi) \\
\dot{X}_W =& \dot{x'}\cos(\psi) - \dot{y'}\sin(\psi)
\end{aligned} \tag{1}$$

where:

- m is the vehicle's mass.
- $\dot{x'}$ is the vehicle's velocity in direction of its longitudinal axis, assumed constant.
- I_z is the vehicle's yaw moment of inertia.
- l_f, l_r are the distances from the vehicle's center of mass G to its front and rear wheels, respectively.
- $C_{\alpha f}, C_{\alpha r}$ are the cornering stiffness for the front and rear wheels, respectively.

The set of equations (1) defines the nonlinear model $\dot{x} = f(x, u)$ to be used in the MPC controller. It is worth to note that small tire slip angles are assumed. Hence, the wheel's lateral forces acting on the vehicle can be considered as linear functions of the corresponding slip angles, with the cornering stiffness acting as the relating proportionality constant. The slip angle is defined as the angle between the velocity vector acting on the tire and its orientation. Moreover, small angle approximations are made in regard to the vehicle's slip angle, defined as the angle between the longitudinal axis of the vehicle and its velocity vector.

3. NONLINEAR MPC PROBLEM FORMULATION FOR LANE CHANGES

For controlling the lateral dynamics of the vehicle while tracking a reference for changing a lane with nonlinear MPC can be formulated, as follows:

$$\min_{u(k+1),\ldots,u(k+N)} \sum_{j=1}^{N} ||y_{ref} - y(k+j)||_Q^2 + ||u(k+j)||_R^2 \tag{2}$$

Subject to:

$$x(k+j+1) = f(x(k+j), u(k+j)) \tag{3}$$
$$y(k+j) = Y_W(k+j) \tag{4}$$
$$|u(k+j)| \leq u_{max} \ \ \forall j = 1, ..., N \tag{5}$$
$$|\Delta u(k+j)| \leq \Delta u_{max} \ \ \forall j = 1, ..., N \tag{6}$$

From the objective function (2), N is the prediction horizon given in number of samples and Q and R are the tuning matrices of the MPC controller. The model constraint (3) is obtained by discretizing the STM, given by the set of equations (1). It is assumed that the global coordinate Y_W of the vehicle can be measured, as stated in constraint (4). Finally, the operative constraints (5) and (6) give the maximum and minimum values for the front wheel steering angle and its maximum rate of change.

Parameter	Value	Units
m	1573	Kg
I_z	2873	Kg · m^2
l_f	1.10	m
l_r	1.58	m
C_{af}	80000	N · m/rad
C_{ar}	80000	N · m/rad
$\dot{x'}$	5.56	m/s
T_s	0.05	s
N	100	-
δ_{min}	-0.17	rad
δ_{max}	0.17	rad
$\Delta\delta_{max}$	1.5	rad/s

Table 1. Simulation parameters

4. SIMULATION RESULTS FOR A SINGLE VEHICLE

The MPC controller presented in section 3 has been tested in simulation using Matlab® and Simulink, based on the parameters found in Table 1. The parameters related to the vehicle are the same used by (Rajamani, 2011), and correspond to a passenger sedan. Satisfactory results were obtained with a sample time $T_s = 0.5$, though $T_s = 0.05$ could have been used, as in (Anderson et al., 2010), where real experiments have been made. The prediction horizon was chosen as $N = 10$, since the interest is on modelling lane changes. This value corresponds to 5s, and can be regarded as the approximate maximum duration of a lane change, according to (Toledo and Zohar, 2015). Finally, the, δ_{min}, δ_{max} and $\Delta\delta_{max}$ restrictions values have been taken from (Anderson et al., 2010) and (Falcone et al., 2007).

Figure 2 shows the trajectory obtained with the proposed MPC. A reference value of 3.3m in the Y_W state was set,

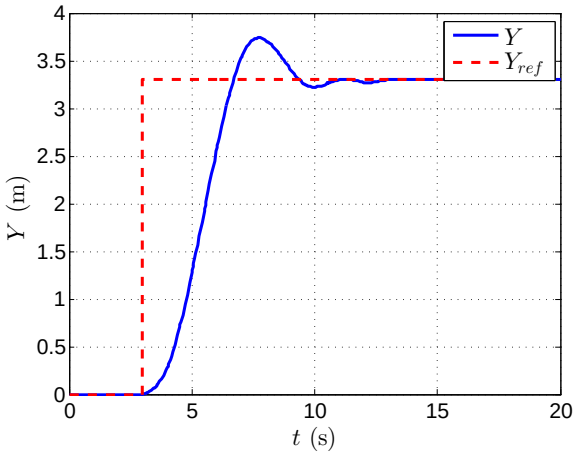


Fig. 2. Simulation results for the single vehicle scenario, showing the obtained trajectory

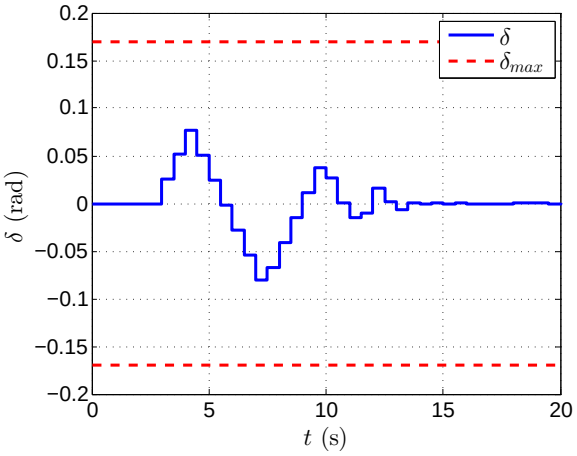


Fig. 3. Simulation results for the single vehicle scenario, showing the front wheel steering angle

which can be taken as the approximate distance between the center of two lanes. Figures 3 and 4 show the control action calculated by the MPC, its rate of change and the corresponding constraints.

5. INCLUDING VEHICLES IN THE TARGET LANE

Consider a scenario with one vehicle that wants to execute a lane change (called "subject vehicle", from now), identified with the superscript p, and vehicles in the target lane identified with the superscripts $p-1$ and $p+1$ as showed in figure 5. An additional constraint is required to prevent the subject vehicle to collision with the vehicles in the target lane, which can be defined, as follows. Let:

$$\mathbf{r_W^p}(k+j) = [X_W^p(k+j), Y_W^p(k+j)]^T \quad (7)$$

$$\mathbf{r_W^q}(k+j) = [X_W^q(k+j), Y_W^q(k+j)]^T \quad (8)$$

With $p \neq q$. Then:

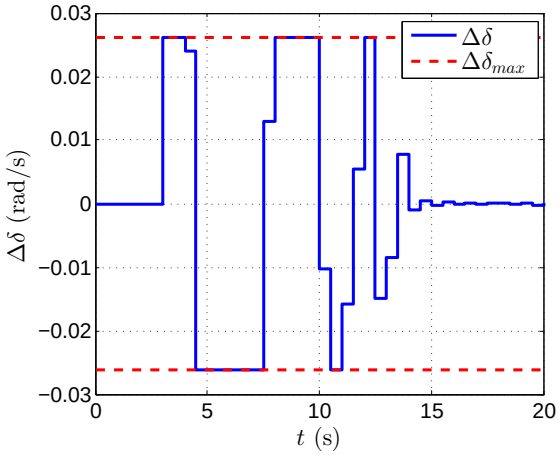


Fig. 4. Simulation results for the single vehicle scenario, showing the rate of change of the front wheel steering angle

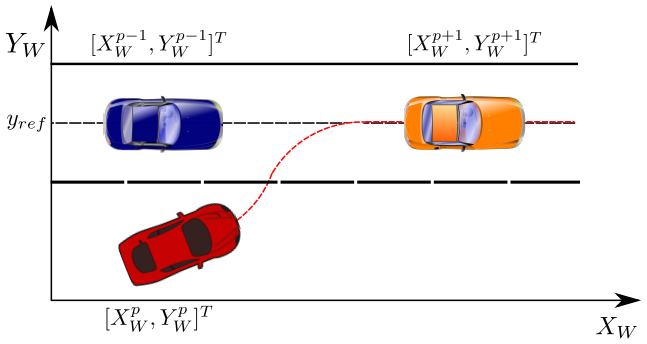


Fig. 5. Notation for the position of vehicles involved in a lane change

$$d_{p,q}(k+j) \geq d_{safe}, \forall j = 1, ..., N \quad (9)$$

With:

$$d_{p,q}(k+j) = \|\mathbf{r_W^p}(k+j) - \mathbf{r_W^q}(k+j)\| \quad (10)$$

Equations (9) and (10) state that the euclidean distance between the subject vehicle and the vehicles in the target lane must be greater than a given safety distance d_{safe}. Note that this approach assumes that the subject vehicle knows the position of those in the target lane, which could be achieved with technologies such as vehicular communications, DGPS and computer vision. Another interesting aspect is that the subject vehicle should accurately predict the position of vehicles in the target lane within the prediction horizon, which could be achieved through one of the many well-known car-following models (Li and Sun, 2012).

6. SIMULATION RESULTS CONSIDERING VEHICLES IN THE TARGET LANE

The strategy for incorporating vehicles in the target lane was tested in Matlab® and Simulink taking into account the same parameters of Table 1. The prediction of the position of vehicles in the target lane was simplified by assuming that they move at a constant speed of 5.56 m/s (20 km/h) in free flow conditions, i.e. they do not interact with each other. Additionally, the safe distance was chosen

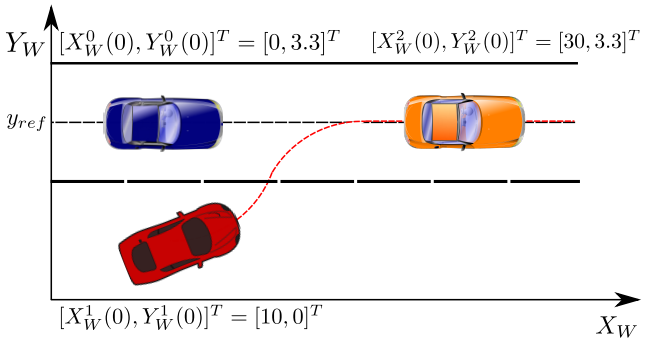


Fig. 6. Initial conditions for the simulation scenario that incorporates vehicles in the target lane

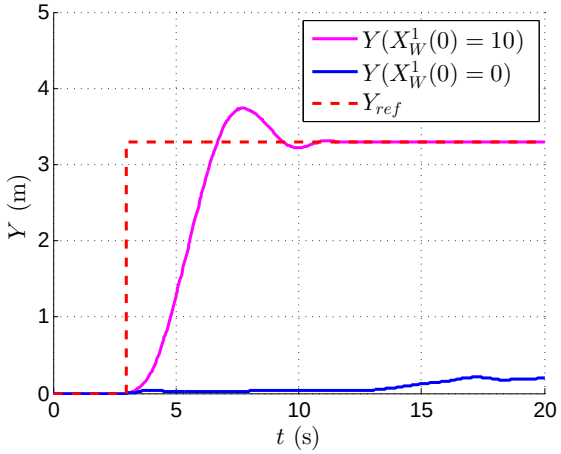


Fig. 7. Simulation results for the scenario that includes vehicles in the target lane, showing the obtained trajectory of the subject vehicle

as $d_{safe} = 2.5\,\text{m}$. Finally, the initial conditions for the simulation are showed in figure 6.

Figure 7 shows trajectories obtained for two cases: one with the initial conditions showed in figure 6, and another changing the initial condition of the subject vehicle to $[X_W^1(0), Y_W^1(0)]^T = [0, 0]^T$. In the second case note that, because the vehicle is moving at a constant speed, the initial condition does not allow it to execute the lane change, despite the MPC outputs some tries, as showed in Figure 8. Furthermore, a simulation of the second case disabling the distance constraint given in equation (9) was performed to show that this constraint is not met with one of the vehicles in the target lane, as showed in figure 9.

7. CONCLUSIONS

In this article, a nonlinear Model Predictive Control (MPC) of a passenger vehicle aimed for lane changes was presented. This MPC is based on a simplified Single Track Model (STM), including two states that describe the position of the vehicle in a global coordinate system, so that proper reference values can be tracked in order to execute the lane change maneuver. The effectiveness of the proposed MPC could be validated in simulation. Furthermore, an additional constraint was included in order to consider the influence of vehicles in the target

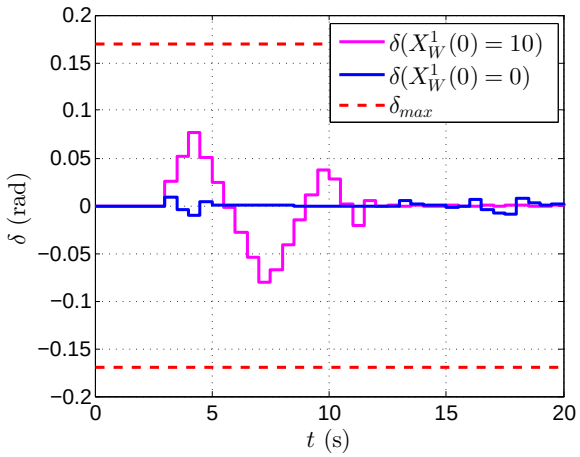


Fig. 8. Simulation results for the scenario that includes vehicles in the target lane, showing the obtained front wheel steering angle of the subject vehicle

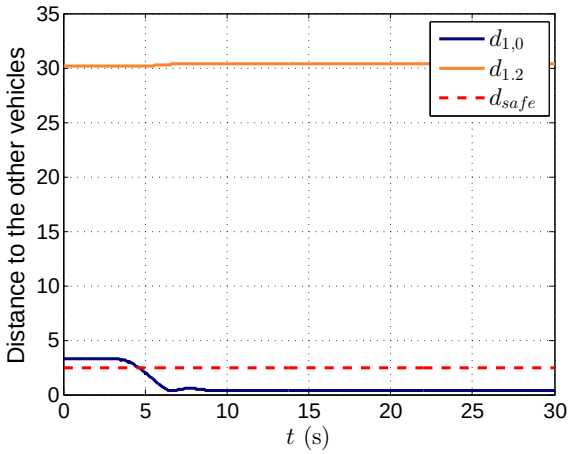


Fig. 9. Distance with vehicles in the target lane disabling the safe distance constraint

lane. Several simulations were performed to validate this strategy.

ACKNOWLEDGEMENTS

Authors would like to thank Proyecto Colciencias 111856 934640 contrato - FP44842-202-2015: Modelamiento y Control de tráfico urbano en la ciudad de Medellín Fase 2. Convocatoria 669.

REFERENCES

Anderson, S.J., Peters, S.C., Pilutti, T.E., and Iagnemma, K. (2010). An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios. *International Journal of Vehicle Autonomous Systems*, 8(2), 190–216.

Bifulco, G.N., Pariota, L., Simonelli, F., and Di Pace, R. (2013). Development and testing of a fully Adaptive Cruise Control system. *Transportation Research Part C: Emerging Technologies*, 29, 156–170. doi: 10.1016/j.trc.2011.07.001.

Falcone, P., Borrelli, F., Asgari, J., Tseng, H., and Hrovat, D. (2007). Predictive Active Steering Control for Autonomous Vehicle Systems. *IEEE Transactions on Control Systems Technology*, 15(3), 566–580. doi: 10.1109/TCST.2007.894653.

Gerdts, M., Karrenberg, S., Müller-Beßler, B., and Stock, G. (2009). Generating locally optimal trajectories for an automatically driven car. *Optimization and Engineering*, 10(4), 439–463.

Guvenc, B.A., Güvenç, B.A., and Karaman, S. (2009). Robust yaw stability controller design and hardware-in-the-loop testing for a road vehicle. *Vehicular Technology, IEEE Transactions on*, 58(2), 555–571.

Li, Y. and Sun, D. (2012). Microscopic car-following model for the traffic flow: the state of the art. *Journal of Control Theory and Applications*, 10(2), 133–143.

Maerivoet, S. and De Moor, B. (2005). Transportation planning and traffic flow models. *arXiv preprint physics/0507127*. URL `http://arxiv.org/abs/physics/0507127`.

Park, J.M., Kim, D.W., Yoon, Y.S., Kim, H.J., and Yi, K.S. (2009). Obstacle avoidance of autonomous vehicles based on model predictive control. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 223(12), 1499–1516.

Rajamani, R. (2011). *Vehicle dynamics and control*. Springer Science & Business Media.

Sen, B., Smith, J.D., and Najm, W.G. (2003). Analysis of lane change crashes. Technical report.

Toledo, T. (2007). Driving behaviour: models and challenges. *Transport Reviews*, 27(1), 65–84.

Toledo, T. and Zohar, D. (2015). Modeling duration of lane changes. *Transportation Research Record: Journal of the Transportation Research Board*.

Zheng, Z. (2014). Recent developments and research needs in modeling lane changing. *Transportation research part B: methodological*, 60, 16–32.

Nonlinear State Estimation for Batch Process with Delayed Measurements

Jhon A. Isaza * **Julián E. Rendón** ** **Juan Pablo Viana** ** **Héctor A. Botero** **

* *Faculty of Engineering and Architecture, Universidad Nacional de Colombia, Sede Manizales, Carrera 27 No 64-60, Colombia (e-mail: jaisazah@unal.edu.co)*
** *Department of Electrical Energy and Automatica, Universidad Nacional de Colombia, Sede Medellín, Carrera 80 No 65-223, Colombia (e-mail: jerendonr@unal.edu.co, jpvianav@unal.edu.co, habotero@unal.edu.co)*

Abstract: This paper presents a nonlinear state estimation subject to delayed measurement for the biomass in a batch bioprocess. The estimator scheme is based on an Extended Kalman Filter with state augmentation method to incorporate delayed measurements. A methodology to use the sample-state augmentation method is described. The proposed estimator is applied in the δ-endotoxins production of *Bacillus thuringiensis*. Simulation results show the feasibility of the proposed estimator.

Keywords: Batch Process, Extended Kalman Filter, Fixed Lag Smoothing, Delayed Measurements.

1. INTRODUCTION

The constant research and development in state estimation techniques have applications in electrical, electromechanical, navigation systems and now have been found a great potential for application in chemical and biotechnological processes (Mohd Ali et al., 2015). However, to achieve those practical applications is necessary to solve some problems from the academic field, such as the handling of nonuniform and delayed information.

For the handling of nonuniform and delayed information some authors have developed different methods in state estimation techniques (Gopalakrishnan et al., 2011; Guo and Huang, 2015; Guo et al., 2014; Patwardhan et al., 2012; Peñarrocha et al., 2012; Wang et al., 2012). The methods fall into two types: based on measurements fusion and based on augmented state. The based on measurement fusion method only applies to discrete systems and is designed for the Kalman filter and its variations. By contrast, the based on augmented state method retains the representation of the state space, making it more promising to facilitate its extension to different types of estimators. Furthermore, the conservation of the state space representation allows the subsequent analysis of features such as: convergence, observability and robustness.

However, in the mentioned papers the tools have been applied to specific problems, and therefore a few applications in batch bio-process with delayed measurement has been reported, except some cases with fed-batch bio-process without the dissolved oxygen dynamic (Zhao et al., 2015; Guo and Huang, 2015).

Therefore, this paper describes a methodology for incorporating delayed and multisampling measurements in nonlinear state estimation techniques, based in literature (Gopalakrishnan et al., 2011), to estimate the biomass in the batch production process δ-endotoxins of *Bacillus thuringiensis (Bt)* considering the dissolved oxygen concentration. As a technique of estimation a Filter Extended Kalman is used, however the methodology can be extended other to process.

The paper is organized as follows. In Section 2 the methodology for incorporate delayed measurement in estimation state techniques is described. Then in Section 3 is presented the mathematical model of δ-endotoxins production process of *Bt* followed by the estimator scheme based on a Extended Kalman Filter with sample-state augmentation method. The Section 4 presents simulation results of the estimation scheme for the δ-endotoxins production process of *Bt*. Finally, the conclusions of this paper are exposed in the Section 5.

2. METHODOLOGY FOR INCORPORATE DELAYED MEASUREMENT IN ESTIMATION STATE TECHNIQUES

2.1 Methods for incorporate delayed measurements

In a Supervisory Control And Data Acquisition (SCADA) system all information is stored discreetly, according the following assumptions:

Assumption 2.1. Sampling delays associated with online measurements are considered negligible compared to sampling delays associated with off-line measurements.

Assumption 2.2. The information obtained from the online sensors is more susceptible to problems of

noise and precision that obtained from laboratory or specialized equipment analysis. The first is subject to the characteristics of the signal conditioning system sensors. In the second, it is assumed strict adherence to high quality standards and metrology.

Assumption 2.3. The storage of offline information is subject to human error at check data to the SCADA system. Errors can be represented with spurious or missing data.

A characterization of phenomena occurring in the acquisition and storage of each source of information is presented in Figure 1.

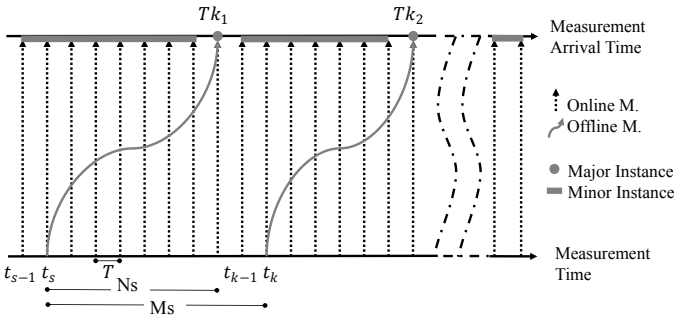


Figure 1. Characterization of phenomena occurring in the acquisition and storage of information sources. Source: modified by the author of (Guo and Huang, 2015).

In Figure 1, the lower horizontal line represents the time instants in which the measurements or sampling are performed. Moreover, the upper horizontal line represents time instant in which measurements are obtained and stored in the SCADA system. The straight, dashed and vertical lines represent the measurements online − obtained by the sensors −. It is noteworthy that these measurements are sampled with a sampling period fixed T and delay measurement is considered negligible. Offline measurements − obtained from the analysis of samples in laboratory or specialized equipment − are represented with a continuous and curve line. Note that each of the off-line measurements may have different measurement delays Tk_i, that is, the time between sample taken and measuring arrival $N_s = \delta + \tau_s$ is variant. M_s represents the time period between two successive samplings offline.

Moreover, to incorporate the different measurements on state estimation techniques two moments are presented: major instance and minor instance. Major instance refers to the moment in which both measures are available (online and offline). Minor instance refers to moment in which only available online measurement. When they used information sources with different characteristics, collateral problems such as: multi-sampling or asynchronism, missing and spurious data, redundant information, among others are presented (Guo and Huang, 2015; Guo et al., 2014).

In this sense, there are different methods to manage and incorporate nonuniform and delayed information in stochastic state estimation techniques. The methods can be classified into two types: fusing measurements and

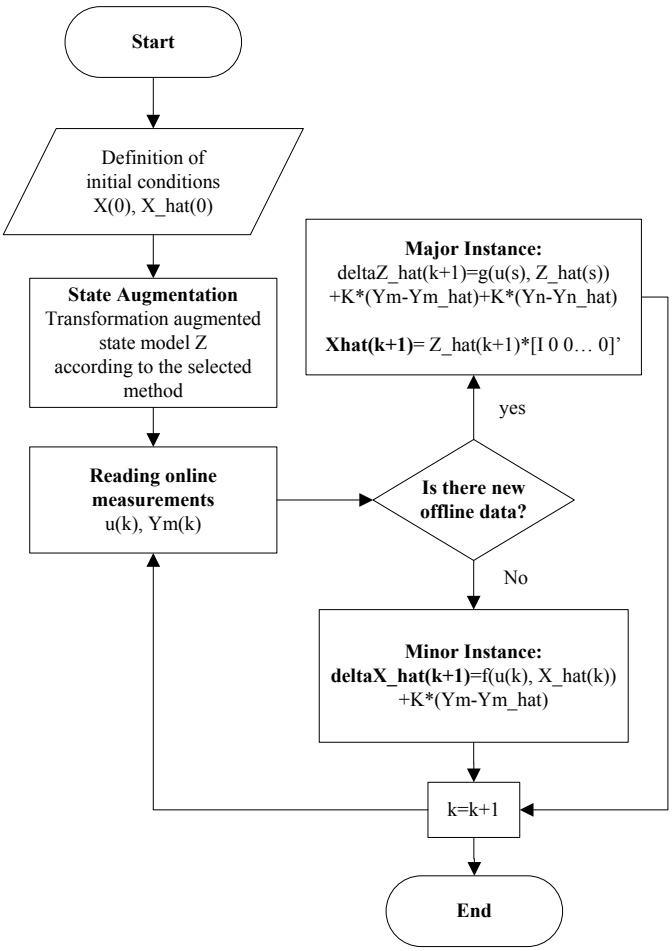


Figure 2. Flowcharts for programming algorithms by state augmentation methods. Source: Author's elaboration.

state augmentation (Gopalakrishnan et al., 2011; Guo and Huang, 2015; Guo et al., 2014; Patwardhan et al., 2012; Peñarrocha et al., 2012; Wang et al., 2012).

Fusing measurements method is based on the readjustment of the estimate in major instance. The readjustment of the present state is performed by recalculating entire trajectory of the Kalman filter (Prasad et al., 2002). The methods based on fusing measurements only applies to discrete systems and are conceived to the Kalman filter and its amendments (Alexander, 1991; Larsen et al., 1998).

On the other side, the augmented state method is based on increasing the state space with information of offline measurement and subsequently extended model is incorporated into the state estimation technique. As shown in the algorithm of Figure 2, in these methods is only modified the model and the representation of the state space is conserved. Because of this, these methods become favorable for its extension to different estimation and control techniques. In (Anderson and Moore, 2005) state augmentation is defined for fixed-lag smoothing method. In the Fixed-Lag Smoothing method, the N_s past states are smoothed based on online measurements of the minor instances. When offline and delayed measurement is obtained, both measurements (offline and online) are used to smooth out the state between s and $s + N_s$. Methods based on augmentation state, retains the state

space representation, favoring its extension to different types of estimators (Anderson and Moore, 2005; Simon, 2006) including deterministic estimation techniques.

2.2 Augmented state method

Next, a brief mathematical description of representing a linear system augmented state is presented based on (Gopalakrishnan et al., 2011).

Consider the following discrete-time linear system:

$$x(k+1) = A_k x(k) + B_k u(k) + \varepsilon(k) \tag{1}$$

for the system (1) nonuniform and delayed measurements can be represented as:

$$\begin{aligned} y^1(k) &= C_k^1 x_a(k) + v^1(k) \\ y^2(s) &= C_s^2 x_b(s) + v^2(s) \end{aligned} \tag{2}$$

where $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^l$ and $y^i(k) \in \mathbb{R}^m$ are states, inputs and outputs of the system respectively. $\varepsilon(k)$ and $v^i(k)$ correspond to uncertainty modeling and measurement noise respectively. A_k, B_k and C_k are the matrices representing the state space. Should be noted that the outputs $y^i(k)$ can be divided into online and offline measurements (time k and s, respectively).

Now, applying the concept of state augmentation, the order of the system can be increased such that contains the information of nonuniform and delayed measurements, as shown in Equation (3).

$$\begin{aligned} Z(k+1) &= \Phi_k Z(k) + \Gamma_k U(k) + \Psi_k \epsilon(k) \\ Y(k) &= \Xi_k Z(k) + v^1(k) \end{aligned} \tag{3}$$

where the matrices of the augmented state space Φ_k, Γ_k, $\Psi_k Q \Psi_k^T$ and Ξ_k are used in place of A_k, B_k, Q and C_k respectively. Using the fixed-lag smoothing method, the augmented state (4) and the augmented matrices (5) are defined as:

$$Z(k) = [x^T(k)\ x^T(k-1)\ \cdots\ x^T(k-Ns)]^T \tag{4}$$

$$\Phi = \begin{bmatrix} A_k & 0 & \cdots & 0 & 0 \\ I & 0 & \cdots & 0 & 0 \\ 0 & I & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & I & 0 \end{bmatrix}; \Gamma = \begin{bmatrix} B_k \\ 0 \\ 0 \\ \vdots \end{bmatrix};$$
$$\Psi = \begin{bmatrix} I \\ 0 \\ 0 \\ \vdots \end{bmatrix}; \Xi = \begin{bmatrix} C_k^1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & C_s^2 \end{bmatrix} \tag{5}$$

This representation can be extended to nonlinear systems. Consider the following discrete-time nonlinear system with nonuniform and delayed measurements:

$$\begin{aligned} x(k+1) &= f(x(k), u(k), \varepsilon(k)) \\ y^1(k) &= h_1(x_a(k), v^1(k)) \\ y^2(s) &= h_2(x_b(s), v^2(s)) \end{aligned} \tag{6}$$

where f, h_1 and h_2 are nonlinear functions.

As in the expression (4), the concept of state augmentation can be applied, and redefine the matrix representation as follows:

$$Z(k) = [f(x(k), u(k))^T\ x^T(k-1)\ \cdots\ x^T(k-Ns)]^T \tag{7}$$

$$\Phi^* = \begin{bmatrix} F_k^* & 0 & \cdots & 0 & 0 \\ I & 0 & \cdots & 0 & 0 \\ 0 & I & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & I & 0 \end{bmatrix}; \Xi = \begin{bmatrix} H_k^{1*} & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & H_s^{2*} \end{bmatrix} \tag{8}$$

where $F_k^* = \frac{\partial f}{\partial x}|_{(x(k),u(k))}$, $H_k^{1*} = \frac{\partial h_1}{\partial x}|_{(x(k),u(k))}$ and $H_s^{2*} = \frac{\partial h_2}{\partial x}|_{(x(s),u(s))}$, are the Jacobian matrices for the system (6).

This method results in smoothing of the past N_s states based on the online measurements at the minor time instance. When the delayed offline measurement arrives, both offline and online measurements are used to obtain smoothed estimates from s to $s + N_s$.

In this method, the representation of the state space is preserved, so the method can be applied with different state estimation techniques. Following the application of the methodology for estimating biomass in the δ-endotoxins production of *Bt* it is proposed. As state estimation technique a Kalman Filter Extended (KFE) is used.

3. STATE ESTIMATION IN A BATCH PROCESS MODEL WITH DELAYED MEASUREMENTS

The model of the δ-endotoxins production of *Bt* proposed on (Amicarelli et al., 2010, 2013; Rómoli et al., 2016) is used. In this paper the nomenclature of some parameters are modified. The model equations are:

$$\begin{aligned} \dot{s}_p &= -\left(\frac{\mu}{y_{x/s}} + m_s\right) x_v \\ \dot{o}_d &= K_3 Q_A \left(o_d^* - o_d\right) - \left[K_1 \dot{x}_T + K_2 x_T\right] \\ \dot{x}_v &= \left(\mu - k_s - k_e(t)\right) x_v \\ \dot{x}_s &= k_s x_v \end{aligned} \tag{9}$$

where s_p is the substrate concentration, o_d is the dissolved oxygen concentration, x_v is the vegetative cells concentration, x_s is the sporulated cells concentration, μ is the specific growth rate, $y_{x/s}$ is the growth yield, m_s is the maintenance constant, Q_A is the airflow that enters the bioreactor, o_d^* is the oxygen saturation concentration, K_1 is the oxygen consumption dimensionless constant by growth, K_2 is the oxygen consumption constant for maintenance, K_3 is the ventilation constant, k_s is the spore formation kinetics and $k_e(t)$ is the specific cell death rate.

Furthermore, the constitutive equations for μ (Monod-based), k_s and k_e are given by:

$$\begin{aligned} \mu &= \mu_{\max} \frac{s_p}{K_s + s_p} \frac{o_d}{K_o + o_d} \\ k_s &= k_{s,\max} \left(\frac{1}{1 + e^{G_s(s_p - P_s)}} - \frac{1}{1 + e^{G_s(s_{p,\mathrm{ini}} - P_s)}}\right) \\ k_e(t) &= k_{e,\max} \left(\frac{1}{1 + e^{-G_e(t - P_e)}} - \frac{1}{1 + e^{-G_e(t_{\mathrm{ini}} - P_e)}}\right) \\ x_T &= x_s + x_v \end{aligned} \tag{10}$$

where $\mu_{\max}$ is the maximum specific growth rate, K_s is the substrate saturation constant, K_o is the oxygen saturation

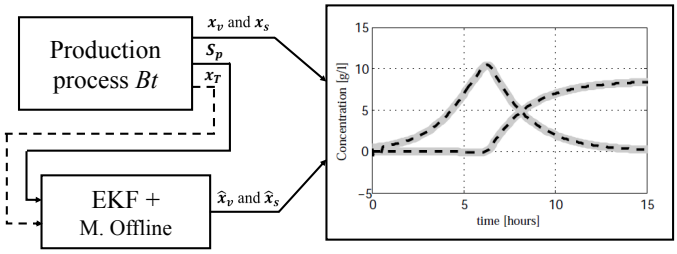


Figure 3. EKF scheme. Source: modified by the author of (Amicarelli et al., 2013).

constant,$k_{s,\max}$ is the maximum spore formation, $k_{e,\max}$ is the maximum specific cell death rate, G_s is the gain constant of the sigmoid equation for spore formation rate, G_e is the gain constant of the sigmoid equation for specific cell death rate, P_s is the position constant of the sigmoid equation for spore formation rate, P_e is the position constant of the sigmoid equation for specific cell death rate, $s_{p,\text{ini}}$ is the initial glucose concentration and t_{ini} is the initial fermentation time.

The nominal parameters for the system (9)-(10) are given in Table 1.

Table 1. Nominal Parameters of the *Bt* model.

Parameter	Values	Unit
$\mu_{\max}$	0.65	h^{-1}
$y_{x/s}$	0.37	$\text{g} \cdot \text{g}^{-1}$
K_s	3	$\text{g} \cdot \text{L}^{-1}$
K_o	1×10^{-4}	$\text{g} \cdot \text{L}^{-1}$
m_s	5×10^{-3}	$\text{g} \cdot \text{g}^{-1} \cdot \text{h}^{-1}$
$k_{s,\max}$	0.5	h^{-1}
G_s	1	$\text{g} \cdot \text{L}^{-1}$
P_s	1	$\text{g} \cdot \text{L}^{-1}$
$k_{e,\max}$	0.1	h^{-1}
G_e	5	h
P_e	4.9	h
K_1	3.795×10^{-3}	dimensionless
K_2	0.729×10^{-3}	h^{-1}
K_3	2.114×10^{-3}	L^{-1}
Q_A	1320	$\text{L} \cdot \text{h}^{-1}$
o_d^*	0.00759	$\text{g} \cdot \text{L}^{-1}$
t_{ini}	0	h
$s_{p,\text{ini}}$	32	$\text{g} \cdot \text{L}^{-1}$

Moreover for the estimation scheme proposed it is assumed that the measurement of the output s_p is online and the output x_T is offline (see in Figure 3). That is, the output s_p is sampled without delay and at the same rate of numerical solution of the EKF. Moreover, the output x_T is measured every M_s sampling times and also is obtained with a delay of N_s sampling times. In this paper it is used a common Extended Kalman Filter (EKF) of two steps. The equation (11) show the step of prediction and the equation (12) show the step of updating of the state respectively. However the methodology can be applied any extension of the Kalman filter.

$$\begin{aligned} \hat{x}(k|k-1) &= f(\hat{x}(k-1|k-1), k) \\ P(k|k-1) &= A(k) P(k|k-1) A^T(k) + Q \end{aligned} \tag{11}$$

$$\begin{aligned} \tilde{y}(k) &= y(k) - H\hat{x}(k|k-1) \\ K(k) &= P(k|k-1) H^T \left[HP(k|k-1) H^T + R\right]^{-1} \\ \hat{x}(k|k) &= \hat{x}(k-1|k-1) + K(k) \tilde{y}(k) \\ P(k|k) &= (I + K(k) H) P(k|k-1) \end{aligned} \tag{12}$$

As shown in the flowchart of Figure 2, during the minor instance, the gain of the Kalman filter K and the covariance matrix P are calculated with the linearized model without increasing state space of the system (9). Otherwise, during the major instance, are calculated with the linearized model of augmented state space (5) of the system (9). In the next section the simulation results are presented.

4. SIMULATION RESULTS

The numerical simulation results of the proposed estimation structure applied to a model of δ-endotoxins production of *Bacillus thuringiensis (Bt)* are presented in this section. All simulations presented here were conducted using the Euler integration method, with a fundamental step size of $0.1[h]$ for a total time of simulation of $15[h]$. The model parameters are shown on Table 1. The parameters shown in this table were taken according to the range to $20\,[\text{g}\cdot\text{L}^{-1}] < s_{p,\max} < 32\,[\text{g}\cdot\text{L}^{-1}]$ (Amicarelli et al., 2010). The value $s_{p,\max}$ corresponds to the initial condition of s_p since $\dot{s}_p \leq 0$. It is considered that the substrate s_p is measured online every $0.1[h]$. The total biomass x_T is measured offline with a delay interval measurement $N_s = 0.5[h]$ and successive sampling interval $M_s = 0.7[h]$. In addition, the initial time where the offline measurement is sampled is $s(0) = 0.1[h]$. In other words, in the minor instance it is only available the measurement of s_p and in the major instance the measurements of s_p and x_T is available.

The initial conditions for the model were selected to: $s_p(0) = 32\ [\text{g} \cdot \text{L}^{-1}]$, $o_d(0) = 0.74 \times 10^{-2}\ [\text{L} \cdot \text{h}^{-1}]$, $x_v(0) = 0.645\ [\text{g} \cdot \text{L}^{-1}]$ and $x_s(0) = 1 \times 10^{-5}\ [\text{g} \cdot \text{L}^{-1}]$; and for the EKF to: $\hat{s}_p(0) = s_p(0)\ [\text{g}\cdot\text{L}^{-1}]$, $\hat{o}_d(0) = o_d^*\ [\text{L}\cdot\text{h}^{-1}]$, $\hat{x}_v(0) = 10x_v(0)\ [\text{g} \cdot \text{L}^{-1}]$ and $\hat{x}_s(0) = 10x_s(0)\ [\text{g} \cdot \text{L}^{-1}]$.

Finally, initializing the elements of the matrices P, Q and R for the two filters (with and without delayed measurements) were empirically adjusted to obtain the best fit possible so that the results are comparable (see (13)-(16)).

$$P_{0|0} = diag\left(\left[10^{-1}\ 10^{-1}\ 10^{-1}\ 10^{-7}\right]\right) \tag{13}$$

$$Q = diag\left(\left[10^{-2}\ 10^{-2}\ 1\ 10^{-4}\right]\right) \tag{14}$$

$$R_1 = 1 \tag{15}$$

$$R_2 = 10^{-2} \tag{16}$$

Figures 4, 5, 6, 7 and 8 show the comparison between the actual variables x, and estimated variables $\hat{x}$ only with the on-line measurement and estimated variables with online and off-line measurements. In these figures *measured* means data simulations with noise and *real* means data simulations without noise. The figures corresponding to substrate concentration s_p, dissolved oxygen concentration o_d, vegetative cell concentration x_v and sporulated cells concentration x_s respectively.

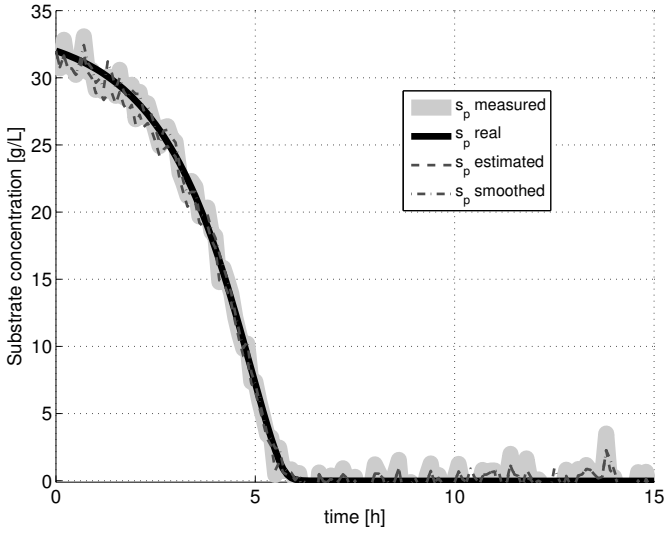


Figure 4. Substrate concentration s_p.

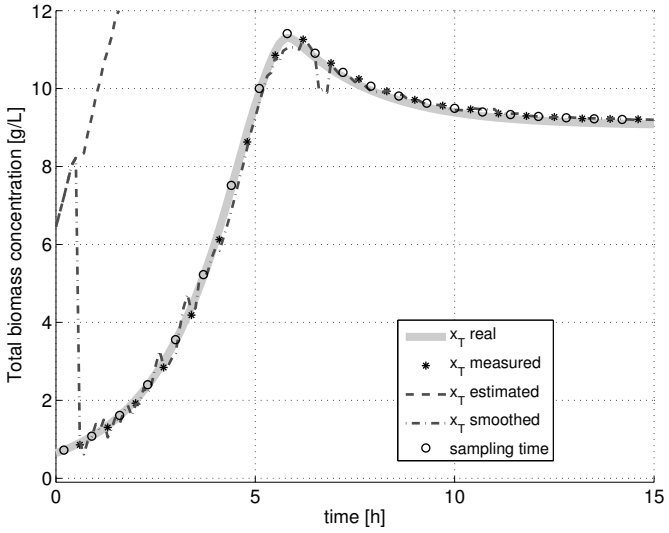


Figure 5. Total biomass concentration x_T.

In Figure 4 is possible to observe the correct estimation of the substrate concentration s_p, but this variable is not important to estimate because can be measured online. However, it is possible to notice the effect of filtering performed by the estimator in the variable s_p. In Figure 5, the EKF without off-line measurements produces an estimate that deviates from the actual value since the beginning, however, the EKF with offline measurements converges quickly when such measurement arrives. This tendency to converge continuing with each new additional off-line measurement. Additionally, Figures 6, 7 and 8 show a similar behavior is observed to the other estimated variables. In these figures, arrival times and sampling time of off-line measurement are detailed in order to visualize the effect of the off-line measurement in the EKF improved.

Finally, it was observed that it is possible to estimate the concentrations online in a batch process that includes the dynamic of dissolved oxygen concentration. The results can be used for monitoring, control or fault detection in batch processes.

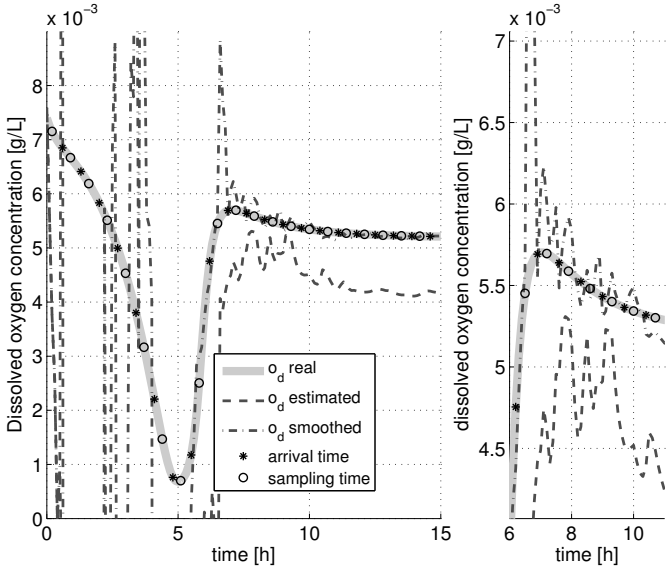


Figure 6. Dissolved oxygen concentration o_d.

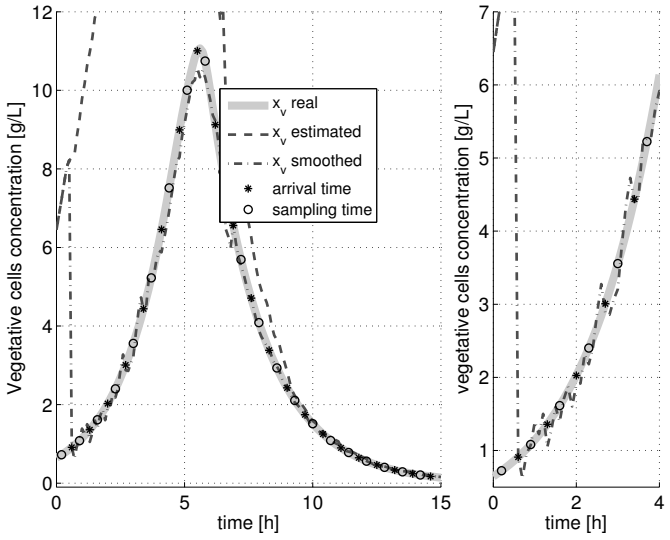


Figure 7. Vegetative cells concentration x_v.

5. CONCLUSIONS

In this paper was presented a nonlinear state estimation subject to delayed measurement for the biomass in a batch bioprocess including the dissolved oxygen dynamic. It was described the methodology for to use sample-state augmentation method. The proposed estimator was applied to the δ-endotoxins production of *Bacillus thuringiensis* and the incorporating of delayed measurements to EKF improved the performance of the estimate. Simulations show the feasibility of the proposed estimator.

REFERENCES

Alexander, H.L. (1991). State estimation for distributed systems with sensing delay. *Data Structures and Target Classification, SPIE*, 1470, 103–111. doi: 10.1117/12.44843.

Amicarelli, a., Di Sciascio, F., Toibero, J.M., and Alvarez, H. (2010). Including dissolved oxygen dynamics into

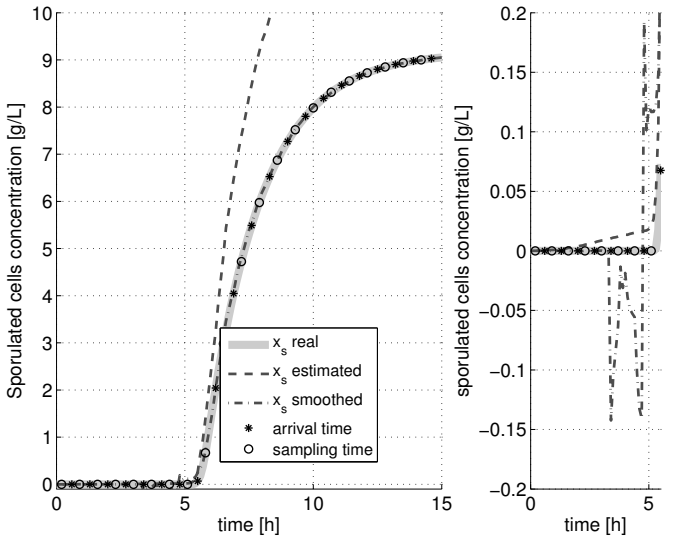


Figure 8. Sporulated cells concentration x_s.

the *bt* δ-Endotoxins Production process model and its application to process control. *Brazilian Journal of Chemical Engineering*, 27(1), 41–62.

Amicarelli, A., Quintero, O., and di Sciascio, F. (2013). Behavior comparison for biomass observers in batch processes. *Asia-Pacific Journal of Chemical Engineering*. doi:10.1002/apj.

Anderson, B.D.O. and Moore, J.B. (2005). *Optimal Filtering*. Dover Books on Electrical Engineering. Dover Publications.

Gopalakrishnan, A., Kaisare, N.S., and Narasimhan, S. (2011). Incorporating delayed and infrequent measurements in Extended Kalman Filter based nonlinear state estimation. *Journal of Process Control*, 21(1), 119–129. doi:10.1016/j.jprocont.2010.10.013.

Guo, Y. and Huang, B. (2015). State estimation incorporating infrequent, delayed and integral measurements. *Automatica*, 58, 32–38. doi:10.1016/j.automatica.2015.05.001.

Guo, Y., Zhao, Y., and Huang, B. (2014). Development of soft sensor by incorporating the delayed infrequent and irregular measurements. *Journal of Process Control*, 24(11), 1733–1739. doi:10.1016/j.jprocont.2014.09.006.

Larsen, T., Andersen, N., Ravn, O., and Poulsen, N. (1998). Incorporation of time delayed measurements in a discrete-time Kalman filter. *Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No.98CH36171)*, 4(December), 3972–3977. doi:10.1109/CDC.1998.761918.

Mohd Ali, J., Ha Hoang, N., Hussain, M., and Dochain, D. (2015). Review and classification of recent observers applied in chemical process systems. *Computers & Chemical Engineering*, 76, 27–41. doi:10.1016/j.compchemeng.2015.01.019.

Patwardhan, S.C., Narasimhan, S., Jagadeesan, P., Gopaluni, B., and L. Shah, S. (2012). Nonlinear Bayesian state estimation: A review of recent developments. *Control Engineering Practice*, 20(10), 933–953. doi:10.1016/j.conengprac.2012.04.003.

Peñarrocha, I., Sanchis, R., and Romero, J. (2012). State estimator for multisensor systems with irregular sampling and time-varying delays. *International Journal of Systems Science*, 43(8), 1441–1453. doi:10.1080/00207721.2011.625482.

Prasad, V., Schley, M., Russo, L.P., and Wayne Bequette, B. (2002). Product property and production rate control of styrene polymerization. *Journal of Process Control*, 12(3), 353–372. doi:10.1016/S0959-1524(01)00044-0.

Rómoli, S., Amicarelli, A.N., Ortiz, O.A., Scaglia, G.J.E., and di Sciascio, F. (2016). Nonlinear control of the dissolved oxygen concentration integrated with a biomass estimator for production of Bacillus thuringiensis δ-endotoxins. *Computers & Chemical Engineering*, 93, 13–24. doi:10.1016/j.compchemeng.2016.05.017.

Simon, D. (2006). *Optimal State Estimation: Kalman, H, and Nonlinear Approaches*. doi:10.1002/0470045345.

Wang, W., Huang, X.h., and Wang, M. (2012). Survey of sequence measurement filtering algorithm. *Control and Decision*, 1, 2.

Zhao, L., Wang, J., Yu, T., Chen, K., and Liu, T. (2015). Nonlinear state estimation for fermentation process using cubature Kalman filter to incorporate delayed measurements. *Chinese Journal of Chemical Engineering*, 23(11), 1801–1810. doi:10.1016/j.cjche.2015.09.005.

Nonlinear state estimation using online FTIR spectroscopy in polymerization processes

C. Zuluaga-Bedoya * **and J. Garcia-Tirado** *

* *Instituto Tecnológico Metropolitano, Calle 73 No 76A - 354, 050034, Medellín - Colombia. Facultad de Ciencias Económicas y Administrativas, Grupo de Calidad, Metrología y Producción, (email: zuluagabedoya@gmail.com, josegarcia@itm.edu.co).*

Abstract: Fourier-Transform Infrared spectroscopy (FTIR) is a powerful tool for inferring chemical composition of a sample from the analysis of its infrared spectrum of absorption or emission. The industrial usage of this analyzer is typically reduced to species characterization, without taking advantage of the potential in quantitative analysis. In the case of the polymers, there are some variables related with product quality, which are strongly related to molecular properties of polymer chains, such as Molecular Weight Distribution (MWD). Using state estimation techniques and monomer concentration measurements is possible to obtain an average of molecular weight distribution. A comparison using a nonlinear state estimation technique, the Extended Kalman Filter (EKF), is presented using either the reactor temperature or the monomer concentration as measured variable under different scenarios of noise and sampling time. The EKF based on monomer concentration produced better estimates according with the used performance indexes.

Keywords: State estimation, Extended Kalman Filter, Optical spectroscopy, Polymerization, Average molecular weight distribution, Process monitoring.

1. INTRODUCTION

Polymers have some variables related with product quality (rheological properties, mechanical strength, chemical resistance, thermal stability, etc.), and they are strongly related to molecular properties of polymer chains, such as Molecular Weight Distribution (MWD), Chain Sequence distribution (CSD) and Long-Chain Branching (LCD) (Apostolos et al., 2006). Therefore, an effective control of these properties would lead to obtain the desired product specifications. In order to accomplish control tasks, measurements must be carefully selected. However, these measurements are not often available. As Santos and coworkers pointed out in (Santos et al., 2005), the lack of instruments able to measure and monitor the quality of the polymer resin has been recognized as one of the most important problems in the field of polymerization reactor control.

State estimators are used when a complete measurement of the process variables is not feasible, which occurs when instruments generate high investment costs or do no meet the process requirements, that is, when there exist corrosive environments, signal transport issues, or even, when there is no technology for account some variables.

The main purpose of this work is to make a connection between FTIR and nonlinear state estimation techniques to account for molecular chain properties that define the quality of the products. With a proper online monitoring of these variables, such the average molecular weight, more efficient control loops can be implemented in the industry. In this paper, a polymerization of MMA benchmark is used in order to test the behavior of two EKF formulations for the estimation of the product quality variables by incorporating the composition measurements from the FTIR.

The paper is organized as follows: in Section 2, an introduction about FTIR spectroscopy is presented, highlighting its principal features and potential use in online monitoring. Then, an overview of literature for online monitoring of average molecular weight in polymerization processes and an estimation scheme with FTIR measurements is shown in Section 3. Section 4 contains a MMA free-radical polymerization case study with two EKF applications using different measurements. Final comments are given in Section 5.

2. FTIR TECHNOLOGY IN STATE ESTIMATION

Strictly, FTIR is a subdivision of Near-Infrared spectroscopy (NIR). This technology was developed to overcome limitations of dispersive spectrometers. The absorptive signals are collected simultaneously for the whole wavelength spectrum through the use of an interferometer. This advantage compared to dispersive NIR spectrometers is the reason of faster scanning speed, higher detector sensibility, simpler mechanical design, internal wavelength calibration, constant resolution, and negligible stray light. Consequently FTIR allows a more wide use in the chemical process industry (Santos et al., 2005).

In Figure 1, a typical FTIR spectrum of poly methyl methacrylate (PMMA) is shown. The spectrum has some representative peaks, according to the chemical groups present in the sample. Each chemical group has an absorption band at a characteristic wavenumber. The peak area can be related to the component concentration by using the Beer-Lambert law. However, in industrial applications, multiple components are present in a sample. Therefore, multivariate statistical methods are required, which are enclosed in a discipline called chemometrics. Appropriated experiments must be conducted in order to develop a chemometric model, including: pre-calibration, calibration,

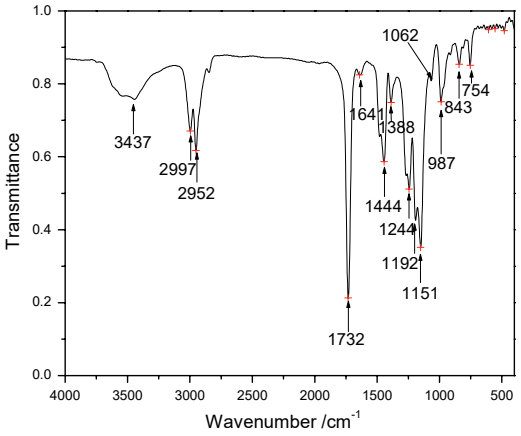


Fig. 1. FTIR spectrum of PMMA (Duan et al., 2008).

online validation, and model maintenance (Feng et al., 2015). Chemometric models are based on multivariate statistics and are usually identified using partial least-squares (PLS), principal component regression (PCR) or methods based on Petri nets. Depending on structure of the calibration models, recalibration periods should be specified.

3. PROCESS MONITORING USING ONLINE SPECTROSCOPY

It is well-known that most industries still have challenges related to measurement and control issues. For instance, the oil sand industry requires online measurements of many chemical and physical properties such as bitumen, water content, clays fraction, chloride concentration, viscosity, etc. The first challenge is to develop in-situ probes resistant to aggressive environments (erosion and corrosion) reducing replacing time. Another challenge is related to presence of multi-phase flows and process units where interface measurements are required. Also, there are some complex compositions in the streams, characterized by different particle size distributions (PSD). The last challenge is about the timing of lab data analysis, which are not suitable for online applications as real-time process control and optimization (Feng et al., 2015). Applications using FTIR including polymerization processes are presented as follows.

3.1 FTIR applications

Applications of FTIR in the chemical process industry are focused in polymerization and crystallization reactions, where some internal properties such particle size distribution (PSD), molecular weight distribution (MWD) or crystal size distribution (CSD) are fundamental to assess the quality of the products. However, there are additional industries where FTIR is successfully employed, such as the pharmaceutic industry, medicine, and atmospheric gas monitoring. In (Gallignani et al., 2014), a quality control procedure using FTIR to determine concentration of furosemide is presented. This work emphasize the advantages of FTIR compared to common pharmaceutical analysis. Other applications are found in the area of gas observations in greenhouses and FTIR imaging for histopathology in prostate cancer. Additionally to monitoring and control purposes, FTIR spectrometry can be useful for the study of kinetics in high-throughput reactive systems. In polymerization, it is a powerful technique for determination of catalyst activity, copolymer concentration, degree of homogeneity, among other properties.

3.2 FTIR in polymerization

Polymers have some variables related to the product quality (rheological properties, mechanical strength, chemical resistance, thermal stability, etc.), and these in turn are strongly related to molecular properties of polymer chains, such as Molecular Weight Distribution (MWD), Chain Sequence distribution (CSD) and Long-Chain Branching (LCD) (Apostolos et al., 2006). In this sense, process control is being redirected to the control of these properties in order to produce desired product specifications.

Due to the complexity of most polymerization processes, control algorithms require the use of reliable process models, which can provide useful insight about molecular properties of polymer chains. Molecular properties are typically studied by means of population balances at the expense of a considerable computational burden which is not feasible in monitoring and control applications. Consequently, there are some methods to approximate a model based on population balances, including the method of moments, that is the most applied, but some alternatives like orthogonal collocation method or fixed pivot methods have been used with prominent results (Apostolos et al., 2006).

Estimation techniques in polymerization are focused in the use of Kalman filtering strategies. The work of Kiparissides and coworkers presented an online optimizing control of molecular weight properties in a batch polymerization reactor (Kiparissides et al., 2002). The strategy used a state/parametric estimation step. In the first case, an Extended Kalman filter (EKF) produced an estimate of the state variables and the termination rate constant, which is a time-varying kinetic parameter. In a different contribution, a comparison among three formulations of the EKF and a nonlinear moving horizon estimator (MHE) is made to predict the concentration of impurities at the beginning of the batch using the complete nonlinear model and available measurements (Salau et al., 2014). The results showed acceptable results for the MHE at expenses of high computational burden. Similar approaches has been studied in (Shahrokhi and Fanaei, 2002), where a EKF was implemented for MWD estimation. Then, a cascade control loop was designed, with molecular weight $\overline{M}_w$ as controlled variable of the master loop and the reactor temperature T as controlled variable of the slave loop.

In (Shahrokhi and Fanaei, 2001) a discussion about the drawbacks and implementation issues of the EKF is presented. In this contribution, the importance of the observability condition, and its fulfillment for a proper estimation is remarked. However, the authors also remarked the possibility of designing a reduced-order state estimator when the observability condition is not met. The observable states are estimated by an EKF and the remaining states by an open-loop observer if the detectability property is guaranteed. This observability issues are found in most polymerization cases, because moments of dead polymer are not present in measurable properties.

The control of molecular properties of polymers are tackled using feedforward loops in (Graichen et al., 2005), where an EKF is designed in order to estimate the reaction heat and heat transfer coefficients. Again, as molecular properties moments

are not observable, they are calculated using an open loop observer, which is fed by measurements and estimated states computed by the EKF.

In the work of (Othman et al., 2004), a complete study of the dynamic evolution of styrene suspension polymerization is presented, using near infrared spectroscopy to feed a nonlinear control of molecular weight. Average molecular weight and monomer concentration are estimated by a near infrared spectrometer. The authors remark the feasibility of spectroscopy as a complement or possible substitute of gel permeation chromatography (GPC) analysis for molecular weight measurements as well as techniques to measure component concentrations (high pressure liquid chromatography- HPLC), because these methods have an appreciable lag between sampling and availability of the measurement in spite of their online feature. This work also remarked the need to estimate instantaneous molecular weight while polymerization takes place. The estimation procedure requires off-line and online calibration using PLS technique. Once near-infrared spectrometer measures are validated, a process model for styrene suspension polymerization and kinetics relations were used to obtain expressions for instantaneous molecular weight and average molecular weight. Afterwards, high gain observers were designed for some kinetic parameters as the reaction rate R_p and the ratio k_f/k_p. Also, effects of monomer, initiator, and solvent concentrations on the average molecular weight were analyzed. Finally, the authors selected monomer concentration (monomer inlet flow rate) as a manipulated variable for instantaneous molecular weight control. Feedback performance initially depended on convergence of high gain observers, but once stabilized, the strategy permitted the production with approximately constant molecular weight and polydispersity index, which is very attractive for industrial applications.

3.3 Monomer measurements

Some contributions related to polymerization monitoring and control use monomer concentration measurements instead of temperature as measured variables of the state estimator. This is because information about reaction kinetics is better transferred from the former source of information. In the work of (Eaton and Ricker, 1995) an Extended Kalman Filter is developed for tracking the particle size trajectory using online composition measurements. In (Ling and Kravaris, 2016), a state observer is designed for a series of polycondensation reactors. The authors proposed two measurements sources, with different sampling times. Also a correction for inter-sample behavior is implemented as well as Smith predictor for dead time issues.

In the present work, the Extended Kalman Filter (EKF) is used due to its predominant use in most of industrial applications regarding specially to chemical processes (Shenoy et al., 2010; Kiparissides et al., 2002). Previous works showed adequate performances of the EKF when plant-model mismatch is negligible under Gaussian uncertainties, therefore, an appropriate process model is of utmost importance for an acceptable filter performance.

4. CASE STUDY

The selected case study is the free-radical MMA polymerization studied in (Monsalve-Bravo et al., 2014) and it is considered a benchmark in polymerization modeling. This process

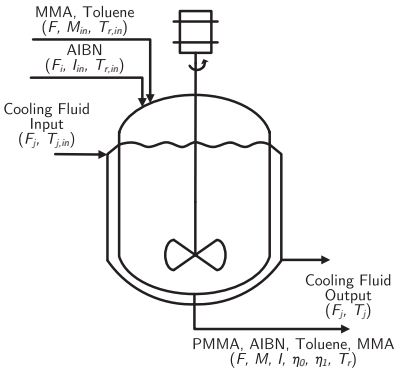


Fig. 2. Process flow diagram for MMA polymerization reactor (Monsalve-Bravo et al., 2014)

was selected due to the importance of polymer chain properties, specially, the average molecular weight. Typically, the reactor temperature is used as measured variable. In the present work, the behavior of the designed EKF will be compared using two possible sources of information, i.e., temperature and monomer composition from a FTIR.

4.1 Process description

The process is carried out in a continuous stirred tank reactor (CSTR) with azo-bis-isobutyronitrile (AIBN) as initiator and toluene as a solvent. This reaction is highly exothermic and the generated heat is removed through a cooling jacket. The process is performed in a heterogeneous phase. Figure 2 shows the process flow diagram for the reactor. This polymerization process keeps a set of reactions that take place in series when only a polymer chain is analyzed and in parallel when focus is in the whole reaction medium.

Initiation:	$I \xrightarrow{k_0} 2R^*$
	$R + M \xrightarrow{k_I} P_1$
Propagation:	$P_i + M \xrightarrow{k_0} P_{i+1}$
Monomer transfer:	$P_i + M \xrightarrow{k_{fm}} P_i + D_i$
Addition termination:	$P_i + P_j \xrightarrow{k_{tc}} D_{i+j}$
Disproportionation termination:	$P_i + P_j \xrightarrow{k_{td}} D_i + D_j$

Regarding to the energy expenditure, the propagation step is the most sensitive since the generated reaction heat is greater than the reaction heats of the remaining reactions. Moreover, monomer transfer and termination stages have higher impact in the molecular weight distribution.

4.2 Mathematical model

Following assumptions of (Daoutidis et al., 1990), conservation principles around reactor vessel and jacket are applied, giving the following material and energy balances:

$$\frac{dM}{dt} = -(k_p + k_{fm})MP_0 + \frac{F(M_{in} - M)}{V} \quad (1)$$

$$\frac{dI}{dt} = -k_I I + \frac{F_I I_{in} - FI}{V} \quad (2)$$

$$\frac{d\eta_0}{dt} = (0.5k_{tc} + k_{td})P_0^2 + k_{fm}MP_0 - \frac{F\eta_0}{V} \quad (3)$$

$$\frac{d\eta_1}{dt} = M_{w,m}(k_p + k_{fm})MP_0 - \frac{F\eta_1}{V} \quad (4)$$

$$\frac{dT}{dt} = \frac{(-\Delta H)k_p M}{\rho C_p}P_0 - \frac{UA}{\rho C_p V}(T - T_j) + \frac{F(T_{in} - T)}{V} \quad (5)$$

$$\frac{dT_j}{dt} = \frac{F_j(T_{j,in} - T_j)}{V_j} + \frac{UA}{\rho_j C_{p,j} V_j}(T - T_j) \quad (6)$$

with the additional equations for the molar concentration of the live polymer chains and reaction kinetics:

$$P_0 = \sqrt{\frac{2fIk_I}{k_{td} + k_{tc}}} \quad (7)$$

$$k_q = k_{0,q}e^{-E_q/RT}, \qquad q = p, fm, I, td, tc \quad (8)$$

where M, I, η_0, η_1, T_r and T_j are the monomer and initiator concentration, the zero-order and first-order moments of dead polymer, an the reactor and cooling jacket temperature, respectively. The input variables are the inlet volumetric flow rate F, the inlet monomer concentration M_{in}, and the volumetric flow rate in the cooling jacket F_j. All model parameters values are taken from (Daoutidis et al., 1990). The complete input variables vector is measured because each one can be used to close control loops. The output variables are either the reactor temperature or the monomer concentration given by the FTIR spectrometer. Here, the calibration model of FTIR is not shown due to space limitations. As mentioned before, the most important variables in polymerization processes are related to the chain distribution. In the MMA process, the variable showing the quality of the polymer is the average molecular weight $\overline{M}_w$ which is given by:

$$\overline{M}_w = \frac{\eta_0}{\eta_1} \quad (9)$$

4.3 Estimator design

As stated before, observability must be analyzed before any state estimation design. In this process, observability is analyzed in Table 1, showing the rank and condition number of the observability matrix $\mathcal{O}$ around the operating point. As it can be seen, $\mathcal{O}$ does not have complete rank by measuring either the reactor temperature or the monomer concentration. The system is only observable when the first order moment is measured. Since the first order moment is a fictitious state, there is no way to provide a real measurement and hence observability cannot be guaranteed. In the literature, the lack of observability in this system is attributed to the two non-observable states η_0 and η_1 (Shahrokhi and Fanaei, 2001; Ray, 1985).

If the system dynamics is reduced at only four state variables M, I, T_r and T_j, as is proposed by (Shahrokhi and Fanaei, 2001), observability is guaranteed with different combination of sensors as shown in Table 2. The practitioner might find more practical the selection of the temperature sensor for state estimation. Nonetheless, FTIR spectrometer helps out to provide a better estimates as suggested by the condition number of the observability matrix. This fact suggests that with the monomer concentration measurement, obtained from a FTIR spectrometer, observability is better conditioned with respect to the traditional temperature measurement.

Table 1. Observability condition of complete model

Measurements	Obs.	Rank	Cond. Numb
M	✗	4	6.31E+24
T_r	✗	4	1.58E+26
T_j	✗	4	3.49E+28
$[M\ T_r]$	✗	4	1.30E+25
$[\eta_0\ T_j]$	✗	5	1.28E+27
$[M\ \eta_0\ T_j]$	✗	5	6.78E+24
$[\eta_0\ \eta_1\ T_j]$	✓	6	5.26E+10

Table 2. Observability condition of reduced model

Measurements	Obs.	Rank	Cond. Numb
M	✓	4	5.64E+4
T_r	✓	4	3.27E+8
T_j	✓	4	5.75E+10
$[M\ T_r]$	✓	4	6.02E+4
$[M\ T_j]$	✓	4	3.01E+6

The next step is to design two EKF to contrast the performance using different measurements, the reactor temperature, as usually, and the monomer concentration provided by a FTIR spectrometer. Both filters are tuned equally with same initial conditions. The *a posteriori* estimation error covariance matrix is initialized as:

$$P_0^+ = 10^2 \cdot I_6 \quad (10)$$

where I_6 stands for the identity matrix with same order of the system. The initial value of the *a posteriori* estimated state is set as:

$$\hat{x}_0^+ = [7,7608\ 0, 10001\ 0, 00032\ 20, 925\ 328\ 297]^T \quad (11)$$

where variables are presented with proper units. The tuning matrices Q and R are also the same for both EKF filters and are fixed as:

$$Q = \begin{bmatrix} 10^{-8} & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^{-12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^{-16} & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^{-12} & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^{-9} & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^{-9} \end{bmatrix} \quad (12)$$

$$R = 10^{-6} \quad (13)$$

These matrices were found by trial-error methods, but taking into account some considerations around variables, units and squared measurement uncertainties. In that sense, $R = R_u R_u^T$ where R_u is the matrix that contains measurements uncertainties. The values of Q related to η_0 and η_1 are the lowest, due to this variables are based on moments approximation for the polymer chain.

4.4 Results and discussion

The nonlinear model of the process, (1) to (8), was simulated in MATLAB® using the fourth order Runge-Kutta numeric method with a time-step of $1\ s$. The simulation time was $10\ h$. The measurement noises followed a normal distribution with zero-mean and standard deviations of $0.01\ ^\circ C$ for reactor temperature measurement and $0.001\ kmol/m^3$ for monomer concentration measurement. Plant simulation was initialized in the steady state point of the system:

$$x_0 = [7,7908\ 0, 1006\ 0, 00031\ 20, 9425\ 329, 59\ 296, 65]^T \quad (14)$$

$$u = [1\ 8\ 3.26]^T \quad (15)$$

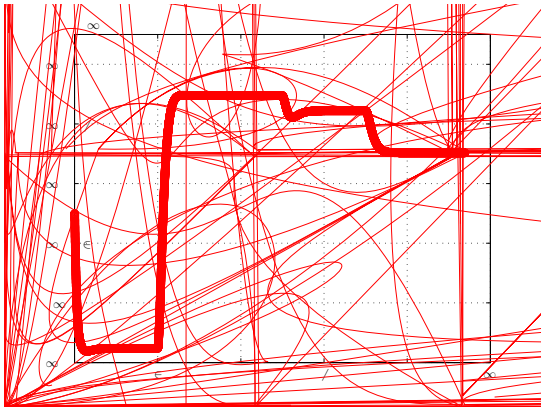


Fig. 3. State estimation without measurement noise and different initial state

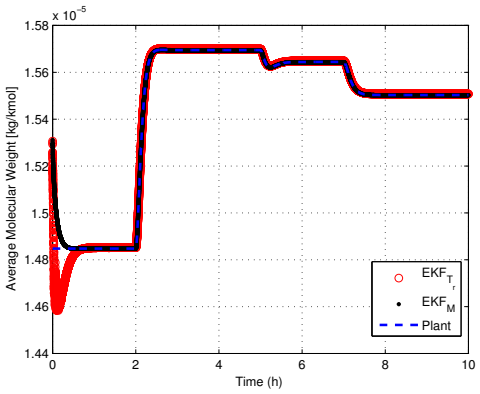


Fig. 4. State estimation with measurement noise and unknown initial state

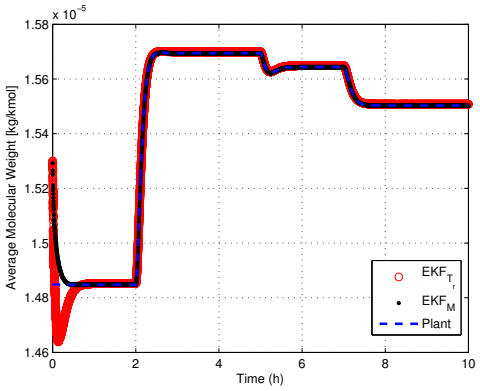


Fig. 5. State estimation for different sampling time of filters

Model uncertainties were not considered. At 2 hours, the inlet volumetric flow F changed from 1 m^3/h to 1.1 m^3/h. After 5 hours of simulation, monomer inlet concentration M_{in} was changed from 8 $kmol/m^3$ to 8.2 $kmol/m^3$. Finally, at 7 hours, the cooling jacket volumetric flow was changed from 3.2636 m^3/h to 3.5 m^3/h. Also, the sampling time of both EKF is set in 1 s, which same of time-step solution of the plant.

When the estimator is initialized in the same as plant values, it shows the fitting between both estimates and the real values is appropriate, which is reasonable because all inputs are known. The changes in input variables are followed without any mismatch.

Effects from changes in the initial state When different initial conditions are set to the plant and filters, both filters showed convergence to the real values. Normally, the real value of the initial state is unknown, so the filter is initialized at an approximated guess. Figure 3 presents the estimated state when initial values are different from plant (nominal). The simulation shows that the difference is noticed by the filters. The EKF_{T_r} has a greater overshoot than the EKF_M. In fact, this minimal difference is evidenced in Table 3, where some performance indexes are shown for the different scenarios. The changes on input variables were also followed.

Effect of measurement noise. Measurement noise is added to the reactor temperature for EKF_{T_r} and to the monomer concentration for the EKF_M filter. In this case, a zero-mean white noise with R as in (13) is considered. Although measurement noise has appreciable effects over all state variables, only the average molecular weight is of interest in this process.

The effect of noise is evidenced in Figure 4. The EKF_{T_r} converges slower than EKF_M. An appreciable difference in the time-error index, suggesting a possible steady state error is shown in Table 3. Again EKF_M is prominently better in first hours of simulation, despite that it is stated elsewhere that estimation with monomer concentration in first hours can lead a wrong estimated of average molecular weight (Othman et al., 2004).

Effect of sampling time. In real applications, monomer measurements take more time than temperature measurements. Despite the use of FTIR for online measurement of monomer concentration, this data present a larger sampling time, compared with availability of temperature data. Literature has encountered that for FTIR spectrometry, depending on calibration model, availability of the data is around 6-10 seconds, that mean, that there is a delay time between radiation measurement and final concentration value retrieving. In this sense, a change in sampling time of EKF_M was performed in order to analyze its effects on the estimation quality of the average molecular weight. The sampling time for EKF_M was switched from 1 s to 10 s, while sampling time of EKF_{T_r} was kept in 1 s, with the aim of run more realistic simulations. Figure 5 shows the better performance for EKF_M in spite of a larger sampling time. This result is very important for industrial implementations. In the Figure 6 a zoom on the first 0.45 h can be seen in order to appreciate the transient behavior of the filters.

In Table 3, the error indexes are presented for all studied cases, i.e., with same initial state, with measurement noise, and with different sampling time. MSE is the mean square error and ITAE stands for integral of time absolute error.

5. FINAL COMMENTS

Advantages of using spectroscopy for online monitoring were discussed with emphasis on FTIR spectroscopy. When complex nonlinear processes are dealt with, a reliable process model together with available measurements give an appealing alternative for process monitoring. The observability analysis showed that even if the observability property is not fulfilled, some states can be estimated if the detectability property is guaranteed. Additionally, FTIR measurements can improve the performance of the EKF estimation scheme. It is important to

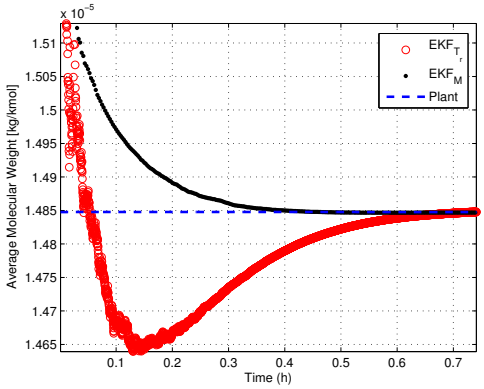


Fig. 6. State estimation for different sampling time of filters (detailed)

Error Index	$\mathbf{EKF_{T_r}}$	$\mathbf{EKF_M}$
Without noise and $\hat{x}_0 = x_0$		
MSE	1.985×10^{-20}	1.985×10^{-20}
ITAE	9.252×10^{-10}	9.253×10^{-10}
Effect of initial estimated state		
MSE	1.048×10^{-15}	8.515×10^{-16}
ITAE	5.974×10^{-9}	5.459×10^{-9}
Effect of measurement noise		
MSE	3.292×10^{-15}	1.266×10^{-15}
ITAE	2.410×10^{-7}	3.850×10^{-8}
Effect of sampling time		
MSE	7.611×10^{-16}	7.614×10^{-16}
ITAE	2.332×10^{-7}	3.531×10^{-8}

Table 3. Error index for all scenarios

take into account realistic aspects as the sampling time of the composition measurements, because EKF_M preserves the best performance compared to conventional EKF_{T_r}, even when information is available every 10 s. In that way, current process monitoring in polymerization system can be improved, combining both FTIR spectroscopy and nonlinear state estimation techniques.

Finally, it is important to emphasize that many estimation schemes and especially the EKF needs a statistical identification step to find out the real noise covariances by using appropriate methods like the autocovariance least square (ALS). Additional analysis to be performed consist to incorporate the temperature and monomer concentration measurements together to achieve a more robust estimator, and that probably may deal with time-delay and asynchronous information. Finally, more reliable nonlinear filters like the Unscented Kalman Filter (UKF), the Nonlinear Moving Horizon Estimator (NMHE), and the Particle Filters need to be tested for comparison purposes.

REFERENCES

Apostolos, K., Dimitris, M., Vassilis, S., Christos, C., and Costas, K. (2006). *Dynamic Optimization of Molecular Weight Distribution in Batch Polymerization Reactors*, volume 39. IFAC.

Daoutidis, P., Soroush, M., and Kravaris, C. (1990). Feedforward/feedback control of multivariable nonlinear processes. *AIChE Journal*, 36(10), 1471–1484.

Duan, G., Zhang, C., Li, A., Yang, X., Lu, L., and Wang, X. (2008). Preparation and characterization of mesoporous zirconia made by using a poly (methyl methacrylate) template. *Nanoscale Research Letters*, 3(3), 118–122.

Eaton, M.T. and Ricker, N.L. (1995). Extended Kalman Filtering for Particle Size Control in a Fed-batch Emulsion Polymerization Reactor. In *American Control Conference (ACC), 1995*, June, 2697–2701. Seattle, Washington.

Feng, E., Domlan, E., and Kadali, R. (2015). Spectroscopic measurements in oil sands industry-from laboratories to real-time applications. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 48(8), 199–204.

Gallignani, M., Rondón, R.a., Ovalles, J.F., and Brunetto, M.R. (2014). Transmission FTIR derivative spectroscopy for estimation of furosemide in raw material and tablet dosage form. *Acta Pharmaceutica Sinica B*, 4(5), 376–383.

Graichen, K., Hagenmeyer, V., and Zeitz, M. (2005). Adaptive Feedforward Control with Parameter Estimation for the Chylla – Haase Polymerization Reactor. 3049–3054.

Kiparissides, C., Seferlis, P., Mourikas, G., and Morris, A.J. (2002). Online Optimizing Control of Molecular Weight Properties in Batch Free-Radical Polymerization Reactors. *Industrial & Engineering Chemistry Research*, 41(24), 6120–6131.

Ling, C. and Kravaris, C. (2016). State Observer Design for Monitoring the Degree of Polymerization in a Series of Melt Polycondensation Reactors. *Processes*, 4(1), 4.

Monsalve-Bravo, G.M., Moscoso-Vasquez, H.M., and Alvarez, H. (2014). Scale-up of continuous reactors: Using phenomenological-based models. *Chimica Oggi/Chemistry Today*, 32(3), 20–26.

Othman, N.S., Fevotte, G., Peycelon, D., Egraz, J., and Suau, J. (2004). Control of polymer molecular weight using near infrared spectroscopy. *AIChE Journal*, 50(3), 654–664.

Ray, W.H. (1985). Polymerization Reactor Control. *1985 American Control Conference*, 3–8.

Salau, N.P.G., Trierweiler, J.O., and Secchi, A.R. (2014). State Estimation of Chemical Engineering Systems Tending To Multiple Solutions. *Brazilian Journal of Chemical Engineering*, 31(03), 771–785.

Santos, A.F., Silva, F.M., Lenzi, M.K., and Pinto*, J.C. (2005). Monitoring and Control of Polymerization Reactors Using NIR Spectroscopy. *Polymer-Plastics Technology and Engineering*, 44(1), 1–61.

Shahrokhi, M. and Fanaei, M.A. (2001). State Estimation in a Batch Suspension Polymerization Reactor. *Iranian Polymer Journal*, 10(3), 1026–1065.

Shahrokhi, M. and Fanaei, M.A. (2002). Molecular Weight Control of a Batch Polymerization Reactor. *Iranian Polymer Journal*, 11(6), 403–411.

Shenoy, A.V., Prasad, V., and Shah, S.L. (2010). Comparison of unconstrained nonlinear state estimation techniques on a MMA polymer reactor. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 9(PART 1), 159–164.

Null-space based control for human escorting by using Mobile Robots

Daniel Herrera, Javier Gimenez, Marcos Toibero, Ricardo Carelli

Instituto de Automática-CONICET, Universidad Nacional de San Juan, Av.Libertador Oeste 1109, San Juan-Argentina (e-mail: {dherrera, jgimenez, mtoibero, rcarelli}@inaut.unsj.edu.ar).

Abstract: Sharing the workspace or interacting directly with people in a social way is currently a key challenge in the design of mobile robot applications. An interesting field of study is related with the so-called hidden dimension, which relates the sense of security that a human feels when interacting with another one. Therefore, the researchers have been interested in developing human-friendly applications by abstracting social cues for the robotic design. One of these is the social zone, which acts like a repulsive potential field, and if it is not respected, an uncomfortable situation for humans is produced. In robotics, many approaches have been proposed to define and to avoid these fields, which contributes to improve the human comfort during interactions at least from a qualitative perspective. This paper proposes a novel null-space-based (NSB) algorithm for a non-holonomic mobile robot platform, which is programmed to escort a human in a behavior-based paradigm. The emphasis is posed in the evasion of other humans in the environment by considering them as elliptical potential fields with non-holonomic motion. Finally, simulation results are presented to show the performance of the proposed control algorithm.

Keywords: Robot navigation, Human-centered design, Human factors, Non-linear control.

1. INTRODUCTION

According to the proxemic studies developed by Hall (1963), the human respects social zones during different kind of interactions, i.e. there are distances to describe intimate and social spaces depending of the task, the situation, and even of cultural or personal preferences.

When a robot navigates in a human-shared environment, it is supposed that it must also respect these social zones to improve its social acceptance. As consequence of this hypothesis, in robotics, some conventions have been established. For example, Chi-Pang et al. (2011) discuss different types of personal space for humans according to the situation, e.g., they assume an egg-shaped personal space for the human while it is moving, due to they should have a long and clear space to walk (giving the sense of safety). For this, they consider that the length of the semi-major axis of the potential field is proportional to the human velocity. Scandolo and Fraichard (2011) use personal space in their social cost map model for path simulation. Guzzi et al. (2013) incorporates a potential field that dynamically modifies its dimensions according to the relative distance with the human to avoid an occlusion event or "deadlocks". In Ratsamee et al. (2013), a human-friendly navigation is proposed, where the concept of personal space or "hidden space" is used to prevent uncomfortable feelings when humans avoid or interact with robots. This is based on the analysis of human motion and behavior (face orientation and overlapping of personal space).

In reference to the human following, in Kluge et al. (2002) it is given more relevance to the coordination between human and a mobile robot platform. It is based on the dynamic obstacle avoidance denominated "velocity obstacles" (Fiorini and Shiller, 1998). This allows the robot changing its orientation and velocity according to the locomotion of the followed person in a dynamic environment. In Loper et al. (2008) a robotic system for human following is presented, which is capable to answer verbal and non-verbal instructions under non-structured conditions. For detecting and following a human it is used a Kalman filter and a PID control respectively. Doisy et al. (2012) propose algorithms for human following by using depth images of Kinect® sensor. The first constitutes a following algorithm of the human path, and the second one is an adaptive algorithm that additionally uses a range sensor to identify and generate dynamically a path for the robot in a previously mapped scenario. Guansheng et al. (2013), developed a vision based algorithm by using a mobile robot endowed with a Kinect® sensor to follow humans, where the skeleton tracking function allows to get the human position. Consequently the control is used to maintain this skeleton in the center of the image and in a fixed distance from the robot. Similarly, in Machida et al. (2012) it is proposed a control algorithm for mobile robots based in the 3D Kinect® information to generate linear and angular velocity commands.

Regarding the behavior-based control, the problem is divided into sub-problems to be individually controlled. Generally, there are overlapping interests between these

tasks, and it should be chosen a behavior that balances the sub-objectives. The performance of the control system depends on how these controllers are fused. In Antonelli et al. (2005) an experimental comparison among three approaches are analyzed: the layered control system, the motor scheme control and null-space behavioral control (Chiaverini, 1997). The main difference between them is the way they manage the sub-task outputs. Specifically, the null-space based approach derives from an inverse kinematics solution by exploiting the kinematic redundancy, in which there are more degrees of freedom than necessary to fulfil the task. From a practical point of view, the kinematic redundancy is not strictly present in the robotic system, but the task can give redundancy to the system. In robotic manipulators, the simplest method to define the null-space is based on the inverse Jacobian (Whitney, 1969). This theory for redundant robotic manipulators has been proposed for the execution of different formation-control missions with multi-mobile-robots systems (Antonelli et al., 2009), whose collective task make the system redundant.

This paper proposes to use the null-space based (NSB) approach to establish an cooperative leader-follower control that considers multiple kinematic tasks during the interaction with humans. The first one consists in keeping out of social zones of humans, which are defined by considering an elliptical potential fields moving with a non-holonomic nature. Additionally, two secondary tasks are proposed in the NSB paradigm, which are related to keep the orientation and distance of the formation. The designed control is simulated to test its performance. It has been made emphasis in the evasion of humans with the above mentioned social zones.

In this way, in Section 2 an elliptical potential field is proposed, and, the Jacobian, that relates the potential variation with the motion of the robot is established. Later, in Section 3 the avoidance of this field is defined through the minimum norm solution, i.e. by using the pseudo-inverse of the Jacobian matrix, where additional secondary tasks to keep the formation with the human leader are included as part of a null-space based design. Finally, the performance of the algorithm is tested through simulation in Section 4. Also, the paper conclusions are presented in Section 5.

2. SOCIAL POTENTIAL FIELD

When a human walks between other humans, there are social zones which are interpreted as repulsive potential fields. In general, they are distance dependent.

In this way, given a global reference system $x-y$ and the human posture $\{x_h, y_h, \theta_h\}$ at time t, it is considered the reference system x^*-y^* rotated by θ_h. (see Fig. 1). By using these coordinates, let's define the following potential field,

$$V_h(t) = \exp\left\{-\left(\frac{x^*-x_h^*}{a}\right)^2 - \left(\frac{y^*-y_h^*}{b}\right)^2\right\},$$

where (x^*, y^*) is a point of the potential field in the rotated framework x^*-y^*; and (x_h^*, y_h^*) the human position and the center of this field, a is the major-axis length of the elliptical Gaussian form, and b is the minor-axis length (see Fig. 1).

Let's define a shape-variant potential field by considering the major-axis as time dependent. It is based on a social heuristic that considers that a human needs a long and clear space to move while walking (Chi-Pang et al., 2011).

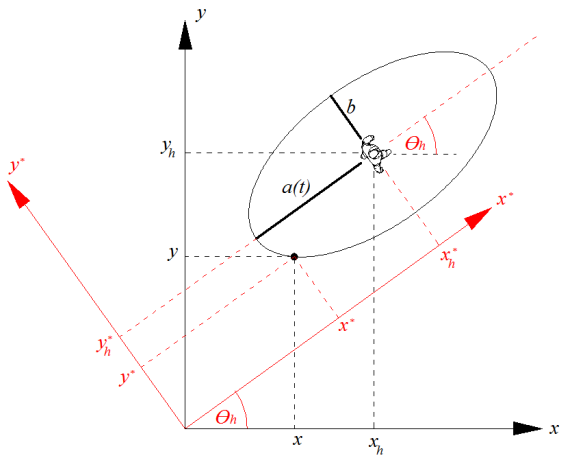


Fig. 1. Schematic description of the social zone.

In consequence, the time-derivative of this field is

$$\frac{dV_h}{dt} = \left[\frac{\partial V_h}{\partial x^*}, \frac{\partial V_h}{\partial y^*}, \frac{\partial V}{\partial x_h^*}, \frac{\partial V_h}{\partial y_h^*}, \frac{\partial V_h}{\partial a}\right] \begin{bmatrix} \dot{x}^* \\ \dot{y}^* \\ \dot{x}_h^* \\ \dot{y}_h^* \\ \dot{a} \end{bmatrix}.$$

Solving this expression through algebraic steps, it results that

$$\begin{aligned}\frac{dV_h}{dt} =& -2V_h\left[(x^*-x_h^*)/a^2, (y^*-y_h^*)/b^2\right]\dot{\mathbf{x}}^* + \cdots \\ &+ 2V_h\left[(x^*-x_h^*)/a^2, (y^*-y_h^*)/b^2\right]\dot{\mathbf{x}}_h^* + \cdots \\ &+ 2V_h\frac{(x^*-x_h^*)^2}{a^3}\dot{a}, \end{aligned} \quad (1)$$

where $\dot{\mathbf{x}}^* = [\dot{x}^*, \dot{y}^*]^T$, $\dot{\mathbf{x}}_h^* = [\dot{x}_h^*, \dot{y}_h^*]^T$. It can be compactly expressed like

$$\dot{V}_h = \mathbf{J_s}\dot{\mathbf{x}}^* - \mathbf{J_s}\dot{\mathbf{x}}_h^* + J_f\dot{a}. \quad (2)$$

By considering the rotation matrix,

$$\mathbf{R} = \begin{bmatrix} \cos\theta_h & \sin\theta_h \\ -\sin\theta_h & \cos\theta_h \end{bmatrix},$$

it is possible to express it in the $x-y$ global framework, i.e.,

$$\begin{bmatrix} x^* \\ y^* \end{bmatrix} = \mathbf{R}\begin{bmatrix} x \\ y \end{bmatrix}, \quad \begin{bmatrix} x_h^* \\ y_h^* \end{bmatrix} = \mathbf{R}\begin{bmatrix} x_h \\ y_h \end{bmatrix}.$$

By considering the time-derivative of this expressions, it is obtained

$$\begin{aligned}\dot{\mathbf{x}}^* &= \begin{bmatrix} \cos\theta_h & \sin\theta_h & -\sin\theta_h x + \cos\theta_h y \\ -\sin\theta_h & \cos\theta_h & -\cos\theta_h x - \sin\theta_h y \end{bmatrix}\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta}_h \end{bmatrix}, \\ &= \mathbf{R}\dot{\mathbf{x}} + \mathbf{p}\dot{\theta}_h, \end{aligned} \quad (3)$$

where $\mathbf{p} := \begin{bmatrix} -\sin\theta_h x + \cos\theta_h y \\ -\cos\theta_h x - \sin\theta_h y \end{bmatrix}$. Similarly for the human position,

$$\dot{\mathbf{x}}_h^* = \begin{bmatrix} \cos\theta_h & \sin\theta_h & -\sin\theta_h x_h + \cos\theta_h y_h \\ -\sin\theta_h & \cos\theta_h & -\cos\theta_h x_h - \sin\theta_h y_h \end{bmatrix} \begin{bmatrix} \dot{x}_h \\ \dot{y}_h \\ \dot{\theta}_h \end{bmatrix},$$

$$= \mathbf{J}_h^* \begin{bmatrix} \dot{x}_h \\ \dot{y}_h \\ \dot{\theta}_h \end{bmatrix}.$$

If additionally it is considered a non-holonomic motion for the human gait (Arechavaleta et al., 2006; Leica et al., 2014), i.e,

$$\begin{bmatrix} \dot{x}_h \\ \dot{y}_h \\ \dot{\theta}_h \end{bmatrix} = \begin{bmatrix} \cos\theta_h & 0 \\ \sin\theta_h & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \nu_h \\ \omega_h \end{bmatrix} = \mathbf{J}_h \begin{bmatrix} \nu_h \\ \omega_h \end{bmatrix},$$

then it results,

$$\dot{\mathbf{x}}_h^* = \mathbf{J}_h^* \mathbf{J}_h \begin{bmatrix} \nu_h \\ \omega_h \end{bmatrix}, \tag{4}$$

where $\mathbf{J}_h^* \mathbf{J}_h := \begin{bmatrix} 1 & -\sin\theta_h x_h + \cos\theta_h y_h \\ 0 & -\cos\theta_h x_h - \sin\theta_h y_h \end{bmatrix}$.

In this way, by substituting (3) and (4) in (2), results that

$$\dot{V}_h = \mathbf{J_s R}\dot{\mathbf{x}} + \mathbf{J_s p}\omega_h - \mathbf{J_s} \mathbf{J}_h^* \mathbf{J}_h \begin{bmatrix} \nu_h \\ \omega_h \end{bmatrix} + J_f \dot{a},$$

$$:= \mathbf{J}_o \dot{\mathbf{x}} + g.$$

Therefore, the total repulsive effect over each agent i in a position $\mathbf{x}_i := (x_i, y_i)$ is calculated as the sum of all the repulsive effects V_{hj} generated by n human obstacles in the shared scenario, i.e.

$$V_i = \sum_{j=1}^{n} V_{hj},$$

and, in consequence

$$\dot{V}_i = \sum_{j=1}^{n} \dot{V}_{hj} = \sum_{j=1}^{n} \mathbf{J}_{oj} \dot{\mathbf{x}}_i + \sum_{j=1}^{n} g_j. \tag{5}$$

3. NULL-SPACE BASED CONTROL

Let $\mathbf{q} \in \mathbb{R}^m$ be the task variables that represent the states of the system to be controlled. Also, let $\mathbf{x} := [x_1, x_2, \cdots, x_n, y_n]^T \in \mathbb{R}^{2n}$ be the array with the positions (x_n, y_n) of the n members of the formation. Then, the relation between $\mathbf{q}$ and $\mathbf{x}$ is expressed like

$$\mathbf{q} = f(\mathbf{x}).$$

By considering its temporal derivative

$$\dot{\mathbf{q}} = \frac{df(\mathbf{x})}{d\mathbf{x}} \dot{\mathbf{x}} = \mathbf{J}(\mathbf{x}) \dot{\mathbf{x}}, \tag{6}$$

where $\mathbf{J} := \mathbf{J}(\mathbf{x}) \in \mathbb{R}^{m \times 2n}$ is the Jacobian matrix associated to the task. In this way, the redundancy of this system is given by the condition $m < 2n$, where $2n$ is interpreted as the degrees of freedom of the system. In this way, the problem is focused on defining control actions for each member to carry out the formation control objectives. A solution to this problem is typically given by considering the minimum norm solution, i.e., by using the pseudo-inverse of the Jacobian matrix $\mathbf{J}^\dagger := \mathbf{J}^\dagger(\mathbf{x}) = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}$ as follows

$$\dot{\mathbf{x}}_d = \mathbf{J}^\dagger \dot{\mathbf{q}}. \tag{7}$$

The desired velocities $\dot{\mathbf{x}}_d$ could be applied directly to certain kind of robots, however it is usual to define the i-tasks through position errors, therefore it is used the so-called closed-loop inverse-kinematics (CLIK) version of the algorithm for each task, namely

$$\dot{\mathbf{x}}_{di} = \mathbf{J}_i^\dagger(\mathbf{x}) \left(\dot{\mathbf{q}}_{di} + \mathbf{K}_i \tilde{\mathbf{q}}_i \right). \tag{8}$$

where $\mathbf{K}_i$ is a positive defined design matrix and, $\tilde{\mathbf{q}}_i$ is the task error defined as $\tilde{\mathbf{q}}_i := \mathbf{q}_{di} - \mathbf{q}$. i is the task number which are organized by priority. In this way, according to (Baillieul et al., 1984; Chiaverini, 1997), a non-minimun norm solution of CLIK for three tasks can be written as

$$\dot{\mathbf{x}}_d = \dot{\mathbf{x}}_{d1} + \left(\mathbf{I} - \mathbf{J}_1^\dagger \mathbf{J}_1 \right) \left[\dot{\mathbf{x}}_{d2} + \left(\mathbf{I} - \mathbf{J}_2^\dagger \mathbf{J}_2 \right) \dot{\mathbf{x}}_{d3} \right]. \tag{9}$$

Each task velocity is computed individually and the overall NSB control action always fulfills the highest priority task at non-singular configurations. However the fulfillment of the lower priority tasks should be discussed in a case-by-case basis.

3.1 Stability analysis

By multiplying each member of (9) by $\mathbf{J}_1$ under the assumption that it is a full-range matrix, and noting that $\mathbf{J}_1(\mathbf{I} - \mathbf{J}_1^\dagger \mathbf{J}_1) = 0$, then

$$\dot{\mathbf{q}}_1 = \dot{\mathbf{q}}_{d1} + \mathbf{k}\tilde{\mathbf{q}}_1. \tag{10}$$

This expression can be expressed as $\dot{\tilde{\mathbf{q}}} = -\mathbf{k}\tilde{\mathbf{q}}_1$. If it is also defined a Lyapunov candidate function with the form

$$\mathcal{V} = \frac{1}{2} \tilde{\mathbf{q}}_1^T \tilde{\mathbf{q}}_1, \tag{11}$$

then the time derivative of this function results

$$\dot{\mathcal{V}} = -\tilde{\mathbf{q}}_1 \mathbf{k} \tilde{\mathbf{q}}_1,$$

where $\mathbf{k}$ as a positive defined matrix makes the system asymptotically stable, i.e. $\tilde{\mathbf{q}}_1 \to 0$ with $t \to \infty$.

To establish the stability for secondary tasks, it is supposed that the secondary task does not affect the primary task and the two tasks are in consequence compatible.

It is expressed by the linearly independence of the subspaces $\mathcal{N}(\mathbf{J}_1)$ and $\mathcal{N}(\mathbf{J}_2)$, namely

$$\mathcal{N}^\perp(\mathbf{J}_1) \cap \mathcal{N}^\perp(\mathbf{J}_2) = \{0\},$$

when

$$\mathcal{N}^\perp(\mathbf{J}_1) \equiv \mathcal{N}(\mathbf{J}_2)$$

i.e, the subspaces $\mathcal{N}^\perp(\mathbf{J}_1)$ and $\mathcal{N}^\perp(\mathbf{J}_2)$ are orthogonal. It can be expressed as

$$\mathcal{R}(\mathbf{J}_2^\dagger) \subseteq \mathcal{N}(\mathbf{J}_1), \tag{12}$$

or well,

$$\mathbf{J}_2 \mathbf{J}_1^\dagger = 0.$$

In this way, if (9) is pre-multiplied by $\mathbf{J}_2$ then it results that

$$\dot{\mathbf{q}}_2 = \dot{\mathbf{q}}_{d2} + \mathbf{k}\tilde{\mathbf{q}}_2,$$

which analogously to (10) by considering the Lyapunov candidate function (11) results to be asymptotically stable.

Similarly for the third task,

$$\mathcal{R}(\mathbf{J}_3^\dagger) \subseteq \mathcal{N}(\mathbf{J}_1) \cap \mathcal{N}(\mathbf{J}_2). \tag{13}$$

In this way, following a similar procedure for the second task, the third task is demonstrated to be also asymptotically stable, provided that the third task does not affect the second and the first tasks.

3.2 Problem statement

Let us consider a formation of two agents of which one is a human. For this, it must be guaranteed two facts: the first one is related to the mutual coordination between formation members, and, the second one related to the presence of another humans in the environment.

To guarantee the stability of the first problem, it must be designed a control in which the human acts like a leader, i.e., he is capable to move freely, while the formation is moving accordingly to maintain a pre-established formation posture or shape. In this way, the formation is named as human-centered because it is not possible to govern the human actions.

On the other hand, for the second fact, the presence of humans and its avoidance must be established as primary goal. These two design qualities are established in the following Sections.

First, let's $\mathbf{x} := [x_1, y_1, x_2, y_2]^T$ be a vector with the Cartesian positions $(x_i, y_i), i = \{1, 2\}$ of both agents, where the first agent is a robot follower and the second agent a human leader. Later, the variables for each task are defined as follows.

3.3 Principal task: Social zone avoidance

The meddling avoidance of the robot into human social zones is defined as the principal task. Then, let's V_i be the task variable that represents the repulsive effect over each individual because of the presence of other individuals. For this, consider (5) to be expressed for both agents (human and robot) like

$$\begin{aligned}\begin{bmatrix}\dot{V}_1\\ \dot{V}_2\end{bmatrix} &= \begin{bmatrix}\mathbf{J}^{(1)} & \mathbf{0}_{1\times 2}\\ \mathbf{0}_{1\times 2} & \mathbf{J}^{(2)}\end{bmatrix}\dot{\mathbf{x}} + \begin{bmatrix}g_1\\ g_2\end{bmatrix},\\ &:= \mathbf{J}_1\dot{\mathbf{x}} + \mathbf{g}, \end{aligned} \tag{14}$$

where $\mathbf{J}^{(i)} := \sum_{j=1}^{n}\mathbf{J}_{oj}, g_i := \sum_{j=1}^{n} g_j$ are the Jacobians and compensation motion factor for each individual $i = \{1, 2\}$ under the presence of n human obstacles.

In this way, the minimal norm solution for the formation in this task is expressed as,

$$\dot{\mathbf{x}}_{d1} = \mathbf{J}_1^{\dagger}\left(\dot{\mathbf{V}}_d + \mathbf{K}_1\tilde{\mathbf{V}} - \mathbf{g}\right), \tag{15}$$

where $\mathbf{V}_d := [V_{d1}, V_{d2}]^T$ and $\tilde{\mathbf{V}}_d := [V_{d1} - V_1, V_{d2} - V_2]^T$.

Note that in the expression (14) the resultant Jacobian $\mathbf{J}_1$ brings out an uncoupled condition for the agents, i.e., each individual is in charge of avoiding its obstacles. In consequence, the robot agent is controlled to avoid its human obstacles, and, as a trivial fact, the human agent avoids its obstacles by himself without affecting the system stability.

The incorporation of $\mathbf{g}$ improves the performance of the evasion of humans compared to other approaches for common dynamic obstacles. This is because it does not only take into account the linear motion of the obstacle, but also it is related to the angular motion of the human obstacle, and, its dynamic shape.

3.4 Secondary task: Formation posture

Let's $\mathbf{q}_2 := [\gamma, x_f, y_f]^T$ be the formation posture variables, where

$$\gamma = \operatorname{atan}\frac{y_1 - y_2}{x_1 - x_2}, \quad x_f = x_2, \quad y_f = y_2, \tag{16}$$

with γ the orientation of the formation and (x_f, y_f) the position of the formation. In consequence,

$$\mathbf{q}_2 = f(\mathbf{x}).$$

Note that the position of the formation is centered on the human leader. It is given to guarantee that the human is free to move in whatever direction he wants; therefore, the formation is capable to keep the formation posture by itself.

By considering its time derivative, it results

$$\dot{\mathbf{q}}_2 = \mathbf{J}_2\dot{\mathbf{x}} \tag{17}$$

where,

$$\mathbf{J}_2 := \begin{bmatrix}\frac{y_1 - y_2}{d^2} & -\frac{x_1 - x_2}{d^2} & -\frac{y_1 - y_2}{d^2} & \frac{x_1 - x_2}{d^2}\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}, \tag{18}$$

and, its pseudo-inverse is expressed like

$$\mathbf{J}_2^{\dagger} = \begin{bmatrix} y_1 - y_2 & \frac{(y_1 - y_2)^2}{d^2} & -\frac{(x_1 - x_2)(y_1 - y_2)}{d^2}\\ -(x_1 - x_2) & -\frac{(x_1 - x_2)(y_1 - y_2)}{d^2} & \frac{(x_1 - x_2)^2}{d^2}\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix},$$

where $d^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$.

The minimal norm solution for this task is expressed as,

$$\dot{\mathbf{x}}_{d2} = \mathbf{J}_2^{\dagger}\left(\dot{\mathbf{q}}_{2d} + \mathbf{K}_2\tilde{\mathbf{q}}_2\right), \tag{19}$$

where $\mathbf{q}_{2d} := [\gamma_d, x_{fd}(t), y_{fd}(t)]^T$, $\tilde{\mathbf{q}}_2 := \mathbf{q}_{2d} - \mathbf{q}_2$, and, $\mathbf{K}_2 := k_2\mathbf{I}_3$. In this way, if the desired trajectory for the formation is the human trajectory, i.e., $x_{fd}(t) = x_h, \quad y_{fd}(t) = y_h$, then it results in a trivial solution for the human leader and a control action for the robot, which are given by

$$\begin{aligned}\dot{\mathbf{x}}_{d2} &= \mathbf{J}_2^{\dagger}\left[\begin{pmatrix}0\\ \dot{x}_h\\ \dot{y}_h\end{pmatrix} + \mathbf{k}_2\begin{pmatrix}\gamma_d - \gamma\\ 0\\ 0\end{pmatrix}\right],\\ &= \begin{bmatrix}k_2(y_1 - y_2)\tilde{\gamma}\\ k_2(x_1 - x_2)\tilde{\gamma}\\ \dot{x}_h\\ \dot{y}_h\end{bmatrix}.\end{aligned}$$

3.5 Tertiary task: Formation shape

Finally, the tertiary task is related to the formation shape, i.e., the distance between the individuals. Let's consider $q_3 := d$ as the only formation shape variable, where

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}, \tag{20}$$

which is the distance between them. By considering its time derivative, it results

$$\dot{q}_3 = \mathbf{J}_3\dot{\mathbf{x}} \tag{21}$$

where,

$$\mathbf{J}_3 := \left[-\frac{x_1 - x_2}{d}, -\frac{y_1 - y_2}{d}, \frac{x_1 - x_2}{d}, \frac{y_1 - y_2}{d} \right], \quad (22)$$

and, its pseudo-inverse is expressed like

$$\mathbf{J}_3^{\dagger} := \left[-\frac{x_1 - x_2}{2d}, -\frac{y_1 - y_2}{2d}, \frac{x_1 - x_2}{2d}, \frac{y_1 - y_2}{2d} \right]^T, \quad (23)$$

The minimal norm solution for this task is expressed as,

$$\dot{\mathbf{x}}_{d3} = \mathbf{J}_3^{\dagger} \left(\dot{q}_{3d} + k_3 \tilde{q}_3 \right), \quad (24)$$

where $q_{3d} := d_d$, $\tilde{q}_3 := q_{3d} - q_3$ and k_3 is a positive design constant.

Note that this task does not affect the second one.

3.6 Mobile robot control

Once that each task of the formation has been defined in the Cartesian space and, it has been expressed the control actions for each agent as (9), it results

$$\dot{\mathbf{x}}_d = [\dot{x}_d^{(1)} \dot{y}_d^{(1)} \dot{x}_d^{(2)} \dot{y}_d^{(2)}]^T,$$

where $(\dot{x}_d^{(k)}, \dot{y}_d^{(k)}), k = \{1, 2\}$ are the $x-y$ velocities for each k agent. As mentioned the control actions for the human leader are trivial, i.e. $\dot{x}_2 = \dot{x}_d^{(2)}$ and $\dot{y}_2 = \dot{y}_d^{(2)}$. However the control actions generated for the robot must be expressed like robot commands of linear and angular velocity. To this aim, consider the kinematic model of a differential drive mobile robot as follows

$$\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \end{bmatrix} = \begin{bmatrix} \cos\theta_r & -r\sin\theta_r \\ \sin\theta_r & r\cos\theta_r \end{bmatrix} \begin{bmatrix} \nu_r \\ \omega_r \end{bmatrix} := \mathbf{J}_r \mathbf{u}_r \quad (25)$$

where θ_r is the robot orientation, r is the distance from the middle of the wheels to a point in a perpendicular direction, and, $\{x_r, y_r\}$ is the Cartesian robot position. Note that $\dot{\theta}_r$ is not expressed in this model because the robot orientation is not controlled when the objectives are focused on the formation, i.e., it is only controlled the robot relative position to get a formation orientation. In this way by considering (25), it can be expressed the control action for the mobile robot as

$$\mathbf{u}_r := \mathbf{J}_r^{-1} \begin{bmatrix} \dot{x}_d^{(1)} \\ \dot{y}_d^{(1)} \end{bmatrix}. \quad (26)$$

3.7 Discussion

Even when the proposed solution results in a flexible formation by considering as priority the posture of the formation over its shape, it can be also selected other possible variable combination to give rigid characteristics with similar results.

Based on a leader-follower formation it must be always guaranteed that the formation trajectory is given by the leader trajectory. In this way, to get a rigid formation, the variables should be chosen as follows. For the second task, $\mathbf{q}_2 := [d, x_f, y_f]^T$ and for the third task $q_3 := \gamma$.

By following a similar procedure, the task priorities could be redefined with equal valid results, but by giving different priorities to the formation orientation γ and the formation shape d.

4. SIMULATION RESULTS

In order to test the performance of the proposed control, it is considered a simulated Pioneer 3AT robot which is programmed to follow a human. The simulations are focused on showing the performance under the presence of both static and dynamic obstacles, where the human obstacles have been defined as elliptical potential fields with non-holonomic motion as previously detailed.

Additionally, the effect of the compensation factor g, as shown in (15), has been tested by considering the two cases, the first one where the motion of the obstacle is not considered, i.e., $g = 0$ and, the second one when the motion of the human obstacle is characterized by $g \neq 0$, which relates the variation of the potential field with the motion of the robot during the meddling event. This effect is only present during the evasion of dynamic obstacles and it is related with a compensation of the human motion that allows to avoid him in a more suitable way.

In this way, the parameters were chosen as

Table 1. Parameters Simulation

Description	**Parameters**	
Leader trajectory	$x_d = 20\arctan(0.03t)$	$y_d = 0.5$
Social zone	$a = b + 1.2\nu_h$	$b = 0.5$
Formation	$\gamma_d = \pi/4[rad]$, $d_d = 2.5[m]$	
	$k_1 = 0.0001$, $k_2 = 1.2$, $k_3 = 2$	

4.1 Simulation 1: Lateral meddling

For the first simulation let's consider a dynamic human obstacle, which has both: rotation and translation motions. These effects are included as part of the g compensation by considering the human obstacle velocities ν_h and ω_h. For this case, it is considered a constant linear velocity of the human obstacle, i.e., the major-axis of the potential field is fixed while the human is moving in a sinusoidal trajectory. Additionally, it is considered a static human obstacle. The time-parametrized trajectories are presented in Fig. 2.

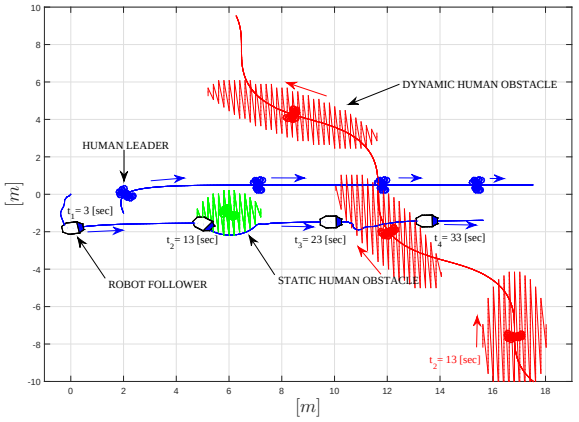
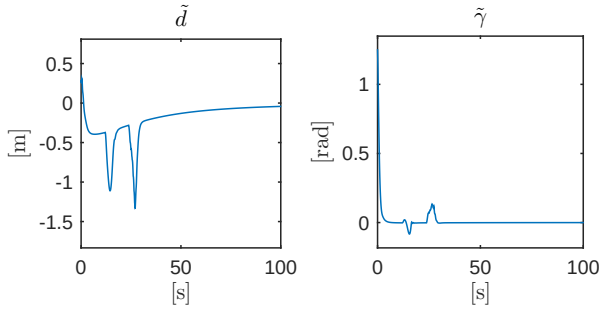


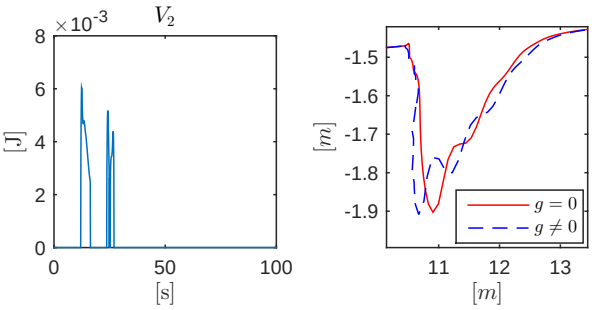
Fig. 2. Top view trajectories.

Also, the motion errors for each sub-task are presented in Fig. 3, where it is also presented an augmented view of the trajectories developed by the robot during the

evasion of the dynamic obstacle. Note that by including the motion compensation term g it is reduced the meddling in the social zone during the evasion as consequence of considering the variation of the field with the motion, which in this case generates a fastest deviation to avoid the human social zone. This effect can be also seen in a lower potential repulsion values over the robot during the simulation time (see Fig. 3c).



(a) Distance error. (b) Orientation error.



(c) Repulsive potential field over robot. (d) Dynamic object "g" effect.

Fig. 3. Formation motion errors during lateral meddling.

4.2 Simulation 2: Front meddling

For this simulation it is proposed to consider a frontal meddling between the robot and a human obstacle with time variant velocity. In this case, it is emphasized the improving of the performance, regarding the avoidance of the meddling in the elliptical social field when its major axis is not fixed but defined as $a := a(t) = b + k_a \nu_h$, where b is a constant that defines the minor axis of the potential field, k_a is a positive constant design, and, ν_h is the linear velocity of the human obstacle.

The time-parametrized trajectories are presented in Fig. 4.

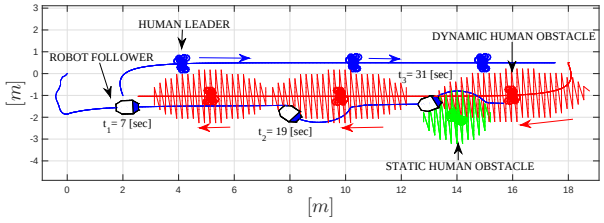
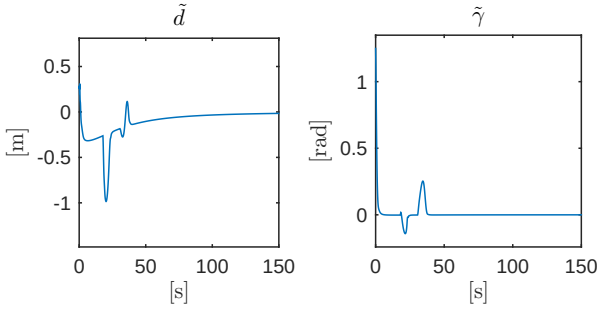
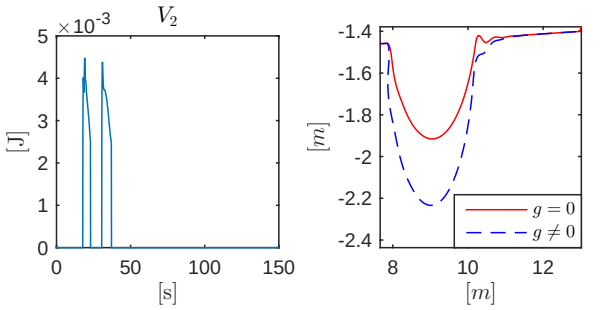


Fig. 4. Top view trajectories.

Additionally, the motion errors for each sub-task are presented in Fig. 5, where it is also presented an augmented view of the trajectories developed by the robot during the evasion of the dynamic obstacle. Note that by including the motion compensation term g, it is reduced the meddling in the social zone during the evasion by getting a longer but an early evasion.



(a) Distance error. (b) Orientation error.



(c) Repulsive potential field over robot. (d) Dynamic object "g" effect.

Fig. 5. Formation motion errors during frontal meddling.

5. CONCLUSION

This paper has presented a novel null-space based (NSB) motion control to follow a human in an unstructured scenario by considering a differential drive mobile robot. For this, it has been proposed the definition of a "social" potential field, which according the researchers, improves the social acceptance of the mobile robots during human-robot interactions. Additionally, it has been supposed a non-holonomic motion of the human obstacles that allows to define a more natural behavior during human obstacle evasion. As a complementary part of the motion control, it has been proposed a flexible formation control where the human is considered as part of the formation. For this, it has been necessary to consider a leader/human-centered scheme. Finally, simulation results are presented to contrast the positive effect that has the inclusion of the potential field variation during the motion of the human obstacle, regarding lower meddling in the social zone of the human obstacles. It is believed that the proposed algorithm improves the social acceptance of mobile robots during interactions with humans.

REFERENCES

Antonelli, G., Arrichiello, F., and Chiaverini, S. (2009). Experiments of formation control with multirobot systems using the null-space-based behavioral control. *IEEE Transactions on Control Systems Technology*, 17(5), 1173–1182.

Antonelli, G., Arrichiello, F., and Chiaverini, S. (2005). Experimental kinematic comparison of behavioral approaches for mobile robots. *IFAC Proceedings Volumes*, 38(1), 295–300.

Arechavaleta, G., Laumond, J.P., Hicheur, H., and Berthoz, A. (2006). The nonholonomic nature of human locomotion: a modeling study. In *IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics*, 158–163.

Baillieul, J., Hollerbach, J., and Brockett, R. (1984). Programming and control of kinematically redundant manipulators. In *23rd IEEE Conference on Decision and Control*, 768–774.

Chi-Pang, L., Chen-Tun, C., Kuo-Hung, C., and Li-Chen, F. (2011). Human-centered robot navigation towards a harmoniously human robot coexisting environment. *IEEE Transactions on Robotics*, 27(1), 99–112.

Chiaverini, S. (1997). Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Transactions on Robotics and Automation*, 13(3), 398–410.

Doisy, G., Jevtić, A., Lucet, E., and Edan, Y. (2012). Adaptive person-following algorithm based on depth images and mapping. *Proc. of the IROS Workshop on Robot Motion Planning*.

Fiorini, P. and Shiller, Z. (1998). Motion planning in dynamic environments using velocity obstacles. *The International Journal of Robotics Research*, 7, pp. 760772.

Guansheng, X., Shuangna, T., Hexu, S., Weipeng, L., and Huawang, L. (2013). People-following system design for mobile robots using Kinect sensor. In *25th ChineseControl and Decision Conference (CCDC)*, 3190–3194.

Guzzi, J., Giusti, A., Gambardella, L.M., Theraulaz, G., and Di Caro, G.A. (2013). Human-friendly robot navigation in dynamic environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, 423–430.

Hall, E.T. (1963). A system for the notation of proxemic behavior. *American Anthropologist*, 65(5), 1003–1026.

Kluge, E.P., Bank, D., and Boris (2002). Key technologies in robot assistants: Motion coordination between a human and a mobile robot. *Transactions on Control, Automation and Systems Engineering*, 4.

Leica, P., Toibero, J.M., Roberti, F., and Carelli, R. (2014). Switched control to robot-human bilateral interaction for guiding people. *Journal of Intelligent and Robotic Systems*, 77(1), 73–93.

Loper, M.M., Koenig, N.P., Chernova, S.H., Jones, C.V., and Jenkins, O.C. (2008). Mobile human-robot teaming with environmental tolerance. In *International Conference on Human-Robot Interaction (HRI)*, 157–163.

Machida, E., Meifen, C., Murao, T., and Hashimoto, H. (2012). Human motion tracking of mobile robot with Kinect 3D sensor. In *SICE Annual Conference (SICE)*, 2207–2211.

Ratsamee, P., Mae, Y., Ohara, K., Kojima, M., and Arai, T. (2013). Social navigation model based on human intention analysis using face orientation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1682–1687.

Scandolo, L. and Fraichard, T. (2011). An anthropomorphic navigation scheme for dynamic scenarios. In *IEEE International Conference on Robotics and Automation (ICRA)*, 809–814.

Whitney, D.E. (1969). Resolved motion rate control of manipulators and human prostheses. *IEEE Transactions on Man-Machine Systems*, 10(2), 47–53.

Observer Designs for a Turbocharger System of a Diesel Engine

María Tejada Zuñiga * **Matti Noack** ** **Johann Reger** **

* *Pontificia Universidad Católica del Perú, Lima, Peru (e-mail: mctejadaz@pucp.pe)*
** *Control Engineering Group, Technische Universität Ilmenau, P.O. Box 10 05 65, D-98684, Ilmenau, Germany (e-mail: matti.noack@tu-ilmenau.de, reger@ieee.org)*

Abstract: We consider observer designs for an exhaust-gas turbocharger based on a third order nonlinear model of the air path of a diesel engine. To propose an observer, the system is analyzed in view of different observability properties for nonlinear systems. We derive explicit constraints for the system state guaranteeing observability. Based on that we design a high-gain observer and a sliding mode observer for the system. The performance of both designs has been assessed through a simulation study.

Keywords: Nonlinear Systems, Observability Analysis, High-Gain Observer, Sliding-Mode Observer, Turbocharger Diesel Engine

1. INTRODUCTION

Over the last decades, diesel engines have gained interest because they provide higher fuel efficiency than gasoline engines. However, diesel engines have the disadvantage to require a complicated exhaust gas treatment. One way to achieve a better power utilization and an improved treatment of exhaust gases is the use of turbochargers. Fig. 1 shows the structure of such turbocharger system.

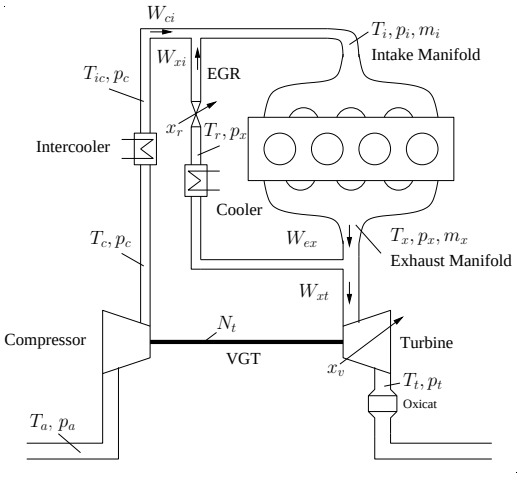


Fig. 1. Schematic diagram of a turbocharged diesel engine with EGR (Herceg et al., 2006).

The function of the turbocharger is to increase the power density of the engine by pushing additional fresh air into the cylinders of the engine in order to burn more fuel without smoke generation, allowing for the injection of more fuel without reaching the smoke bound. The turbine has a variable geometry (VGT) so as to adapt the turbine efficiency based on the operating point of the engine, driven by the energy in the exhaust gas.

The second feedback path to the intake manifold from the exhaust manifold is due to exhaust gas recirculation which is controlled by an exhaust gas recirculation (EGR) valve. The oxygen in the intake is replaced by recirculated exhaust gases, thereby reducing the temperature profile of the combustion and hence the emissions of nitrogen oxides. The interactions are relatively complex. The VGT actuator is usually used to control the intake manifold absolute pressure (MAP) while the EGR valve controls the mass air flow (MAF) into the cylinders. Both the EGR and VGT paths are driven by the exhaust gases. However, both the EGR and VGT actuator are equipped with position sensors. For a detailed description refer to Ladommatos et al. (1996) and references therein.

Vast research has been dedicated to the control of diesel engines. Many controllers were proposed in the literature, e.g., Lyapunov controller design (Janković and Kolmanovsky, 2000), controllers based on a Linear Parameter-Varying (LPV) model for turbocharged diesel engine (Jung and Glover, 2003), indirect passivation (Larsen et al., 2000), predictive control (Ferreau et al., 2007), feedback linearization (Dabo et al., 2009) and sliding mode control (Utkin et al., 2000). Other approaches for air-to-fuel ratio control of turbocharged diesel engines were also proposed in the literature (Fredriksson, 1999; Guzzella and Onder, 2013; Janković et al., 1998). The different solutions are mostly nonlinear and model-based. Usually it is assumed that the state variables be measurable or somehow computable.

However, full state information is not the case for many systems in practice, as was explained in Fredriksson and Egardt (2002). The intake manifold pressure or boost pressure is measurable. The compressor power may be

computed in principle. But the sensors used for computing the compressor power are quite expensive, and, therefore, it is often assumed to be unmeasurable. So as to relax this assumption the use of observers for estimating the unmeasurable states may be considered a solution; first designs have been proposed only in the recent past (Glenn et al., 2011; Salehi et al., 2013; Qu et al., 2015).

The focus of this contribution is on estimating the states of this turbocharger model. To this end, the model is analyzed in terms of observability first. Within the observability analysis we shall consider two concepts of observability for nonlinear systems, namely, weak observability and uniform observability. Local weak observability is an indispensable condition for the design of observers, see Hermann and Krener (1977). The uniform observability, as introduced by Gauthier and Bornard (1981), is a stronger concept and allows for the stable design of High-Gain observers. Afterwards, we investigate the application of two different observer approaches on the turbocharger system: a high-gain observer and a sliding-mode observer. The performances of both are compared in terms of design, complexity and robustness.

The paper is structured as follows: The following section recalls the nonlinear model of the turbocharger system, based on a simplification of a more complete physical model. Section 3 presents the observability analysis of the nonlinear system using the observability rank condition analytically and more extensive numerical calculations. In Sections 4 and 5 we present the high-gain observer and sliding-mode observer design, respectively. Both observers are simulated and results are presented in Section 6. Conclusions are drawn in Section 7.

2. MODELING

The modeling of the air path system of a diesel engine, which is composed of a turbocharger with variable turbine geometry, exhaust gas recirculation and intercooler, is not an easy task and consists largely of identification and map creation, as Wahlström and Eriksson (2011) had presented in detail. Moreover, complicated functional relationships arise, for example, by attaching flow functions.

Through versatile measurements describing the couplings, a comprehensive information flow exists for the system which then can be used for observation. However, this results in a very complex mathematical model to be mastered. Fortunately, simplifications and approximations have been made.

Against this background, here we use a fully parameterized nonlinear model of the turbocharger which is based on the approach from Janković and Kolmanovsky (2000). This is a simplified third-order nonlinear model derived from an eighth-order nonlinear mean-value model of the air path of a turbocharged diesel engine with EGR and VGT. The full-order air path model is of eighth order with states: intake and exhaust manifold pressure (p_i and p_x), oxygen mass fractions in the intake and exhaust manifolds, turbocharger speed and the two states describing the actuator dynamics for the two control signals.

For more simplicity the model is reduced to a third-order model subject to the following assumptions: The oxygen mass fraction variables are ignored since they are not coupled with other engine dynamics and are difficult to measure. The temperature variables are omitted because temperature dynamics are slow when compared with pressure and flow dynamics.

The detailed derivation of the eighth-order nonlinear model of the engine under investigation may be found in Jung et al. (2002). In Table 2 all variables and parameters are listed. Reducing the consideration to the third-order model, we may refer to just three states p_i, p_x, and P_c that represent the intake manifold pressure, the exhaust manifold pressure, the power transferred by the turbocharger, respectively. Then the following system description is obtained:

$$\Sigma: \begin{cases} \dot{p}_i = \dfrac{RT_i}{V_i}(W_{ci} + W_{xi} - W_{ie}) \\ \dot{p}_x = \dfrac{RT_x}{V_x}(W_{ie} + W_f - W_{xi} - W_{xt}) \\ \dot{P}_c = \dfrac{1}{\tau}(-P_c + P_t) \end{cases} \tag{1}$$

Following the analysis of the turbine mass-flow presented by Schollmeyer (2010), the flow rates as well as the compressor and turbine power are given by

$$\begin{aligned} W_{ci} &= \frac{\eta_c}{c_p T_a} \frac{P_c}{(\frac{p_i}{p_a})^{\mu} - 1}, \\ W_{xi} &= \frac{A_{egr}(x_{egr}) p_x}{\sqrt{RT_x}} \sqrt{\frac{2p_i}{p_x}(1 - \frac{p_i}{p_x})}, \\ W_{ie} &= \eta_v \frac{p_i N V_d}{120 R T_i}, \\ W_{xt} &= (a x_{vgt} + b) \left(c \left(\frac{p_a}{p_x} - 1 \right) + d \right) \frac{p_x}{p_{ref}} \\ &\quad \times \sqrt{\frac{T_{ref}}{T_x}} \sqrt{\frac{2p_i}{p_x} \left(1 - \frac{p_a}{p_x} \right)}, \\ P_t &= W_{xt} c_p T_x \eta_t \left(1 - \left(\frac{p_a}{p_x} \right)^{\mu} \right) \end{aligned}$$

with the constant parameters: compressor and turbine efficiency η_c and η_t, volumetric efficiency η_v, intake manifold temperature T_i, exhaust manifold temperature T_x, and time constant of the turbocharger power transfer τ.

The control inputs of the diesel engine model based on (1), are the EGR position x_{egr} and the VGT position x_{vgt}. For simplifying the considerations of the model, however, the effective areas of the valves, i.e. $A_{vgt} = a x_{vgt} + b$ and A_{egr}, are used as control inputs. Furthermore, the control inputs A_{egr} and A_{vgt} are constrained due to the minimal and maximal EGR and VGT positions. The intake manifold pressure or boost pressure, is measurable and forms the system output.

It should be noted that the model has a singularity when the pressure in the intake manifolds equals the ambient pressure, that is, if $p_i = p_a$ then the compressor flow becomes infinite. There exists also another singularity when the turbine stalls. The simplified model can not handle this situation and, as explained in (Fredriksson and Egardt, 2002), the system is only valid on the set $\Omega = \{(p_i, p_x, P_c) : p_i > p_a, p_x > p_a, P_c > 0\}$. Fortunately,

it may be demonstrated that Ω is invariant, thus, every trajectory starting in Ω stays in Ω (Janković et al., 1998).

Table 1. Parameters of the model (Janković and Kolmanovsky, 2000)

	Description	Value	Unit
N	Engine Speed	2000	$[rpm]$
W_f	Fuel rate	5	$[kg/h]$
R	Specific gas constant	287	$\left[\frac{J}{kg\cdot K}\right]$
p_i	Intake manifold pressure	-	$[hPa]$
p_x	Exhaust manifold pressure	-	$[hPa]$
p_{ref}	Reference pressure	101.3	$[hPa]$
p_a	Ambient pressure	101.3	$[hPa]$
P_c	Compressor Power	-	$[kW]$
T_i	Intake manifold temperature	313	$[K]$
T_x	Exhaust manifold temperature	509	$[K]$
T_{ref}	Referent Temperature	298	$[K]$
T_a	Ambient Temperature	298	$[K]$
V_i	Volume of the intake manifold	0.006	$[m^3]$
V_x	Volume of the exhaust manifold	0.001	$[m^3]$
V_d	Displacement Volume	0.002	$[m^3]$
η_m	Turbo mechanical efficiency	98	[%]
η_c	Compressor isentropic efficiency	61	[%]
η_t	Turbine isentropic efficiency	76	[%]
η_v	Volume efficiency	87	[%]
c_P	Specific heat at constant pressure	1014.4	$\left[\frac{J}{kg\cdot K}\right]$
c_v	Specific heat at constant volume	727.4	$\left[\frac{J}{kg\cdot K}\right]$
μ	constant	0.286	–
a	Parameter a	−0.136	–
b	Parameter b	0.176	–
c	Parameter c	0.4	–
d	Parameter d	0.6	–

For our analysis we may refer to system (1) in terms of system

$$\Sigma: \begin{cases} \dot{x} = F(x,u)\,, \\ y = h(x)\,, \end{cases} \tag{2}$$

where here $F: \mathbb{R}^3 \times \mathbb{R}^2 \to \mathbb{R}^3$ and $h: \mathbb{R}^3 \to \mathbb{R}$ differentiable in x sufficiently often. The function expressions of $F(x,u)$ correspond to the right side of the system (1), that is $x = (p_i, p_x, P_C)^\top$ and $h(x) = x_1$.

3. OBSERVABILITY

For nonlinear systems there exist various observability concepts. In this contribution, in particular, we consider *weak observability* and *uniform observability* which are significant for the observer approaches to be applied later, following Besançon (2007).

Consider a single-output system of class Σ from (2). In general, the observability map is

$$\mathfrak{o}_n(x) := \begin{pmatrix} h(x) \\ \mathcal{L}_F h(x) \\ \mathcal{L}_F^2 h(x) \\ \vdots \\ \mathcal{L}_F^{n-1} h(x) \end{pmatrix},$$

where $\mathcal{L}_F^k$ describes the k-th Lie derivative with respect to $F(\cdot, u)$. Observability may then be characterized by the solvability of the resulting system of nonlinear equations for the state x. In view of the implicite function theorem, the Jacobian of this map may be used for showing *local solvability*. The Jacobian of the observability map is called *observability matrix*, given by

$$\mathcal{O}_n(x) := \begin{pmatrix} \frac{\partial}{\partial x} h(x) \\ \frac{\partial}{\partial x} \mathcal{L}_F h(x) \\ \frac{\partial}{\partial x} \mathcal{L}_F^2 h(x) \\ \vdots \\ \frac{\partial}{\partial x} \mathcal{L}_F^{n-1} h(x) \end{pmatrix},$$

where $\frac{\partial}{\partial x}(\cdot)$ denotes the gradient.

Here it is important to note that the observability map and its Jacobian will actually also depend on the input. Excluding that these inputs may cause conflicts with observability, naturally, leads to the stronger concept of *uniform observability*. A system is uniformly observable if the observability map may be solved uniquely for the state, whatever the input. In this case, every input is a so-called universal input.

The analytical expression of the observability matrix for the turbocharger system is rather lengthly, and not easy to analyze. In order to get an impression of observability, the analysis is made first for the free system, i.e. with inputs indentical to zero. In this case the observability matrix is

$$\mathcal{O}_3(x) = \begin{pmatrix} 1 & 0 & 0 \\ * & \beta_1(x_1) & 0 \\ * & * & \beta_1(x_1)\dfrac{\partial \beta_3(x_2)}{\partial x_2} \end{pmatrix} \tag{3}$$

where

$$\beta_1(x_1) = \frac{RT_i}{V_i} \frac{\eta_c}{c_p T_a} \frac{1}{\left(\left(\frac{p_i}{p_a}\right)^{\mu} - 1\right)}$$

$$\beta_3(x_2) = \frac{c_p \eta_t T_x}{\tau} \left(1 - \frac{p_a}{p_x}^{\mu}\right) b \left(c\left(\frac{p_x}{p_a} - 1\right) + d\right) \frac{p_x}{p_{ref}} \times \sqrt{\frac{T_{ref}}{T_x}} \sqrt{\frac{2p_a}{p_x}(1 - \frac{p_a}{p_x})}$$

In order to probe whether the autonomous system is at least *locally weakly observable*. For showing this, we use the observability rank condition presented by Hermann and Krener (1977), i.e., we check whether $\operatorname{rank}\big(\mathcal{O}_3(x)\big) = 3$. Thus, it is only necessary to verify that its diagonal elements are different from zero.

In the region of interest Ω we have that $\beta_1(x_1) \neq 0$. Fig. 2 shows that the values of the third element of the diagonal will also never be zero. Therefore we conclude that the system is at least locally weakly observable on the set Ω.

The remaining analysis is concerned with the evaluation of the inputs to determine whether these are universal. The complete model in affine form is given by

$$F(x,u) = \begin{pmatrix} * \\ * \\ * \end{pmatrix} + \begin{pmatrix} * \\ * \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ * \\ * \end{pmatrix} u_2,$$

in this way it is easy to see that the inputs are not coupled. Then for simplifying the analysis, we analyzed each input independent from each other. That is, analyzing u_1 we set $u_2 \equiv 0$ and vice versa. It turns out that the

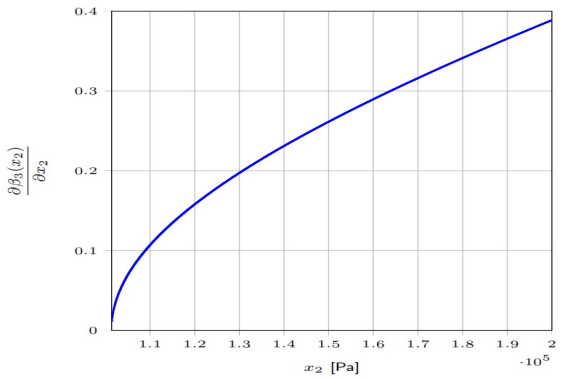


Fig. 2. Values of $\frac{\partial \beta_3(x_2)}{\partial x_2}$ in the region of interest

respective observability matrices $\mathcal{O}_3(x, u_i)$ do not show triangular structure. Therefore, the analytic calculation of its determinant results in a complex task even with the help of computational software. However, for getting an impression we may plot the determinant as a function of states, considering the dynamics of the system and the entire range of input values. Note that the range of operation was obtained fixing two states maximizing the other, Fig. 3 shows that the determinant of $\mathcal{O}_3(x, u_1)$ is never zero for the relevant range of states, analogously Fig. 4 shows the same result for the second input.

Therefrom we may assume that there is no relevant states in which the determinant becomes zero. In view of this analysis, each input turns out an universal input and we may conclude that the turbocharger system is uniformly observable in the operation area of validity Ω.

4. HIGH-GAIN OBSERVER

The rationale of the high-gain (HG) observers is to find a linear output injection which is able to dominate the nonlinear terms for stabilizing the estimation error dynamics, which is governed by a constant gain matrix. Due to their inherent simple structure, HG observers are frequently used in practical applications. This also makes the HG observer applicable to the turbocharger system. It is simple practice to choose the observer gain via eigenvalue placement (Röbenack, 2012), however, formal existence conditions are more complicated. In many cases, there is a finite bound on the maximum feasible Lipschitz constant of the nonlinear part for which the error dynamics can be stabilized. The existing results can be improved significantly if the structure of the linear part is taken into account.

Following Gauthier et al. (1992) and Moreno and Vargas (2000) assume the non-linear system in the form (2). Due to the uniform observability of system (1), $z = \mathfrak{o}_3(x)$ is a local diffeomorphism and the system may be transformed into nonlinear observer normal form (ONF). That is, we have

$$\left.\begin{aligned} \dot{x} &= F(x,u) \\ y &= x_1 \end{aligned}\right\} \quad \underset{\mathfrak{o}_3^{-1}(z)}{\overset{\mathfrak{o}_3(x)}{\rightleftharpoons}} \quad \left\{\begin{aligned} \dot{z} &= Az + b\,\varphi(z,u,\dot{u}) \\ y &= c^\top z \end{aligned}\right. \tag{4}$$

where

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad c^\top = (1\ 0\ 0)\,.$$

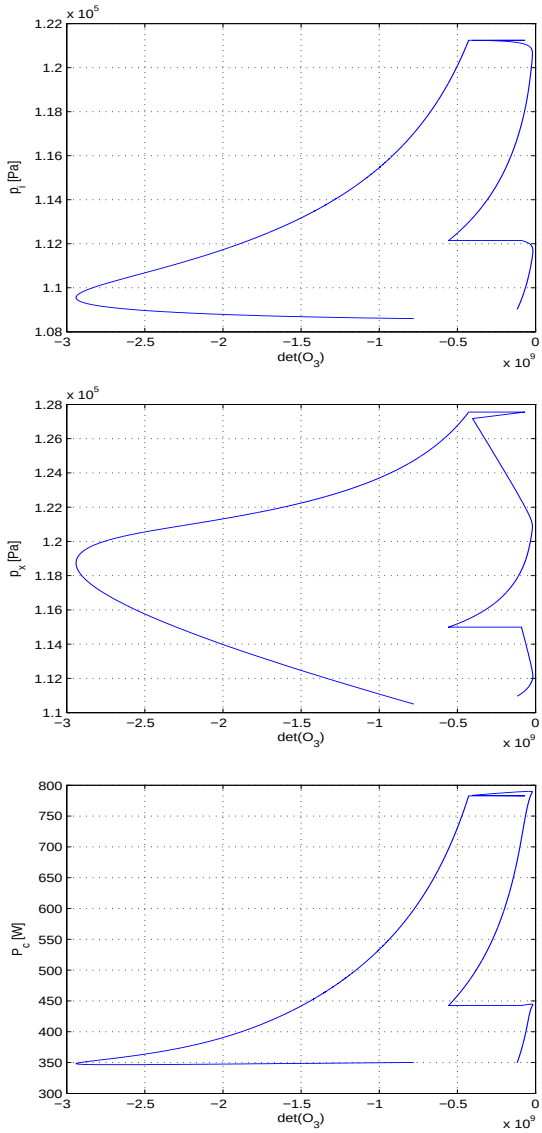


Fig. 3. Range of the determinant of $\mathcal{O}_3(x, u_1)$

In this equation, function φ describes the nonlinearities of the system. Furthermore, we will take benefit from

$$\dot{x} = \left(\frac{\partial \mathfrak{o}_3(x)}{\partial x}\right)^{-1} \dot{z} \tag{5}$$

which is very useful for the observers design in original coordinates. This way, only the observability matrix $\mathcal{O}_3(x)$ needs to be inverted which is much simpler than determining the complete inverse transformation of a nonlinear mapping.

In view of the global observability of the system in ONF (4) we design the observer

$$\begin{aligned} \dot{\hat{z}} &= A\hat{z} + b\,\varphi\left(\hat{z}, u, \dot{u}\right) - l(\epsilon)(\hat{y} - y) \\ \hat{y} &= c^\top \hat{z} \end{aligned} \tag{6}$$

with

$$l(\epsilon) = \begin{pmatrix} \epsilon^{-1} & 0 & 0 \\ 0 & \epsilon^{-2} & 0 \\ 0 & 0 & \epsilon^{-3} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \end{pmatrix} = \Lambda(\epsilon) k$$

for some $\epsilon > 0$ and $k_i > 0$ such that they correspond to the coefficients of a Hurwitz polynomial. Choosing ϵ

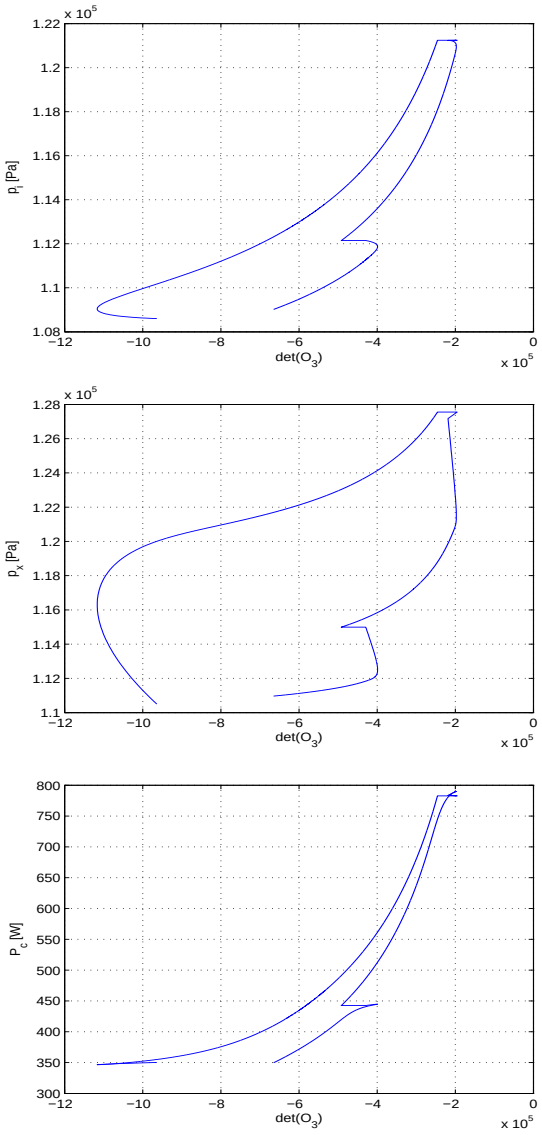


Fig. 4. Range of the determinant of $\mathcal{O}_3(x, u_2)$

sufficiently small, the observers will have a stable state estimation error dynamics. It is clear that for small ϵ the diagonal entries of $\Lambda(\epsilon)l$ will be large, thus the name *high-gain observer*.

In original coordinates this state observer takes the form

$$\begin{aligned} \dot{\hat{x}} &= F(\hat{x}, u) - \left(\frac{\partial \mathfrak{o}_3(\hat{x})}{\partial \hat{x}}\right)^{-1} l(\epsilon)(\hat{y} - y) \\ \hat{y} &= \hat{x}_1 \end{aligned} \tag{7}$$

which shows just the necessity to invert the Jacobian.

5. SLIDING MODE OBSERVER

Continuous observers usually converge to the origin only in the absence of disturbances. Sliding-mode (SM) observers have been designed to resolve this issue by using discontinuous or non-Lipschitz continuous injection terms. SM observers, generally speaking, are known to show a remarkable robustness against disturbances and may further show a finite convergence time, at least theoretically. Different designs for SM observers were proposed, as for example, the generalized super-twisting observer (GSTO) and homogeneous SM observers Moreno (2013). Those approaches may guarantee different properties, but are well-studied only for second order systems. The SM observer that we adopt here for the third order system (1) was first proposed in (Moreno and Dochain, 2013). For a system in ONF this observer takes the form

$$\begin{aligned} \dot{\hat{z}}_1 &= -L\, k_1 |\hat{y} - y|^{\frac{2}{3}} \operatorname{sign}(\hat{y} - y) + \hat{z}_2 \\ \dot{\hat{z}}_2 &= -L^2\, k_2 |\hat{y} - y|^{\frac{1}{3}} \operatorname{sign}(\hat{y} - y) + \hat{z}_3 \\ \dot{\hat{z}}_3 &= -L^3\, k_3 \operatorname{sign}(\hat{y} - y) + \phi(\hat{z}, u, \dot{u}) \\ \hat{y} &= \hat{z}_1 \end{aligned}$$

The gains k_1, k_2 and k_3 again are all positive, selected in order to guarantee the convergence of the observer in absence of external perturbations. The choice of gains may follow the condition $k_1 k_2 > k_3$. It may be motivated by imposing a Hurwitz polynomial for a linear error dynamics. Thus, one is left with choosing some appropriate $L > 0$.

In the original coordinates, the SM observer for system (1) again only requires the inversion of $\mathcal{O}_3(x)$. In order to have a representation similar to the HG observer let $L = \frac{1}{\epsilon}$. Then the observer in original coordinates reads

$$\begin{aligned} \dot{\hat{x}} &= F(\hat{x}, u) - \left(\frac{\partial \mathfrak{o}_3(\hat{x})}{\partial \hat{x}}\right)^{-1} l(\epsilon) \begin{pmatrix} |\hat{y} - y|^{\frac{2}{3}} \operatorname{sign}(\hat{y} - y) \\ |\hat{y} - y|^{\frac{1}{3}} \operatorname{sign}(\hat{y} - y) \\ \operatorname{sign}(\hat{y} - y) \end{pmatrix} \\ \hat{y} &= \hat{x}_1 \end{aligned} \tag{8}$$

Stability and convergence of the associated state estimation error dynamics was shown by Moreno and Dochain (2013).

6. RESULTS

The performance of the observers designed for the turbocharger system was studied by simulations performed in MATLAB. The three states of the turbocharger model shall be estimated from the input manifold pressure p_i as the ouput and the inputs. The values of the normalized inputs for the scenario of simulation were the EGR position $x_{egr} = 0.7$ and the VGT position $x_{vgt} = 0.6$.

The state estimation errors obtained with the HG observer, using different ϵ values are shown in Fig. 5. Furthermore, the values of constants design k_1, k_2 and k_3 were set considering three poles placed at -10. It can be seen that the one with smallest ϵ yields a more pronounced peaking, as expected. Besides the peaking, as known from the literature, for smaller ϵ we also expect

- faster convergence of the observation error,
- amplification of the measurement noise,
- better attenuation of process disturbances.

These trade-offs always have to be taken into account in the design procedure.

The state estimation errors using the SM observer with different values of $L = \frac{1}{\epsilon}$ where $L_1 < L_2 < L_3$ are shown in Fig. 6. For simulating comparable dynamics, the remaining

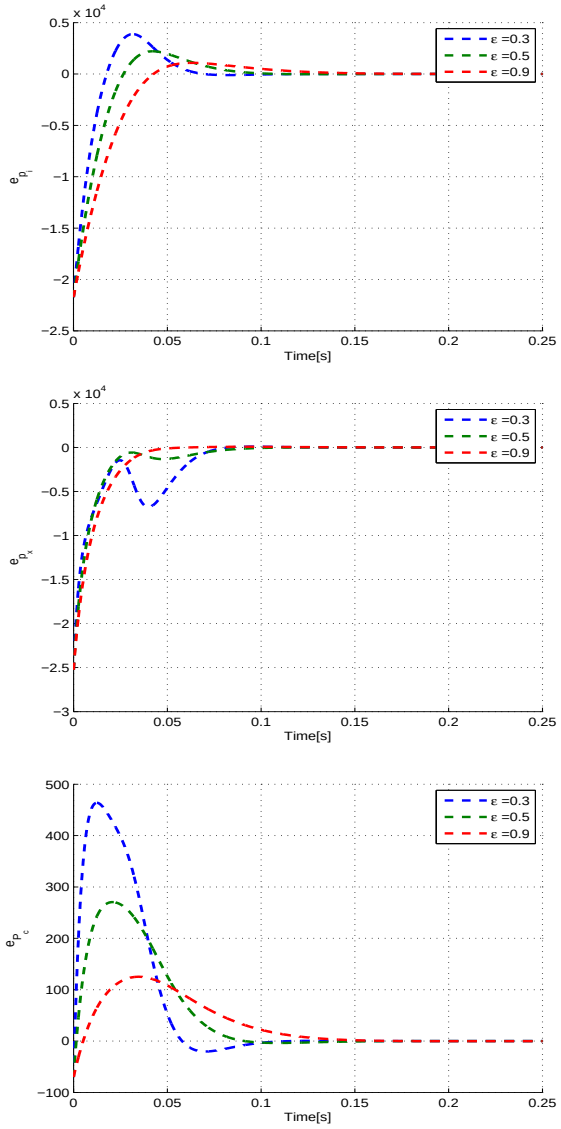


Fig. 5. Estimation errors using the high-gain observer

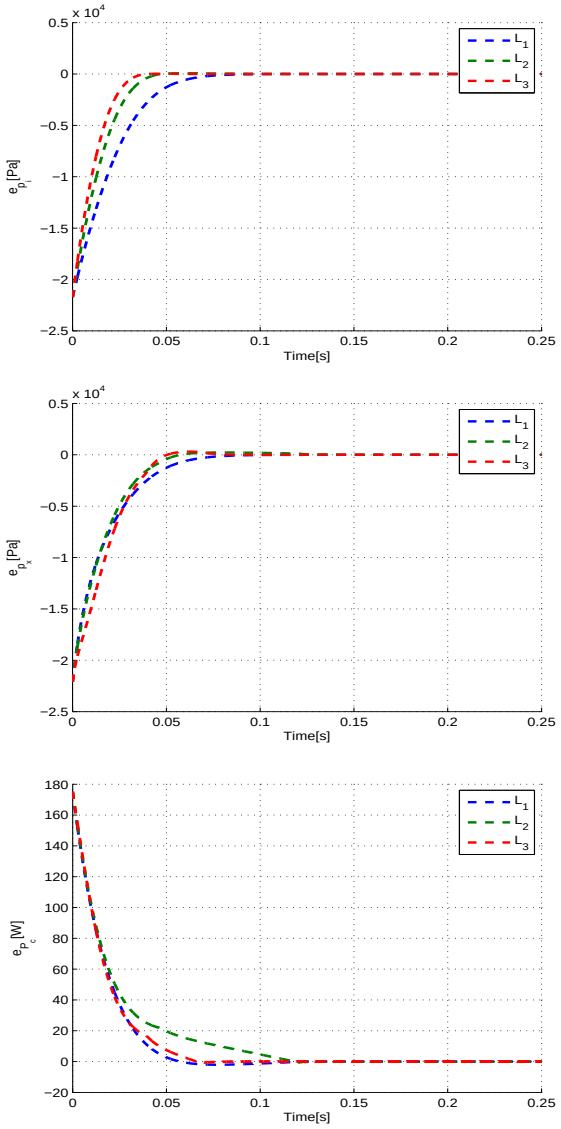


Fig. 6. Estimation errors using the sliding-mode observer

design values have been kept as in the same. No particular parameter tuning was made.

It can be seen that the one with largest L yields a faster convergence of the observation error. Furthermore, this method improves the HG observer since, remarkably, the peaking phenomenon turns out not present in the results. The performance of the designed observers has been assessed by imposing measurement noise. More precisely, in the simulations the measured output was contaminated by additive normally distributed noise with zero mean and standard deviation 3.4832×10^3. The transient performance of the estimation errors is shown in Figures 7–9.

The convergence velocity of the observation error increases while the gain of the nonlinear part increases. There exists a trade-off for the HG observer in the selection of this gain. That is, for high gains the perturbation is dominated better, but to the price that the measurement noise is amplified and the peaking phenomenon is more pronounced.

On the other hand, the Figures 8–9 show a slower convergence of the error for the Sliding Modes observer, however, in this case, the measurement noise is not present in the results.

For the two observer approaches taken in consideration the estimated intake and exhaust manifold pressure as well as the compressor power are very close to the state variables of the real system.

7. CONCLUSION

The observability analysis for a model of turbocharger in a diesel engine has been performed using weak and uniform observability concepts. Operating the system in the invariant set of the usual operation range we have shown that the system is uniformly observable. Based on this, we have designed a high-gain observer and a sliding-mode observer so as to obtain estimates for the turbocharger states. Imposing a similar estimation error dynamics, simulations with experimental testbed-data show that both observers are able to reconstruct suitably the desired states. How-

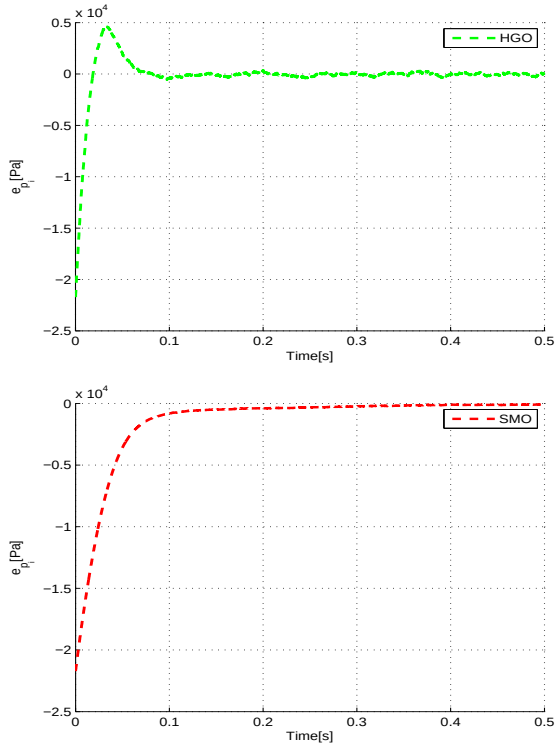


Fig. 7. Estimation error wrt. state p_i in presence of noise

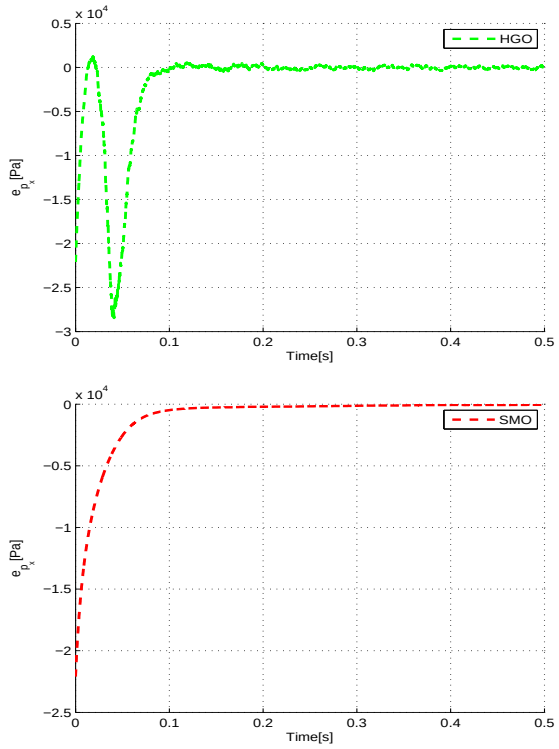


Fig. 8. Estimation error wrt. state p_x in presence of noise

ever, simulations also show that the sliding-mode observer shows no peaking and is less prone to noise.

Future work will concentrate on the design of output-feedack controllers for the turbocharger system, based on the presented observer studies.

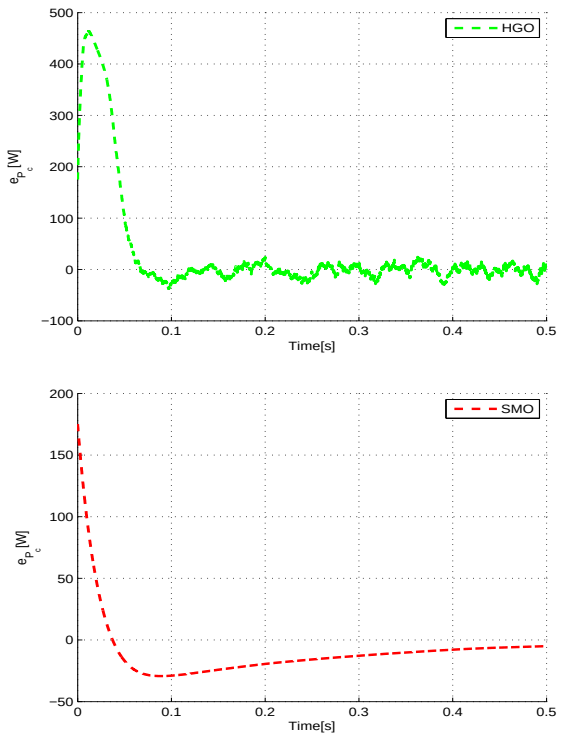


Fig. 9. Estimation error wrt. state P_C in presence of noise

ACKNOWLEDGMENT

The authors gratefully acknowledge insightful discussions with Jaime Moreno on the observability properties and observer design for nonlinear systems.

REFERENCES

Besançon, G. (2007). An overview on observer tools for nonlinear systems. In G. Besançon (ed.), *Nonlinear observers and applications*, volume 363 of *Lecture notes in control and information sciences*, 1–33. Springer, Berlin.

Dabo, M., Langlois, N., and Chafouk, H. (2009). Dynamic feedback linearization applied to asymptotic tracking: generalization about the turbocharged diesel engine outputs choice. In *American Control Conf.*, 3458–3463.

Ferreau, H.J., Ortner, P., Langthaler, P., del Re, L., and Diehl, M. (2007). Predictive control of a real-world diesel engine using an extended online active set strategy. *Annual Reviews in Control*, 31(2), 293–301.

Fredriksson, J. (1999). *Nonlinear control of turbocharged diesel engines*. Ph.D. thesis, Department of Signals and Systems, Chalmers University of Technology.

Fredriksson, J. and Egardt, B. (2002). Estimating exhaust manifold pressure in a turbocharged diesel engine. In *IEEE Conf. on Control Appl.*, 701–706.

Gauthier, J.P. and Bornard, G. (1981). Observability for any u(t) of a class of nonlinear systems. *IEEE Trans. on Autom. Control*, 26(4), 922–926.

Gauthier, J.P., Hammouri, H., and Othman, S. (1992). A simple observer for nonlinear systems applications to bioreactors. *IEEE Trans. on Autom. Control*, 37(6), 875–880.

Glenn, B.C., Upadhyay, D., Utkin, V.I., Washington, G.N., and Hopka, M.B. (2011). Observer design of critical

states for air path flow regulation in a variable geometry turbocharger exhaust gas recirculation diesel engine. *Int. Journal of Engine Research*, 12(6), 501–512.

Guzzella, L. and Onder, C. (2013). *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Springer, Berlin.

Herceg, M., Raff, T., Findeisen, R., and Allgöwer, F. (2006). Nonlinear model predictive control of a turbocharged diesel engine. In *IEEE Conf. on Control Appl.*, 2766–2771.

Hermann, R. and Krener, A.J. (1977). Nonlinear controllability and observability. *IEEE Trans. on Automatic Control*, 22(5), 728–740.

Janković, M., Janković, M., and Kolmanovsky, I. (1998). Robust nonlinear controller for turbocharged diesel engines. In *American Control Conf.*, 1389–1394.

Janković, M. and Kolmanovsky, I. (2000). Constructive Lyapunov control design for turbocharged diesel engines. *IEEE Trans. on Control Systems Techn.*, 8(2), 288–299.

Jung, M. and Glover, K. (2003). Control-oriented linear parameter-varying modelling of a turbocharged diesel engine. In *IEEE Conf. on Control Appl.*, 155–160.

Jung, M., Ford, R.G., Glover, K., Collings, N., Christen, U., and Watts, M.J. (2002). Parameterization and transient validation of a variable geometry turbocharger for mean-value modeling at low and medium speed-load points. Technical report, SAE.

Ladommatos, N., Balian, R., Horrocks, R., and Cooper, L. (1996). The effect of exhaust gas recirculation on combustion and nox emissions in a high-speed direct-injection diesel engine. Technical report, SAE.

Larsen, M., Janković, M., and Kokotović, P.V. (2000). Indirect passivation design for a diesel engine model. In *IEEE Conf. on Control Appl.*, 309–314.

Moreno, J.A. (2013). On discontinuous observers for second order systems: properties, analysis and design. In *Advances in Sliding Mode Control*, 243–265. Springer.

Moreno, J.A. and Dochain, D. (2013). Finite time converging input observers for nonlinear second-order systems. In *52th IEEE Conf. on Decision and Control*, 3554–3559.

Moreno, J.A. and Vargas, A. (2000). Approximate high-gain observers for uniformly observable nonlinear systems. In *39th IEEE Conf. on Decision and Control*, 784–789.

Qu, Q., Wang, H., and Tian, Y. (2015). Nonlinear observer based sliding mode control for a turbocharged diesel engine air-path equipped with EGR and VGT. In *Chinese Automation Congress*, 121–126. Wuhan.

Röbenack, K. (2012). Structure matters – some notes on high gain observer design for nonlinear systems. *Systems, Signals and Devices*, 9, 1–7.

Salehi, R., Shahbakhti, M., Alasty, A., and Vossoughi, G. (2013). Nonlinear observer design for turbocharger in a si engine. In *American Control Conf.*, 5231–5236.

Schollmeyer, M. (2010). *Beitrag zur modellbasierten Ladedruckregelung für Pkw-Dieselmotoren*. Ph.D. thesis, Leibniz Universtät Hannover.

Utkin, V., Chang, H.C., Kolmanovsky, I., Cook, J., et al. (2000). Sliding mode control for variable geometry turbocharged diesel engines. In *American Control Conf.*, 584–588.

Wahlström, J. and Eriksson, L. (2011). Modelling diesel engines with a variable-geometry turbocharger and exhaust gas recirculation by optimization of model parameters for capturing non-linear system dynamics. *Journal of Automobile Engineering*, 225(7), 960–986.

Optimal Power Dispatch in a Microgrid

Martin J. Loza-Lopez * **Riemann Ruiz-Cruz** **
T.B. Lopez-Garcia * **Edgar N. Sanchez** *
Juan Diego Sánchez-Torres **

* *Centro de Investigación y Estudios Avanzados del IPN, Zapopan, Jal 45030 México, (e-mail: {mdloza,tblopez,sanchez}@ gdl.cinvestav.mx)*
** *ITESO University, Periferico Sur Gomez Morin 8585, Tlaquepaque, Jalisco, México C.P. 45604 ,(e-mail: {riemannruiz,dsanchez}@iteso.mx)*

Abstract: This paper is concerned with the power dispatch in a microgrid. The dispatch problem is formulated as linear program. Thus, the proposed solution is the application of neural network that solves linear programming on-line. This proposal in motivated by the increasing electric energy demand and the rising need to incorporate sustainable energy sources to the power grid in a reliable scheme. A microgrid is an interconnection of distributed energy sources (DES), with the tendency to include renewable energies that offer many advantages to customers and utilities. The different DES that compose the microgrid are controlled independently to track the optimal reference provided by the proposed method in order to supply a demanded power output minimizing the consumed power from the main grid.

Keywords: Electric energy distribution, Neurodynamic optimization, Linear programming, Power management

1. INTRODUCTION

The optimization of the power dispatch within a microgrid is a big challenge for many engineering areas as control, power electronics and modeling. Different studies have been performed in this area, some examples are presented in Chowdhury and Crossley (2009). In the presented work, the optimal amount of power to be supplied by each energy source in a microgrid simulation, to produce a maximum amount of energy, is analyzed. Because of the varying output power that renewable energy sources present, large-scale, real-time optimization procedures are required, most of them in the form of linear programming. In contrast to the publications which use recurrent neural networks for microgrid optimization (Aquino et al., 2010), the proposed approach provides *fixed convergence time* to the solution and the tuning of only one network parameter.

The optimization problem is stated to maximize or minimize an objective function with the manipulation of the value of decision variables, sometimes, subject to equality and/or inequality constraints. These problems typically require large-scale real-time linear programming procedures. Most of the time, sequential algorithms as the classical simplex or the interior point methods are implemented. However, these approaches have the disadvantage that the computing time required for a solution is greatly dependent on the problem dimension and structure.

Dynamical systems which can solve real-time optimization were introduced in Pyne (1956) . Since then, other major contributions have been proposed by Korovin and Utkin (1974) , Pazos and Bhaya (2009) , Wang (1993) and Wilson (1986). Due to its inherent massive parallelism, these systems are able to solve optimization problems faster than those using more popular optimization algorithms executed on general-purpose digital computers (Cichocki and Unbehauen, 1993), with great flexibility to parametric variations (Pyne, 1956).

Although the previous studies exhibit high performance, it is necessary to tune several network parameters, that increase linearly with the optimization problem dimension, since for every decision variable there is an individual selection of each activation function. In addition, the fixed time characteristic is not presented in most of the mentioned references.

In this paper, a dynamical system for the solution of linear programming in a predefined convergence time is proposed. Its design is considered as a sliding mode control problem, where the network structure is based on the Karush-Kuhn-Tucker (KKT) optimality conditions (Karush, 1939; Kuhn and Tucker, 1951) and the KKT multipliers are regarded as control inputs. The problem is solved without the individual selection of each stabilizing input, instead, a multivariable function based on the unit control (Utkin, 1992) is used. On the other hand, the fixed time stability (Polyakov, 2012)property ensures system convergence time independent of the initial conditions. This controller is used in the KKT multiplier design, enforcing a sliding mode in which the optimization problem is solved.

The proposed approach has attractive features such as: fixed time convergence to the optimization problem solution and a fixed number of parameters (one in this case), regardless of the optimization problem dimension. Therefore, it offers the scalability characteristic, that

allows the possibility of adding other energy sources to the microgrid.

In the following, Section 2 presents the mathematical preliminaries and some useful definitions. In Section 3 the basis of the linear programming problem are established and the proposed algorithm is presented. Section 4 describes the microgrid connection and the controllers implemented along with the linear programming problem for the power dispatch and shows the simulation results. Finally, in Section 5 the conclusions are presented.

2. MATHEMATICAL PRELIMINARIES

Consider the system

$$\dot{\xi} = f(t, \xi) \tag{1}$$

where $\xi \in \mathbb{R}^n$ and $f : \mathbb{R}_+ \times \mathbb{R}^n \to \mathbb{R}^n$. For this system, its initial conditions or initial states are $\xi(t_0)$ where $t_0 \in \mathbb{R}_+$. The time variable t is defined on the interval $[t_0, \infty)$.

The idea of the sliding mode control is highly related with the finite- time stability. This time however often depends on the initial conditions of the system. The case when convergence time is uniform and independent of the initial conditions is known as fixed time stability Cruz-Zavala et al. (2010). Polyakov (2012) gives a precise statement of the fixed time stability :

Definition 2.1. (Globally fixed-time attraction). Let a non-empty set $M \subset \mathbb{R}^n$. It is said to be globally fixed-time attractive for the system (1) if any solution $\xi(t, \xi_0)$ of (1) reaches M in some finite time moment $t = T(\xi_0)$ and the settling-time function $T(\xi_0) : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\}$ is bounded by some positive number $T_{\max}$, i.e. $T(\xi_0) \leq T_{\max}$ for $\xi_0 \in \mathbb{R}^n$.

Note that there are several choices for $T_{\max}$, for example if $T(x_0) \leq T_{\mathrm{m}}$ for a positive number T_m, also $T(x_0) \leq \lambda T_{\mathrm{m}}$ with $\lambda \geq 1$. This motivates the definition of a set which contains all the bounds of the settling-time function.

Definition 2.2. (Settling-time set). Let the set of all the bounds of the settling-time function for system (1) be defined as follows:

$$\mathcal{T} = \{T_{\max} \in \mathbb{R}_+ : T(x_0) \leq T_{\max}\}. \tag{2}$$

In addition, the minimum bound for the settling-time function of (1) is defined as:

Definition 2.3. (Minimum bound for the settling-time set). Consider the set $\mathcal{T}$ defined in (2), let the time $T_f \in \mathbb{R}$ such that

$$T_f = \{T \in \mathcal{T} : T \leq T_{\max}, \forall T_{\max} \in \mathcal{T}\}. \tag{3}$$

Note that for some systems $T_{\max}$ can be tuned by a particular selection of the system parameters, this notion refers to the prescribed-time stability which is given in Fraguela et al. (2012) and the predefined-time stability Sanchez-Torres et al. (2015). The prescribed-time stability based design and the predefined-time stability based design are explained in the following definitions.

Definition 2.4. (Prescribed-time based design). Consider the set $\mathcal{T}$ defined in (2). The particular case when for the system (1), $T_{\max}$ can be tuned by a particular selection of the system parameters ρ, $T_{\max} = T_{\max}(\rho)$, is referred to the notion of the prescribed-time stability which is given in Fraguela et al. (2012). This design is performed by selecting $T_{\max}(\rho) \in \mathcal{T}$ and calculating the inverse of the settling-time function, allowing the tuning of ρ.

It is worth to note, the true fixed stabilization time for a system designed based on prescribed-time stability is unknown but bounded by $T_{\max}(\rho)$. In contrast, a designed system with predefined-time stability has a known stabilization time.

Definition 2.5. (Predefined-time based design). The particular case when for the system (1), the time T_f defined in (3) can be tuned by a particular selection of the system parameters ρ, $T_f = T_f(\rho)$, is referred to the notion of the predefined-time stability.

With the definition of a predefined-time attractive set, the following lemma provides a Lyapunov characterization of a class of these sets on the state space:

Lemma 2.1. (Lyapunov function Sanchez-Torres et al. (2015)). If there exists a continuous radially unbounded function

$$V : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\}$$

such that $V(x) = 0$ for $x \in M$ and any solution $x(t)$ satisfies

$$\dot{V} \leq -\frac{\alpha}{p} \exp(V^p) V^{1-p} \tag{4}$$

for $\alpha > 0$ and $0 < p \leq 1$, then the set M is globally predefined-time attractive for the system (1) and $T_{\max} = \frac{1}{\alpha} + t_0$.

Proof See Sanchez-Torres et al. (2015).

Definition 2.6. (Predefined-time stabilizing function). For $x \in \mathbb{R}^n$, the predefined-time stabilizing function is defined as

$$\Phi_p(x) = \frac{1}{p} \exp\left(\|x\|^p\right) \frac{x}{\|x\|^p} \tag{5}$$

where $0 < p \leq 1$.

With the definition of the stabilizing function, let the following dynamic system:

Lemma 2.2. (Predefined-time stable dynamical system). For every initial condition x_0, the system

$$\dot{x} = -\frac{1}{T_c} \Phi_p(x) \tag{6}$$

with $T_c > 0$, and $0 < p \leq 1$ is predefined-time stable with settling-time T_c. That is, $x(t) = 0$ for $t > t_0 + T_c$ in spite of the x_0 value.

Proof See Sanchez-Torres et al. (2015).

In order to apply the previous result to control design, consider the dynamic system

$$\dot{\xi} = \Delta(\xi, t) + u \tag{7}$$

with $\xi, u \in \mathbb{R}^n$ and $\Delta : \mathbb{R}_+ \times \mathbb{R}^n \to \mathbb{R}^n$. The main objective is to drive the system (7) to the point $\xi = 0$ in a predefined fixed time in spite of the unknown non-vanishing disturbance $\Delta(\xi, t)$. A solution to this problem which does not require an individual selection of each of the n control variables based on the *unit control* is presented in the following theorem:

Theorem 2.1. (Predefined-time multivariable control). Let the function $\phi(\xi, t)$ to be bounded as $\|\Delta(\xi, t)\| \leq \delta$, with $0 < \delta < \infty$ a known constant. Then, by selecting the control input

$$u = -\left(\frac{1}{T_c} + \delta\right)\frac{\xi}{\|\xi\|}\exp(\|\xi\|)$$

with T_c being a scalar, the system (7) is globally predefined-time stable with settling-time upper bounded by T_c.

Proof: See Sanchez-Torres et al. (2015).

2.1 Linear Programming Problem Statement

Let the following linear programming problem:

$$\begin{cases}\min_x & \mathbf{c}^T x \\ \text{s.t.} & \mathbf{A}x = \mathbf{b} \\ & l \le x \le h\end{cases} \tag{8}$$

where $x = [\, x_1 \ \dots \ x_n \,]^T \in \mathbb{R}^n$ are the decision variables, $\mathbf{c} \in \mathbb{R}^n$ is a cost vector, $\mathbf{A}$ is an $m \times n$ matrix such that $\text{rank}(\mathbf{A}) = m$ and $m \le n$; $\mathbf{b}$ is a vector in $\mathbb{R}^m$ and, $l = [\, l_1 \ \dots \ l_n \,]$, $h = [\, h_1 \ \dots \ h_n \,] \in \mathbb{R}^n$.

Let $y = [\, y_1 \ \dots \ y_m \,]^T \in \mathbb{R}^m$ and $z = [\, z_1 \ \dots \ z_n \,]^T \in \mathbb{R}^n$. Hence, the Lagrangian of (8) is

$$L(x, y, z) = \mathbf{c}^T x + z^T x + y^T(\mathbf{A}x - \mathbf{b}). \tag{9}$$

The KKT conditions establish that x^* is a solution for (8) if and only if x^*, y and z in (8)-(9) are such that

$$\nabla_x L(x^*, y, z) = \mathbf{c} + z + \mathbf{A}^T y = 0 \tag{10}$$

$$\mathbf{A}x^* - \mathbf{b} = 0 \tag{11}$$

$$z_i x_i^* = 0 \text{ if } l_i < x_i^* < h_i, \ \forall i = 1, \dots, n. \tag{12}$$

3. A RECURRENT NEURAL NETWORK (RNN) FOR LINEAR PROGRAMMING PROBLEM

Following the KKT approach, from Loza-Lopez et al. (2015) a recurrent neural network which solves the problem (8) in finite time is proposed. For this purpose, let $\Omega_e = \{x \in \mathbb{R}^n : \mathbf{A}x - \mathbf{b} = 0\}$ and $\Omega_d = \{x \in \mathbb{R}^n : l \le x \le h\}$. According to (8), $x^* \in \Omega$ where $\Omega = \Omega_d \cap \Omega_e$.

From (10), let

$$\dot{x} = -\mathbf{c} + \mathbf{A}^T y + z, \tag{13}$$

then, y and z must be designed such that Ω is an attractive set, fulfilling conditions (10)-(12). For this case, in addition to condition (12), z is considered such that

$$\begin{cases} z_i \ge 0 & \text{if } x_i \ge h_i \\ z_i \le 0 & \text{if } x_i \le l_i \end{cases}, \tag{14}$$

and the variable $\sigma \in \mathbb{R}^m$ is defined as

$$\sigma = \mathbf{A}x - \mathbf{b}. \tag{15}$$

In order to obtain predefined-time stability to the solution x^*, the terms y and z are proposed in (13) as

$$y = (\mathbf{A}\mathbf{A}^T)^{-1}\left[\mathbf{A}\mathbf{c} - \mathbf{A}z + \frac{1}{T_s}\phi(\sigma)\right] \tag{16}$$

and

$$z = \left(\|\mathbf{c}\| + \frac{1}{T_s}\right)\varphi(x, l, h) \tag{17}$$

respectively, where $T_s > 0$.

For this case, the multivariable activation functions are $\varphi(x, l, h) = [\, \varphi_1(x, l_1, h_1) \ \dots \ \varphi_n(x, l_n, h_n) \,]^T$, with $\varphi_i(x, l_i, h_i)$ of the form

$$\varphi_i(x, l_i, h_i) = \begin{cases} -\dfrac{x_i - l_i}{\|x - l\|}\exp(\|x - l\|) & \text{if } x_i \le l_i \\ 0 & \text{if } l_i < x_i < h_i \\ -\dfrac{x_i - h_i}{\|x - h\|}\exp(\|x - h\|) & \text{if } x_i \ge h_i \end{cases} \tag{18}$$

and

$$\phi(\sigma) = -\frac{\sigma}{\|\sigma\|}\exp(\|\sigma\|). \tag{19}$$

Therefore, with the structure given by (13) and the KKT multiplier as in (16) and (17), with activation functions (18) and (19), the following theorem presents a RNN which solves (8) in predefined-time.

Theorem 3.1. (Predefined-time RNN for linear programming). For the RNN

$$\dot{x} = -\mathbf{c}\Lambda + \left(\|\mathbf{c}\| + \frac{1}{T_s}\right)\Lambda\varphi(x, l, h) + \frac{1}{T_s}\mathbf{A}^+\phi(\sigma) \tag{20}$$

where $\Lambda = I - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}$, $\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$ and $T_s > 0$, the point x^* is globally predefined-time stable with settling-time T_s.

Proof: The dynamics of (15) is given by

$$\dot{\sigma} = \mathbf{A}\left(-\mathbf{c} + \mathbf{A}^T y + z\right). \tag{21}$$

Therefore, with the selection of y as in (16), the system (21) reduces to

$$\dot{\sigma} = -\frac{1}{T_s}\frac{\sigma}{\|\sigma\|}\exp(\|\sigma\|).$$

Thus, from *Theorem 2.1*, a sliding mode is induced on the manifold $\sigma = 0$. Therefore, the set Ω_e is predefined-time attractive with settling-time T_s.

On the manifold $\sigma = 0$, the equivalent value of ϕ is the solution of $\dot{\sigma} = 0$. Resulting to $\phi_{\text{eq}} = 0$ or $y_{\text{eq}} = (\mathbf{A}\mathbf{A}^T)^{-1}[\mathbf{A}\mathbf{c} - \mathbf{A}z]$. Therefore, the dynamics of (13) on that manifold is

$$\dot{x} = -\mathbf{c}\Lambda + \Lambda z. \tag{22}$$

With the selection of z as in (17), the resulting system (22) is

$$\dot{x} = -\mathbf{c}\Lambda + \Lambda\left(\|\mathbf{c}\| + \frac{1}{T_s}\right)\varphi(x, l, h).$$

Consider the Lyapunov function $V = \|x\|$. Its derivative is given by $\dot{V} = \frac{x^T}{\|x\|}\dot{x}$. Therefore

$$\begin{aligned}\dot{V} &= \frac{x^T}{\|x\|}\left[-\mathbf{c}\Lambda + \Lambda\left(\|\mathbf{c}\| + \frac{1}{T_s}\right)\varphi(x, l, h)\right] \\ &\le \frac{x^T}{\|x\|}\left[\frac{1}{T_s}\varphi(x, l, h)\right].\end{aligned} \tag{23}$$

Replacing the Lyapunov function

$$\dot{V}\begin{cases} \le -\dfrac{1}{T_s}\exp(V) \text{ if } x < l \text{ or } x > h \\ = 0 \text{ if } l \le x \le h \end{cases} \tag{24}$$

From *Theorem 2.1*, the set Ω_d is predefined-time attractive with settling-time T_s.

In the set Ω_d the equivalent value of φ, φ_{eq}, is the solution to $\dot{x} = 0$. With the application of *Theorem 2.1*, the conditions (11) and (12) are satisfied, providing predefined-time convergence to the set Ω. Now, by using the equivalent control method, the solution of $\dot{x} = 0$ and $\dot{\sigma} = 0$ in (13) for $t > T_s$ has the form

$$\mathbf{c} + \mathbf{A}^T y_{\text{eq}} + z_{\text{eq}} = 0.$$

Therefore, the condition (10) is fulfilled, implying the point $x^* \in \Omega$ is globally predefined-time stable. ■

Remark 3.1. Note that, in contrast to the most of the RNN presented in the literature, this scheme only needs the tuning of one variable, namely T_s in spite of the problem dimensions.

4. OPTIMIZATION OF THE POWER DISPATCH IN A MICROGRID SIMULATION

The optimization algorithm previously presented is applied to the power dispatch problem within a simulated microgrid in order to minimize the consumption of power provided by the utility grid.

4.1 Microgrid Description

The simulated microgrid is connected as in Loza-Lopez et al. (2014), with a *wind power system* (WPS), a connection point with the *utility grid system* (UGS) and a DC voltage bus, which includes the output load, a *solar power system* (SPS), and a *battery bank system* (BBS).

The microgrid simulation is performed in Simulink with the Simscape Power Systems [1] libraries, which include the dynamic simulation of electronic components in order to produce a better approach to a real electrical microgrid. Discrete sliding modes controllers are applied to solar, battery bank and wind power systems.

Wind Power System. The connection and control of the WPS is as in Ruiz et al. (2011) which includes a doubly fed induction generator (DFIG) with a mechanical interconnection to a wind turbine. The DFIG stator is directly connected to the utility grid while the rotor is coupled by a back to back converter. This connection scheme requires two different controllers; one is the grid side controller (GSC), which is in charge of maintaining a fixed voltage in the capacitor, and the second one is the rotor side controller (RSC), which controls the electrical torque T_e according to the mechanical torque T_m in the connection with the wind turbine. Both of these controllers maintain a desired power factor of the energy delivered to the grid.

In Rapheal et al. (2009) the maximum mechanical power for a wind turbine is directly related to a constant value of its tip speed ratio λ :

$$\lambda = \frac{\omega_t R_t}{v}$$

where w_t is the turbine rotor angular speed in rad/s , v is the wind speed m/s and R_t is the wind turbine rotor radius (m). In order to produce the maximum power from the WPS under wind speed variations, λ needs to be transform to an electrical torque (Ruiz et al. (2011)) and be passed as the reference to track by the RSC. In Fig. 1, the performance under wind speed variations of the WPS can be observed, in this case the λ reference to track corresponds to the maximum power point given by the wind turbine block in Simulink.

[1] Simulink/Simscape Power Systems are trademarks of The MathWorks, Inc.

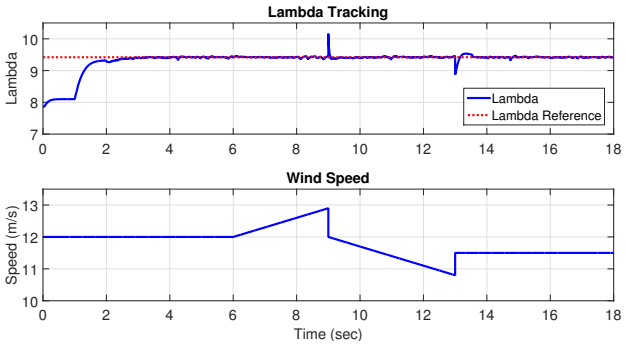


Figure 1. WPS lambda tracking under wind speed variations.

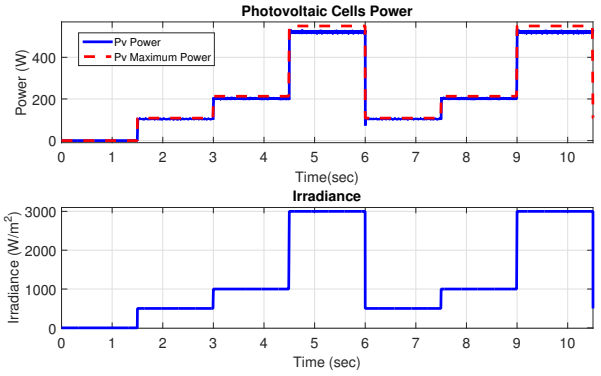


Figure 2. SPS maximum power tracking under irradiance variations.

Solar Power System. The SPS is simulated by a 213 W Simscape Power Systems block with a maximum power point tracker (MPPT) control system, which includes an *perturb and observe* algorithm (de Brito et al. (2013)) to obtain the photovoltaic panel voltage that corresponds to the maximum power $V_{pv_{max}}$, and a DC-DC buck converter in order to track this voltage.

The DC-DC buck converter is directly connected to the output of the SPS as in Koutroulis et al. (2001),this allows the $V_{pv_{max}}$ tracking without the problem of high voltage in the output of the converter. A discrete equivalent control is applied to the insulated-gate bipolar transistor (IGBT) of the DC-DC converter with a pulse width modulation (PWM) block interface. The IGBT switching produces noise in the required measurements for the control system as it would in real time application. In Fig 2, the performance of the SPS along with the MPPT control system under irradiance variations is displayed, the blue line corresponds to the maximum power obtained by the buck converter with the perturb and observe algorithm, and the red dot line is the maximum theoretical power in the photovoltaic panel.

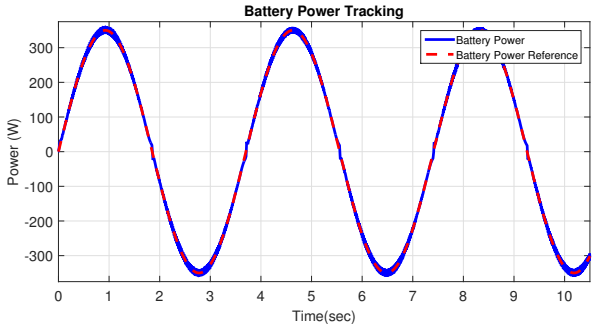


Figure 3. Battery power and reference.

Battery Bank System. The BBS is simulated by a lead-acid 24 V, 100 Ah battery Simulink block. The control system for this BBS is implemented using a parallel buck and boost DC-DC converters; the first one is used under discharge conditions and the second one under charge conditions. A discrete equivalent controller is applied to the each converter, and they switch according to the sign of the reference power to track. In Fig 3, the performance of these controllers acting together is displayed. The blue line is the power given or extracted by the BBS, and the red dot line is a sinusoidal power reference to track.

4.2 Microgrid Optimization Approach

The main goal for this test is to optimize the power dispatch of the microgrid based on the output power of the load at time k (P_{L_k}).

The optimization problem is expressed as follows:

$$\begin{cases} \text{Minimize} & 10P_{G_k} - 500P_{W_k} - 500P_{S_k} - 200P_{B_k} \\ \text{s.t.} & P_{G_k} + P_{W_k} + P_{S_k} + P_{B_k} = P_{L_k} \\ & P_{G_{min}} \leq P_{G_k} \leq P_{G_{max}} \\ & P_{W_{min}} \leq P_{W_k} \leq P_{W_{max}} \\ & P_{S_{min}} \leq P_{S_k} \leq P_{S_{max}} \\ & P_{B_{min}} \leq P_{B_k} \leq P_{B_{max}} \end{cases} \tag{25}$$

In order to match the form of the equation (8), the needed matrices are established as: $\mathbf{c}^T = [10 \ -500 \ -500 \ -200]^T$, $x = [P_{G_k} \ P_{W_k} \ P_{S_k} \ P_{B_k}]^T$, $\mathbf{A} = [1 \ 1 \ 1 \ 1]$, $\mathbf{b} = [P_{L_k}]$, $l = [P_{G_{min}} \ P_{W_{min}} \ P_{S_{min}} \ P_{B_{min}}]$ and $h = [P_{G_{max}} \ P_{W_{max}} \ P_{S_{max}} \ P_{B_{max}}]$. Where $P_{G_k}, P_{W_k}, P_{S_k}$ and P_{B_k} are the UGS, WPS, SPS and BBS powers at time k, and the matrices l and h are their corresponding minimum and maximum power values according to the wind speed in the WPS, the temperature and irradiance in the SPS and the state of charge of the BBS.

4.3 Simulation Results

The presented optimization method uses the measured load power as the vector $\mathbf{b}$ and the matrices defined in the previous section to set the references of power for the microgrid. Due to incentivize the use of the available power given by the SPS, WPS and BBS, the expectation is that the power references for these three systems are set near to their maximum power limits. For this test the settling time T_c is set to $5e^{-5}$ sec to guarantee an appropriate reference

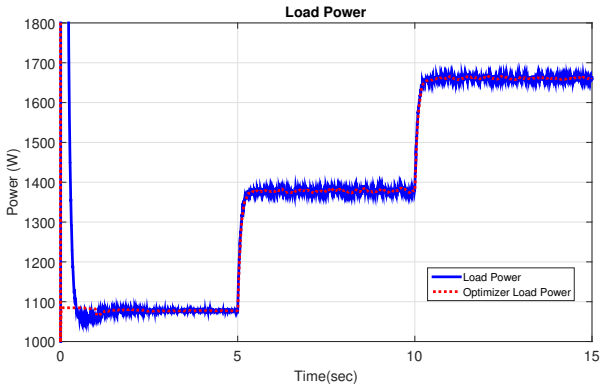


Figure 4. Load power and references sum

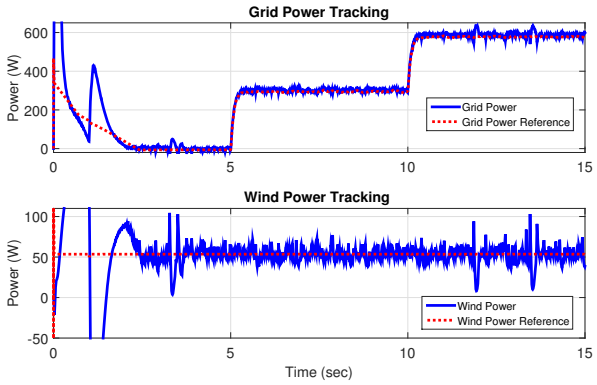


Figure 5. UGS and WPS power tracking

tracking, the wind speed is set to 12 m/s which correspond to a $P_{W_{min}}$ and $P_{W_{max}}$ of 0 and 53.5 W. The temperature and irradiance in the photovoltaic panel are set to 25 °C and 3000 W/m^2 with a $P_{S_{min}}$ and $P_{S_{max}}$ of 0 and 530 W. The state of charge of the battery is 50% which is a neutral state with a discharging and charging maximum powers of −500 and 500 W respectively. Even though the UGS is simulated by an infinite bus, in this test the $P_{G_{min}}$, and $P_{G_{max}}$ are fixed to −1000 and 1000 W . Three different loads are implemented, at the beginning a 0.13 Ω resistor is connected to the DC voltage bus, at 5 seconds a 0.5 Ω is added in parallel and at 10 seconds another 0.5 Ω is connected in the same way.

In Fig 4, the red dot line represents the sum of all the power references given by the optimization method, and the blue line is the measured load power. It can be seen that the equality restriction given in (25) is respected.

In Fig. 5, the UGS and the WPS power dynamics are shown. It can be seen that wind power reference is near to the maximum power limit as expected to the related cost fixed in (25). In the UGS power dynamics it can be seen that when the WPS, BBS and SPS powers are not enough to supply the power of the load, the remaining power is delivered through the UGS. This condition occurs particularly when the two 5Ω parallel resistors are added to the DC bus in 5 and 10 seconds.

In Fig. 6, the blue lines represent the SPS and BBS power and the red dot lines are the optimal power references. It can be seen that the limits established in (25) for these two systems and the reference tracking are fulfilled. The SPS

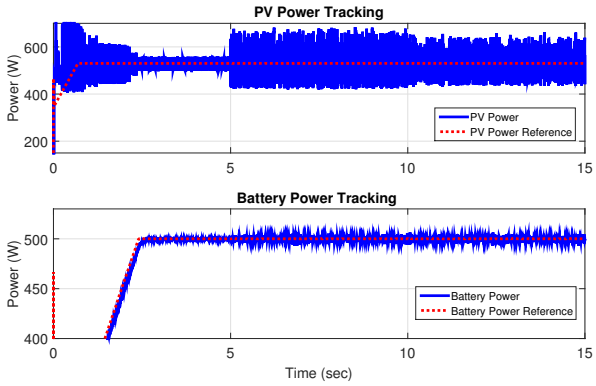


Figure 6. SPS and BBS power tracking.

power reference remains almost all the simulation in the maximum value allowed for this system, this is because of the previously fixed cost related to this system.

5. CONCLUSION

In this paper the optimal power dispatch within a microgrid is found. The microgrid consists of a connection point with the utility grid, a battery bank system, a solar panel system and a wind power system, with appropriate control systems for the last three. A novel recurrent neural network which solves linear programming problem, provides the references to be followed by each controller. The main features of the proposed neural network are predefined convergence time and the tuning of only one parameter. The simulation results validate the use of the presented optimization algorithm. In all simulations, the component dynamics with real parameters were taken into account, which provide a feasible framework for future real-time implementation.

REFERENCES

Aquino, R., Carvalho, M., Neto, O., Lira, M.M.S., de Almeida, G., and Tiburcio, S. (2010). Recurrent neural networks solving a real large scale mid-term scheduling for power plants. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, 1–6.

Chowdhury, S. and Crossley, P. (2009). *Microgrids and Active Distribution Networks*. IET renewable energy series. Institution of Engineering and Technology.

Cichocki, A. and Unbehauen, R. (1993). *Neural networks for optimization and signal processing*. J. Wiley.

Cruz-Zavala, E., Moreno, J., and Fridman, L. (2010). Uniform second-order sliding mode observer for mechanical systems. In *Variable Structure Systems (VSS), 2010 11th International Workshop on*, 14 –19.

de Brito, M.A.G., Galotto, L., Sampaio, L.P., d. A. e Melo, G., and Canesin, C.A. (2013). Evaluation of the main mppt techniques for photovoltaic applications. *IEEE Transactions on Industrial Electronics*, 60(3), 1156–1167.

Fraguela, L., Angulo, M., Moreno, J., and Fridman, L. (2012). Design of a prescribed convergence time uniform robust exact observer in the presence of measurement noise. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 6615–6620.

Karush, W. (1939). *Minima of functions of several variables with inequalities as side constraints*. Master's thesis, Dept. of Mathematics, Univ. of Chicago, Chicago, Illinois.

Korovin, S.K. and Utkin, V.I. (1974). Using sliding modes in static optimization and nonlinear programming. *Automatica*, 10(5), 525 – 532.

Koutroulis, E., Kalaitzakis, K., and Voulgaris, N.C. (2001). Development of a microcontroller-based, photovoltaic maximum power point tracking control system. *IEEE Transactions on Power Electronics*, 16(1), 46–54. doi:10.1109/63.903988.

Kuhn, H.W. and Tucker, A.W. (1951). Nonlinear programming. In *Proc. Second Berkeley Symp. on Math. Statist. and Prob. (Univ. of Calif. Press)*.

Loza-Lopez, M.J., Loukianov, A.G., Sanchez, E.N., Ruiz-Cruz, R., and Snchez-Torres, J.D. (2015). On-line optimization of the power supplied in a microgrid prototype. In *Smart Cities Conference (ISC2), 2015 IEEE First International*, 1–6.

Loza-Lopez, M.J., Sanchez, E.N., and Ruiz-Cruz, R. (2014). Microgrid laboratory prototype. In *Power Systems Conference (PSC), 2014 Clemson University*, 1–5. doi:10.1109/PSC.2014.6808120.

Pazos, F.A. and Bhaya, A. (2009). Control Liapunov function design of neural networks that solve convex optimization and variational inequality problems. *Neurocomputing*, 72(1618), 3863 – 3872.

Polyakov, A. (2012). Nonlinear feedback design for fixed-time stabilization of linear control systems. *IEEE Transactions on Automatic Control*, 57(8), 2106–2110.

Pyne, I.B. (1956). Linear programming on an electronic analogue computer. *American Institute of Electrical Engineers, Part I: Communication and Electronics, Transactions of the*, 75(2), 139–143.

Rapheal, M.S.A., Ram, V.G., Ramachandaramurthy, V.K., and Hew, W.P. (2009). Dynamic response of different wind generator topologies connected to medium size power grid. In *PowerTech, 2009 IEEE Bucharest*, 1–6.

Ruiz, R., Sanchez, E.N., Loukianov, A.G., and Harley, R.G. (2011). Discrete-time block control for a doubly fed induction generator coupled to a wind turbine. In *Proceedings of the IEEE International Conference on Control Applications (CCA)*. Denver, CO, USA.

Sanchez-Torres, J., Sanchez, E., and Loukianov, A. (2015). Predefined-time stability of dynamical systems with sliding modes. In *American Control Conference (ACC), 2015*, 5842–5846.

Utkin, V. (1992). *Sliding modes in control and optimization*. Springer Verlag.

Wang, J. (1993). Analysis and design of a recurrent neural network for linear programming. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 40(9), 613–618.

Wilson, G. (1986). Quadratic programming analogs. *IEEE Transactions on Circuits and Systems*, 33(9), 907–911.

Output-Feedback Model Predictive Control for Dissolved Oxygen Control in a Biological Wastewater Treatment Plant

Maribel Ruiz-Botero*, Manuel Ospina-Alarcon*
Jose Garcia-Tirado**

* *Instituto Tecnológico Metropolitano, Facultad de Ingenierías, Grupo de Automática, Electrónica y Ciencias Computacionales. (e-mail: maribelruiz@itm.edu.co, manuelospina@itm.edu.co).*
** *Instituto Tecnológico Metropolitano, Facultad de Ciencias Económicas y Administrativas, Grupo de Calidad, Metrología y Producción, Calle 73 No 76A - 354, 050034, Medellín - Colombia, (e-mail: josegarcia@itm.edu.co)*

Abstract: The biological wastewater treatment has become a major operation to maintain appropriate levels of organic matter in the wastewater to be discharged. This reduction of organic matter is performed by microorganisms, which require oxygen concentrations adequate to survive. Therefore, the oxygen dissolved control is a critical operation in the biological wastewater treatment plants. In this paper, the design of a coupled estimation and control strategy is presented for a biological wastewater treatment plant. The coupled estimation and control strategy is composed by a Kalman filter and a Model Predictive Controller (MPC). The results obtained showed that for disturbances nearby to the operating point, the estimated state converges to the actual state and the controller maintains the dissolved oxygen levels within a narrow range between 5.3 and 6.7$\frac{mg}{L}$ approximately.

Keywords: Dissolved oxygen, Kalman filter, model predictive control, biological wastewater treatment.

1. INTRODUCTION

Since 1960, terms such as air and water pollution started to be commonly used words. Before that date, these words went unnoticed by the average citizen (Ramalho et al., 1990). Since then mankind has been continuously sensitized about the environment care, with cleaner production processes and rational use of water.

Each industrial process has special requirements regarding to the water quality. To remove such contaminants, the water is subjected to a purification treatment. The effluents of industrial wastewater must also meet minimum conditions to be discharged to the receptor stream. If water effluent does not meet the quality requirements, the stream should be recycled to the treatment plant until these requirements are fulfilled (Lapeña, 1989).

The secondary stage at the biological wastewater treatment refers to all biological processes either aerobics or anaerobic. This process is called activated sludge process and has been used either for the treatment of industrial or urban wastewater for about a century (Ramalho et al., 1990).

Aeration is the process of mixing or dissolving air through a liquid or a substance. This operation of mass transfer is highly important in many industrial environments as well as in the biological wastewater treatment. The industrial and public services applications where aeration systems are found to range from volatile substances removal in liquid currents, subaquatic species culturing for food (industrial aquaculture), wastewater treatment and recombinant proteins design to diverse applications of biological processes where high amounts of enzymes, food, biomedical, and pharmaceutical products are produced.

The control of the dissolved oxygen (DO) concentration is a critical operation for guaranteeing the growth of a diverse group of microorganisms and multicellular organisms. This variable has a non-linear dependence of other variables such as the temperature of the culture medium, pH, biomass concentration, amount of foam among others. The most used procedure to guarantee the oxygenation of microorganisms is by means of aerators, which are basically bioreactors with an ascendant air flow (or pure oxygen flow) from the tank bottom. Main applications of aerators found in the literatures are biological wastewater treatment, general-purpose aerobic cell growth and the culturing of some multicellular organisms (Åmand, 2011; Amicarelli et al., 2010; Atia et al., 2011). In the case of biological treatment of wastewater, there are some aspects that difficult the DO control. First, oxygen mass transfer from gas to liquid phase is considered an activity of high energy consumption (Åmand, 2011). A precise tracking of operation trajectories must be taken into account, in order to avoid the cell death due to oxygen absence or cell stress inhibition by oxygen excess.

Sometimes, there is no inhibition by oxygen excess, but this operation represents cost overruns. In processes where obtaining a metabolite from cell growth, also death and inhibition phenomena are present due to absence or excess of oxygen, respectively. Moreover, production of secondary products (non-desirable products) have been reported, owing to limitations in oxygen mass transfer (Åmand et al., 2013).

Due to of the importance of dissolved oxygen in the biological wastewater treatment plants and the difficulty of monitoring the main process variables a state estimator (Kalman filter) and a model-based control are presented in order to perform an output-feedback loop, as it is shown in Figure 1.

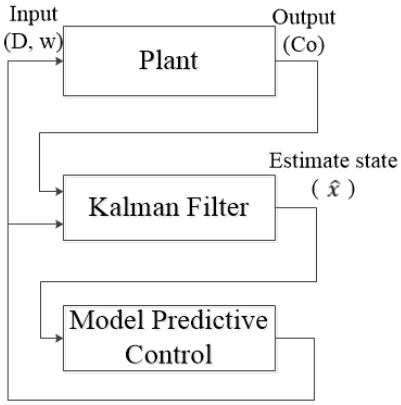


Fig. 1. Estimation and control scheme

The rest of the paper is as follows. In Section 2, materials and methods are presented in this section the mathematical model, state estimator design, and design of model predictive controller are presented. In Section 3 the results and discussion are presented. Final comments are presented in Section 4.

2. MATERIALS AND METHODS

2.1 Dynamic model of a plant for Biological Wastewater Treatment

The biological wastewater treatment is performed by a group of microorganisms called activated sludge, which is responsible for degrading organic matter. In (Nejjari et al., 1999) a model for wastewater treatment plant is presented. The bioprocess is principally constituted by two sequential tanks, an aerator and a settler.

Figure 2 shows the process flow diagram of the biological wastewater treatment plant. This plant consists of an aerated treatment reactor at which the reduction of pollutants is performed from oxidation by the activated sludge. The liquid stream enters the tank, which consists of wastewater and recirculated activated sludge, and air stream, which enters the tank as bubbles through the diffusers. After the treatment with activated sludge, the water enters to a final clarifier, where the activated sludge and the treated water are separated. A part of the activated sludge stream is recycled from the clarifier to the aerated treatment reactor so that the microorganisms content in the reactor is maintained at an equilibrium, and the remainder effluent is discarded as waste sludge (Moltzer, 2008), (Nejjari et al., 1999).

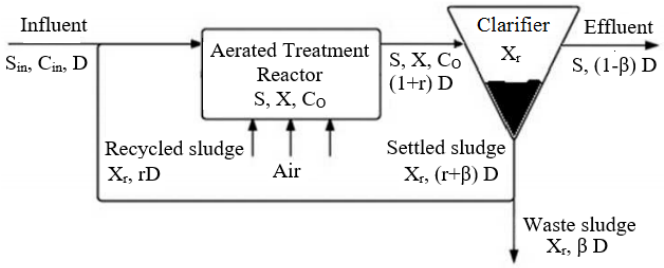


Fig. 2. Process flow diagram for a biological wastewater treatment plant. Adapted from (Moltzer, 2008)

The model is based on the following assumptions:

- The aerated treatment tank is considered to be perfectly mixed so that the concentration of each component is spacially homogeneous.
- Bioreaction does not take place in the clarifier.
- The sludge (biomass) is the only recycled component into the aerated treatment tank.

The model is represented for the Equations (1)-(4). These equations represent the balances of biomass (X), substrate (S), dissolved oxygen (C_o), and recycled biomass (X_r).

$$\frac{dX(t)}{dt} = \mu X(t) - D(t)(1+r)X(t) + rD(t)X_r(t) \quad (1)$$

$$\frac{dS(t)}{dt} = -\frac{\mu}{Y}X(t) - D(t)(1+r)S(t) + D(t)S_{in} \quad (2)$$

$$\frac{dC_o(t)}{dt} = -k_o\frac{\mu}{Y}X(t) - D(t)(1+r)C(t) + D(t)C_{in} + K_La(C_s - C(t)) \quad (3)$$

$$\frac{dX_r(t)}{dt} = D(t)(1+r)X_r(t) - D(t)(\beta+r)X_r(t) \quad (4)$$

$$\mu = \mu_{max}\frac{S(t)}{K_s + S(t)}\frac{C(t)}{K_c + C(t)} \quad (5)$$

In the paper presented by (Nejjari et al., 1999) the dilution rate $D(t)$ and the oxygen transfer coefficient K_La are the manipulated variables. K_La is represented for $K_La = w\alpha$, where the variable parameter is the aeration rate (w) and α is the oxygen transfer rate. The parameters found in (1)-(5) and their assumed numerical values are presented in Table 1.

2.2 Linearized mathematical model

For the design of both the Kalman filter (KF) and the model predictive controller (MPC) a linearized discrete-time model is required. Therefore, in this Section the model linearization is presented. The discrete-time linearized model has the form in (6).

Table 1. Mathematical model parameters and values

Parameter	Name	Value
r	Ratio of recycled flow to influent flow	0.6
Y	Yield of cell mass	0.65
S_{in}	Substrate concentrations in the feed stream	$200\frac{mg}{L}$
K_o	Constant term	0.5
C_s	maximum dissolved oxygen concentration	$10\frac{mg}{L}$
C_{in}	Dissolved oxygen concentrations in the feed stream	$0.5\frac{mg}{L}$
β	Ratio of waste flow to influent flow	0.2
μ_{max}	Maximum specific growth rate	$0.15\frac{mg}{L}$
K_s	Affinity constant	$100\frac{mg}{L}$
K_c	Saturation constant	$2\frac{mg}{L}$
α	Oxygen transfer rate	0.018

$$\begin{aligned} x_{k+1} &= A_d x_k + B_{u_d} u_k + B_{d_d} d_k \\ y_k &= C_d x_k + D_d u_k \end{aligned} \tag{6}$$

where $x_k \in \mathbb{R}^n$ is the state, $u_k \in \mathbb{R}^m$ is the inputs, $w_d \in \mathbb{R}^l$ is the disturbances, $A_d \in \mathbb{R}^{n\times n}$ is the states matrix, $B_{u_d} \in \mathbb{R}^{n\times m}$ is the inputs matrix, $B_{d_d} \in \mathbb{R}^{n\times l}$ is the matrix showing the effect of the disturbance on the system, C_d is the output matrix, and D_d is the input incidence matrix regarding the output.

The linearization of the mathematical model was carried out in the following operating point:

Table 2. Operation point

Parameter	Value
States	
X	$177.69\frac{mg}{L}$
S	$56.66\frac{mg}{L}$
C_o	$5.61\frac{mg}{L}$
X_r	$355.38\frac{mg}{L}$
Inputs	
D	$0.1h^{-1}$
w	$80h^{-1}$
Disturbances	
S_{in}	$200\frac{mg}{L}$

The matrices A_d, B_{u_d}, B_{w_d}, C_d, and D_d of the linearized discrete model were obtained with a sampling time of $0.01h$ as

$$A_d = \begin{bmatrix} 0.9988 & 0.0008 & 0.0033 & 0.0006 \\ -0.0006 & 0.9972 & -0.0051 & -0.0000 \\ -0.0003 & -0.0006 & 0.9816 & -0.0000 \\ 0.0016 & 0.0000 & 0.0000 & 0.9992 \end{bmatrix}$$

$$B_{u_d} = \begin{bmatrix} -0.7100 & 0.0000 \\ 1.0924 & -0.0000 \\ -0.0843 & 0.0008 \\ -0.0006 & 0.0000 \end{bmatrix}$$

$$B_{w_d} = 1 \times 10^{-3} \begin{bmatrix} 0.0004 \\ 0.9986 \\ -0.0003 \\ 0.0000 \end{bmatrix}$$

$$C_d = [0\ 0\ 1\ 0]$$

$$D_d = [0\ 0]$$

The biomass and the substrate in biological wastewater treatment plants are difficult to monitor online, because the sensors available in the industry are expensive or nonexistent, so the measurements of these variables are performed offline. For this reason, the dissolved oxygen is considered as the only measurement variable online in biological wastewater treatment plants.

2.3 Kalman filter desing

The Kalman filter operates by propagating the mean and covariance of the state the over time. This filter is composed by three parts: the linear dynamic system equations, the initial condition of the *a priori* covariance of the estimation error, the Kalman filter gain, and the update equations (Simon, 2006).

Dynamic system equations

$$x_{k-1} = f_k(x_k, u_k, w_k) \tag{7}$$
$$x_k = h_k(x_k, v_k) \tag{8}$$
$$w_k \sim (0, Q_k) \tag{9}$$
$$v_k \sim (0, R_k) \tag{10}$$

Kalman filter initialization

$$P_k{}^- = A_{d_{k-1}} P_k{}^+ A_{d_{k-1}}^T + Q_{k-1} \tag{11}$$
$$K_k = P_k{}^+ C_{d_k}^T R_k^{-1} \tag{12}$$

Kalman filter equations

$$\hat{x}_k^- = A_{d_{k-1}} \hat{x}_{k-1}^+ B_{u_{d_{k-1}}} u_{k-1} \tag{13}$$
$$\hat{x}_k^+ = \hat{x}_k^- K_k (y_k - C_{d_k} \hat{x}_k^-) \tag{14}$$
$$P_k{}^+ = (I - K_K C_{d_k}) P_k{}^- \tag{15}$$

where x_k^- is the *a priori* estimate, P_k^- is the *a priori* covariance of the estimation error, x_k^+ is the *a posteriori* estimate, P_k^+ is the *a posteriori* covariance of the estimation error, K_k is the Kalman gain, $A_{d_{k-1}}$ is the states matrix, $B_{u_{d_{k-1}}}$ is the inputs matrix, Q_{k-1} is the model uncertainty, R_{k-1} is the measurement noise, C_{d_k} is the output matrix and y_k is the measurement.

Initial conditions and filter tuning

The Kalman filter design was performed using the linearized mathematical model presented in Subsection 2.2, adding the disturbance as a state in order to be estimated. The new mathematical model form is then (Davison and Smith, 1971):

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} A & B_{d_d} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ d_k \end{bmatrix} + \begin{bmatrix} B_{u_d} \\ 0 \end{bmatrix} u_k \\ y_k &= [C_d\ 0] \begin{bmatrix} x_k \\ d_k \end{bmatrix} + D_d u_k \end{aligned} \tag{16}$$

It turns out then

$$A_{d_{k-1}} = \begin{bmatrix} A_d & B_{d_d} \\ 0 & 1 \end{bmatrix},\ C_{d_k} = [C_d\ 0],\ B_{u_{d_{k-1}}} = \begin{bmatrix} B_{u_d} \\ 0 \end{bmatrix}$$

The tuning matrices associated with the model uncertainty and the measurement noise, Q_{k-1} and R_k, respectively, were set as

$$R_k = 5 \tag{17}$$

$$Q_{k-1} = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 5 \end{bmatrix} \tag{18}$$

The *a posteriori* covariance (P_k^+) was initialized as

$$P_0^+ = \begin{bmatrix} 60 & 0 & 0 & 0 & 0 \\ 0 & 60 & 0 & 0 & 0 \\ 0 & 0 & 60 & 0 & 0 \\ 0 & 0 & 0 & 60 & 0 \\ 0 & 0 & 0 & 0 & 60 \end{bmatrix} \tag{19}$$

In the simulation, the measurement noise is considered a white noise with zero mean and 0.01 variance.

2.4 Model predictive control design

The main elements of the model predictive control (MPC) are the prediction model, objective function, constrains, and receding horizon principle.

Prediction model

The prediction model selected for the design of the MPC is the linearized discrete model presented in the Subsection 2.2, whose structure is of the form:

$$\begin{aligned} x_{k+1} &= A_d x_k + B_{u_d} u_k + B_{w_d} d_k \\ y_k &= C_d x_k + D_d u_k \end{aligned}$$

Objective function

Cost function used for the MPC design is the used in the linear quadratic predictive control (LQPC) (Garcia et al., 1989). The LQPC structure is presented in (20).

$$J(\Delta\tilde{u}, x_0) = \hat{x}_{N_p}^T P_c \hat{x}_{N_p} + \sum_{k=0}^{N_p} \hat{x}_k^T Q_c \hat{x}_k + \sum_{k=0}^{N_c} \Delta\tilde{u}_k^T R_c \Delta\tilde{u} \tag{20}$$

where J is the cost function, x_k is the state, u_k is the control signal, N_p is the prediction horizon, N_c is the control horizon, P_c is the weight matrix of the control signal, Q_c is the weight matrix of the states, and R_c is the weight matrix of the terminal state.

The optimization of the objective function was performed by using the optimization tool box MATLAB® by means of the *quadprog* command.

Constrains

The MPC design considered the following constraints on the manipulated variables:

$$\begin{aligned} 0.02h^{-1} \leq & D \leq 0.15h^{-1} \\ 0h^{-1} \leq & w \leq 300h^{-1} \end{aligned} \tag{21}$$

State constraints were not implemented since the state remained into safe and realistic values in the performed simulations. Nevertheless, state constraints may be included at any time with no major effort.

Initial conditions and controller tuning

The MPC design used a control horizon $N_c = 30$, and a prediction horizon $N_p = 60$.

Te weight matrices for the terminal state, state and inputs were set as follows.

$$P_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{22}$$

$$Q_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{23}$$

$$R_c = \begin{bmatrix} 1 & 0 \\ 0 & 0.1 \end{bmatrix} \tag{24}$$

The biggest weight was assigned to the dissolved oxygen state because the MPC purpose is to control the dissolved oxygen in the aerated treatment reactor.

3. RESULTS AND DISCUSSION

The results obtained by applying the controller and the estimator are presented below. The behavior of the MPC and the state estimator was evaluated under different disturbances. The considered disturbances were $250\frac{mg}{L}$, $300\frac{mg}{L}$, and $150\frac{mg}{L}$ of the substrate concentration in the feed stream. Simulations of the MPC coupled with state estimator were developed in the MATLAB® software version 2016a, using the Runge-Kutta numerical method with a fixed step of $0.01h$ during $500h$.

Figures 3 and 4 shows the system response with a disturbance of $250\frac{mg}{L}$ at the time $50h$. The former Figure present the response of the state to the disturbance and and the latter Figure presents the control actions and the disturbance estimation. In these Figures, it can be seen that the state estimator achieves the real state of the plant and the disturbance. Furthermore, it can be seen that the plant state change in the open loop regarding the status of the controlled plant. For the state X the steady state value of open loop plant is 21% superior to the state of the closed-loop plant. For the state S, the steady state value is 19% less than the state of the controlled plant. Similarly, the state X_r represents a steady state value of 14% higher with respect to the steady-state closed loop. The dissolved oxygen state presents oscillatory behavior because the rate of air intake to the treatment tank (w) presents sudden

changes between upper and lower restriction, as occurs with the dilution rate (D).

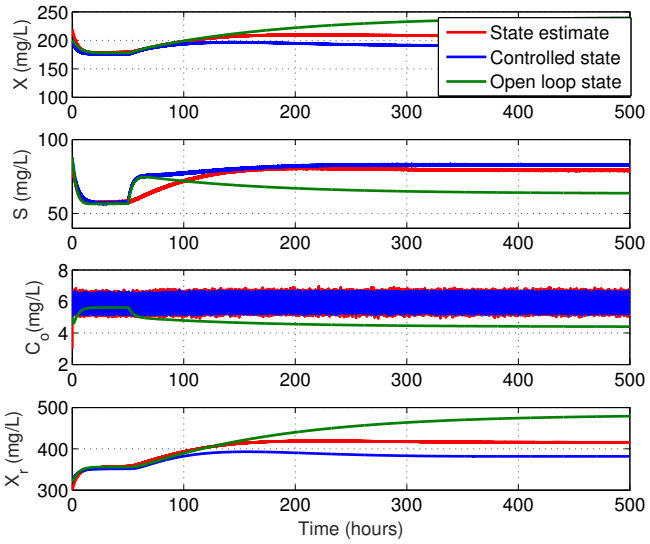


Fig. 3. State behavior with a disturbance of $250\frac{mg}{L}$ in the substrate concentration in the feed stream at the time $50h$

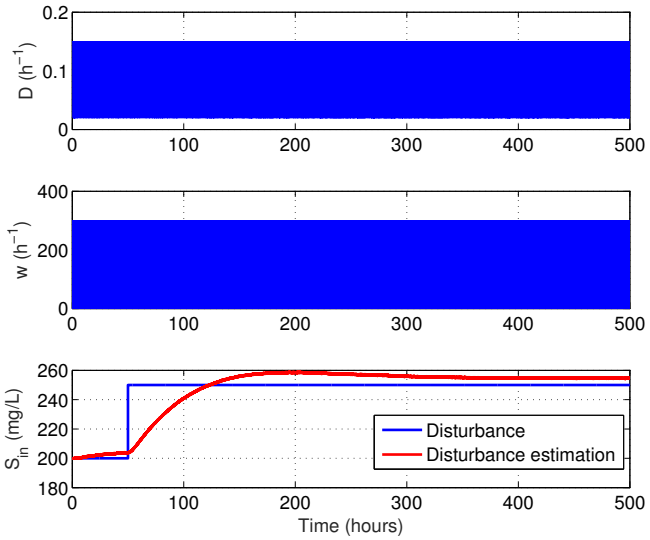


Fig. 4. Control actions and estimation disturbance $(250\frac{mg}{L})$

The behaviors of the states with the different disturbances applied are similar (Figures 5 - 8). The biomass in the reactor and in the recycle stream in open-loop present a steady state value above than the steady state value of the closed-loop system. The substrate state presented a steady state value in open-loop below than the presented by the controlled system. The behavior of the state substrate in the closed-loop is due to that is presents lower production of biomass and recycled biomass, so that there is an insufficient concentration of microorganisms to be responsible for degrading the substrate.

Figures 5 and 7 shown the state behavior with a disturbance of $300\frac{mg}{L}$ and $150\frac{mg}{L}$, respectively. The first figure presents the biomass behavior, the second figure the substrate concentration behavior, the third figure the dissolved oxygen concentration behavior, and the final figure the recycled biomass concentration. Figures 6 and 8 present the control actions and the disturbance estimation for the plant with the aforementioned disturbances.

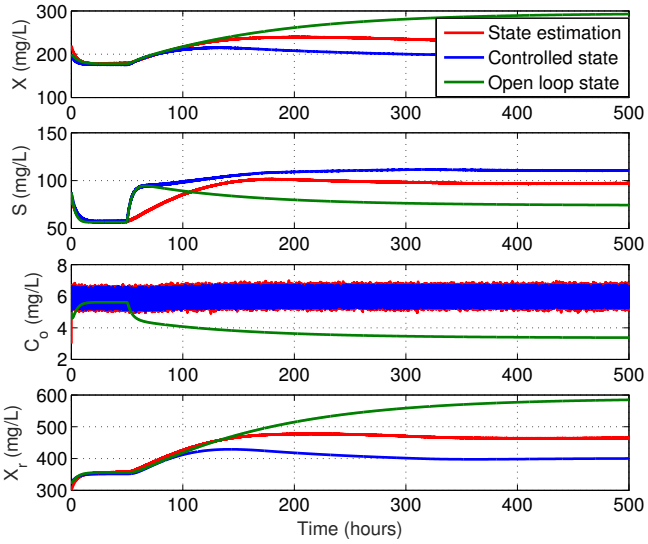


Fig. 5. State behavior with a disturbance of $300\frac{mg}{L}$ in the substrate concentration in the feed stream at the time $50h$

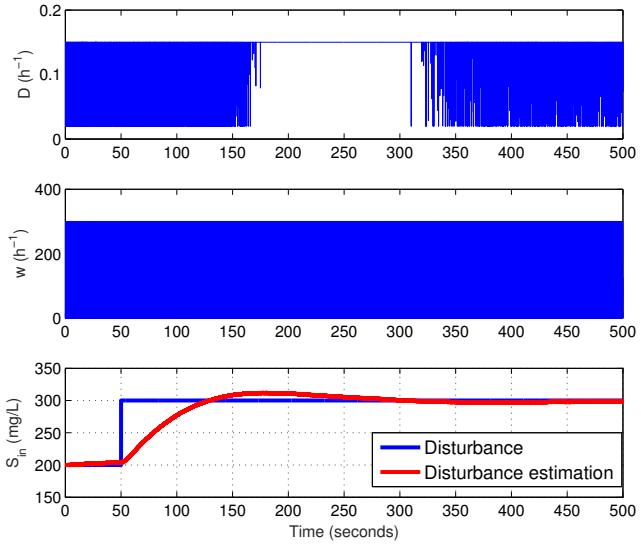


Fig. 6. Control actions and estimation disturbance $(300\frac{mg}{L})$

In Figures 7 and 8, the estimator converges faster if compared to previous scenarios.

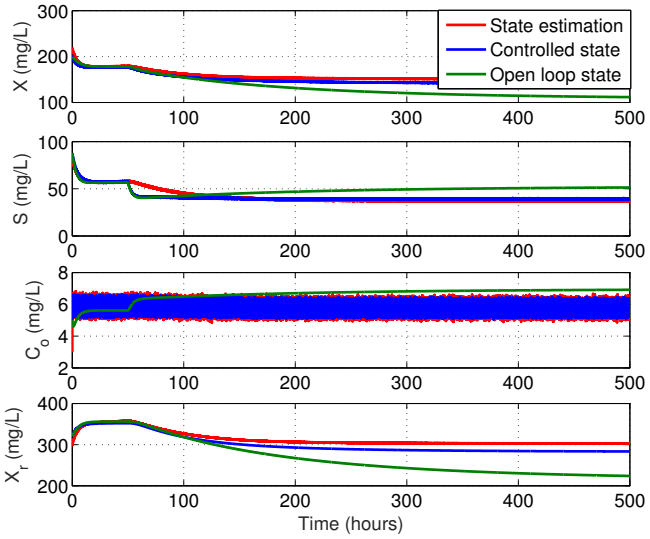


Fig. 7. State behavior with a disturbance of $150\frac{mg}{L}$ in the substrate concentration in the feed stream at the time $50h$

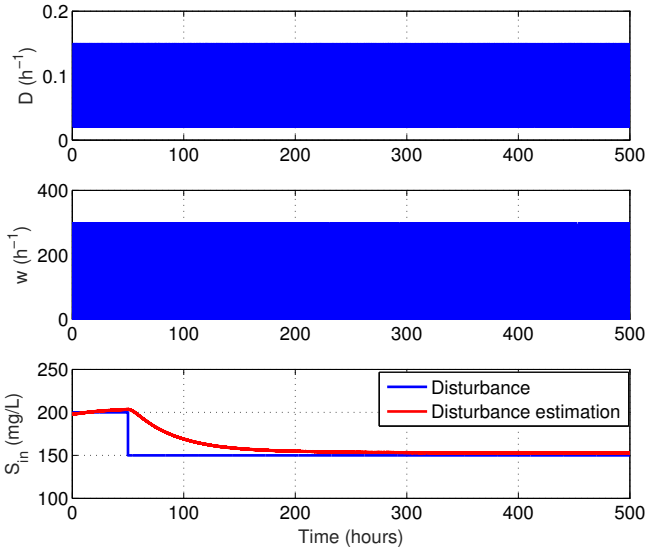


Fig. 8. Control actions and estimation disturbance $(150\frac{mg}{L})$

FINAL COMMENTS

According to the results, it can be concluded that the joint estimation strategy and control are suitable for controlling dissolved oxygen in biological treatment plants wastewater, because although oxygen has an oscillatory behavior was maintained between 5.3 and $6.7\frac{L}{mg}$ approximately. Given that the operating point for dissolved oxygen state is $5.61\frac{mg}{L}$, it can be said that the controller kept the concentration of dissolved oxygen at desired levels.

Due to the nonlinearity of the system as future work the design of a joint nonlinear strategy of estimation and control will be proposed, in order to obtain better performances both the estimator and the controller, and check the behavior of dissolved oxygen state it can be improved with this new strategy. Additionally, this new strategy could improve the oscillations in the control actions, causing these have a softer behavior.

REFERENCES

Åmand, L. (2011). Control of aeration systems in activated sludge processes–a review. *IVL Swedish Environmental Research Institute/Department of Information Technology, Uppsala University, Uppsala, Sweden.*

Åmand, L., Olsson, G., and Carlsson, B. (2013). Aeration control - A review. *Water Science and Technology*, 67, 2374–2398. doi:10.2166/wst.2013.139.

Amicarelli, A., Di Sciascio, F., Toibero, J.M., and Alvarez, H. (2010). Including dissolved oxygen dynamics into the Bt δ -Endotoxins Production process model and its application to process control. *Brazilian Journal of Chemical Engineering*, 27(01), 41–62.

Atia, D., Fahmy, F., Ahmed, N., and Dorrah, H. (2011). Design and Control Strategy of Diffused Air Aeration System. *World Academy of Science, Engineering and Technology*, (3), 666–670.

Davison, E. and Smith, H. (1971). Pole assignment in linear time-invariant multivariable systems with constant disturbances. *Automatica*, 7(4), 489–498.

Garcia, C.E., Prett, D.M., and Morari, M. (1989). Model predictive control: theory and practice—a survey. *Automatica*, 25(3), 335–348.

Lapeña, M.R. (1989). *Tratamiento de aguas industriales: aguas de proceso y residuales*, volume 27. Marcombo.

Moltzer, M. (2008). *Analysis of Robust Stability of Model Predictive Control for Biological Wastewater Treatment Plants.* Tesis de maestría, Eindhoven University of Technology, Eindhoven, Holanda.

Nejjari, F., Benhammou, a., Dahhou, B., and Roux, G. (1999). Non-linear multivariable adaptive control of an activated sludge wastewater treatment process. *International Journal of Adaptive Control and Signal Processing*, 13(May 1998), 347–365.

Ramalho, R.S., Beltrán, D.J., and de Lora, F. (1990). *Tratamiento de aguas residuales.* Reverté.

Simon, D. (2006). *Optimal state estimation: Kalman, H infinity, and nonlinear approaches.* John Wiley & Sons.

Output-Feedback Model Predictive Control for Tight Glycaemic Control in Patients at the Intensive Care Unit

E. Aguirre-Zapata * **and J. Garcia-Tirado** **

* *Instituto Tecnológico Metropolitano, Calle 73 No 76A - 354, 050034, Medellín - Colombia. Facultad de Ingenierías, Grupo de Automática, Electrónica y Ciencias Computacionales. (email: estefaniaaguirre97289@correo.itm.edu.co).*
** *Instituto Tecnológico Metropolitano, Calle 73 No 76A - 354, 050034, Medellín - Colombia. Facultad de Ciencias Económicas y Administrativas, Grupo de Calidad, Metrología y Producción, (email: josegarcia@itm.edu.co).*

Abstract: the glucose homoeostasis is responsible for regulating the blood glucose concentration to stay around of 5.56 *mmol/L*. However, this regulatory mechanism may be affected by Diabetes Mellitus (Type 1, Type 2, and gestational) or as a side effect of a critical condition especially in patients who are at an intensive care unit (ICU). From the control engineering point of view, there is an approach to deal with pathological conditions leading to the glucose impairment and is based on the automatic control of insulin infusion in order to avoid dangerous conditions as hyperglycaemia and hypoglycaemia. This paper describes the design and simulation of a model predictive controller combined with an Extended Kalman filter for tight glycemic control of patients at the ICU. From the simulations performed, the controller is able to deal with the enteral feeding delivered to the patient, which is an unavoidable disturbance in a realistic scenario. The paper concludes that care must be paid to perform the controller tuning, especially the matrix related to the state penalty. Finally, future would be directed to the design of nonlinear model predictive controllers in order to obtain better controller performance.

Keywords: Model predictive control, Intensive care patients, Extended Kalman filter, output-feedback, tight glycemic control, Diabetes Mellitus.

1. INTRODUCTION

In healthy subjects, the glucose homeostasis is responsible for guaranteeing the healthy levels of blood glucose. This natural regulation is performed by means of neuro-hormonal compensations, acting selectively on excess and lack of carbohydrates (Barrett, 2013). In this control loop, the pancreas acts as an actuator (releases insulin and glucagon) and some elements present in the blood are as sensors carrying information about the blood glucose concentration levels to the liver and pancreas. However, this control loop may be broken due to a deficiency in the production of the insulin hormone (Type 1 Diabetes Mellitus), resistance to the action of insulin (Type 2 Diabetes Mellitus) or as side effect of other pathologies, especially in patients who are at the ICU (Aldworth et al., 2015), (Bequette, 2007).

Patients at the ICU may present serious episodes of hyperglycemia in response to stress after severe trauma. The organism of the ill triggers a series of neuro-hormonal stimuli that in the acute phase of illness increases glucose production and decreases its uptake by the muscle, resulting in a significant increase in the blood glucose levels (Villamarín and Puentes, 2009). Studies have shown that hyperglycemia episodes increased the hospital stay, mortality rates, and the probability of patient follow up once they leave the ICU (Umpierrez et al., 2002). The above motivated the research in tight glycaemic control at the ICU by manipulating a continuous insulin infusion. This led to a decrease in the mortality rate regarding to the malfunction of the glucose homeostasis (Van den Berghe et al., 2006). Existing approaches of tight glycaemic control using controllers based on modified versions of the Bergma's minimal model use the plasma glucose as the measured variable. Although plasma glucose is the controlled variable and hence the main variable, the measurement of the plasma glucose, even at the ICU, is invasive, expensive, and in most ICU facilities not continuously available. In this sense, alternative measurement approaches as from a continuous glucose monitor (CGM) should be taken into account (Block et al., 2008).

Using a CGM at the ICU allow tighter control of glucose levels in blood given that the availability of continuous measurements of interstitial glucose every 5 min for $24h/day$ would free the patient of the invasive tests and would represent a decrease in the costs of treatment and monitoring of hyperglycemia for the health system (Block et al., 2008).

Model Predictive Control (MPC) became an appealing alternative to perform glycaemic control and mainly to avoid hypoglycemic episodes. MPC includes a process model in order to predict the future process behavior using an optimization framework, allowing a direct addressing of process constraints.

A coupled estimation-control strategy is tested in a scenario of tight glycaemic control at the ICU. To perform the design of the estimator and controller, a modified version of the Bergman's minimal model is used as it is presented in (Lin et al., 2011) together with an additional equation relating the interstitial glucose with the plasma glucose and hence allowing the use of a CGM (King et al., 2007).

The remainder of the paper is as follows. In Section 1, the problem is stated. In Section 2, the equations of the mathematical model for controlling hyperglycemia in ICU patients is presented. A new equation is added in order to allow the measurement of the interstitial glucose instead of plasma glucose. The output feedback strategy using the coupling of a MPC and an Extended Kalman Filter is presented in Section 3. Results are presented in Section 4. Conclusions and final comments are addressed in Section 5.

2. PROBLEM STATEMENT

Nowadays, the control of blood glucose levels in patients at the ICU in Colombian hospitals is performed by taking blood samples for subsequent evaluation of plasma glucose levels. The samples are analyzed using an analytical procedure in the laboratory causing delays for the decision making. Following the protocol of Colombian hospitals, blood glucose must be monitored every 2h at the ICU which leads to 12 laboratory assays every day at around US 40 (120.000 COP) (Hospital Universitario de Bucaramanga S.A, 2012).

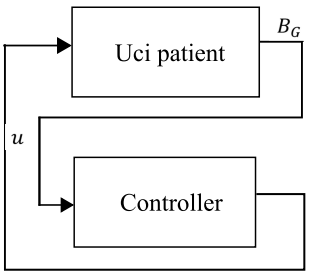


Fig. 1. Normal control of glucose levels in ICU patients

In Figure 1, the typical glycaemic control scheme, from measuring plasma glucose is presented (Magni et al., 2007) where B_G and u stand for the blood glucose and exogenous insulin input. In this paper, an output-feedback controller based on an Extended Kalman Filter (EKF) coupled to a MPC is proposed as shown in Figure 2 where I_G, the interstitial glucose, is the measured variable instead of B_G and x_{ext} is the estimated state.

The interstitial glucose monitoring using a CGM represents the possibility of performing tight glycemic control in patients at the ICU with a reduction of hypoglycemic episodes and a decrease in the cost of monitoring for the local health system.

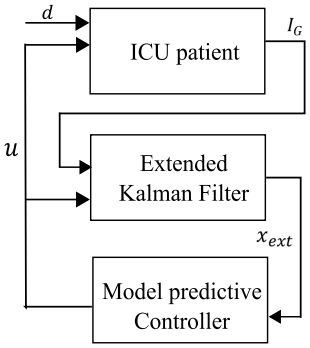


Fig. 2. Control of glucose levels in ICU patients proposed

3. MATHEMATICAL MODEL FOR TIGHT GLYCAEMIC CONTROL

A model for tight glycaemic control, based on the Bergman's minimal model is available in (Lin et al., 2011). From the equations, it is possible to identify four main compartments, i.e., blood, interstitium, stomach, and intestine, where the processes of metabolism, distribution, and use of insulin and glucose occur. An abstraction of the compartments involved in the extended mathematical model and the dynamic relationships between the various terms is presented for the sake of comprehension in Figure 3 where continuous arrows represent mass transfer among compartments and dotted arrows represent the diffusion from the blood compartments to the interstitium.

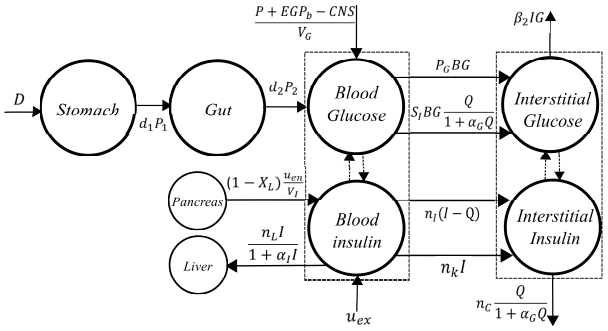


Fig. 3. Model Compartments including the Interstitium

The mathematical model under study includes five states: blood glucose (BG), insulin concentration in the interstice (Q), insulin in plasma (I), the amount of glucose in the stomach (P_1) where carbohydrates are broken down in smaller molecules, and the amount of glucose in the gut (P_2) where glucose molecules are absorbed and transported into the bloodstream. The equations of the mathematical model are summarized as follows:

$$\frac{dB_G}{dt} = -P_G B_G - S_i \frac{B_G Q}{1+\alpha_G Q} + \frac{P + EG_{P_b} - CNS}{V_G} \quad (1)$$

$$\frac{dQ}{dt} = n_I(I - Q) - n_C \frac{Q}{1+\alpha_G Q} \quad (2)$$

$$\frac{dI}{dt} = -n_K I - n_L \frac{I}{1+a_I I} - n_I(I - Q) + \frac{u_{ex}}{V_I} + (1 - x_L)\frac{u_{en}}{V_I} \quad (3)$$

$$\frac{dP_1}{dt} = -d_1 P_1 + D; \quad (4)$$

$$\frac{dP_2}{dt} = -\min(d_2 P_2, P_{max}) + d_1 P_1; \quad (5)$$

where $P(t)$, as it is defined in (6), corresponds to the glucose appearance into the bloodstream from the enteral nutrition and u_{en} is the endogenous insulin production.

$$P(t) = min(d_2 P_2, P_{max}) + PN \quad (6)$$

$$u_{en} = k_1 e^{-I(t)\frac{k_2}{k_3}} \quad (7)$$

The model proposed by Lin was developed for the control of blood glucose levels by administering exogenous glucose and insulin to patients at the ICU. However, although this model allows to establish the existing relationship between glucose and insulin, it does not give a way to include the interstitial glucose to relate the measurements from a CGM to the plasma glucose. In order to include the use of a CGM at the ICU, an equation relating the interstitial glucose and plasma glucose is added as presented in (8) (King et al., 2007).

$$\frac{dI_G}{dt} = \beta_1 B_G - \beta_2 I_G \quad (8)$$

A description of the model variables and parameters together with its units and numerical values are presented in the Appendix A.

4. STATE ESTIMATION AND MODEL BASED CONTROL

4.1 ***Extended Kalman Filter design***

The Extended Kalman Filter (EKF) is a variation of the Kalman filter addressing nonlinear systems with smooth nonlinearities (Pascual, 2004). The EKF derivation is based on the linearization of the nonlinear system around a nominal trajectory state and was originally proposed by Stanley Schmidt to allow the use of the Kalman filter in a nonlinear systems setting (Simon, 2006) (Schmidt, 1966).

The discrete-time EKF is composed by four parts (Simon, 2006): the equations of the nonlinear dynamic system (9)-(12), the initialization to the *a priori* estimation error covariance and the *a priori* estimated state (13) and (14), the linearization point to point (15)-(16) and (19)-(20), and the update equations (17)-(18) and (21)-(23).

Mathematical formulation

(1) Dynamic system equations:

$$x_{k-1} = F_k(x_k, u_k, w_k) \quad (9)$$

$$y_k = h_k(x_k, v_k) \quad (10)$$

$$w_k \sim N(0, Q_{e,k}) \quad (11)$$

$$v_k \sim N(0, R_{e,k}) \quad (12)$$

(2) EKF initialization:

$$\hat{x}_0^+ = E(x_0) \quad (13)$$

$$P_{e,0}^+ = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] \quad (14)$$

(3) for $k = 1, 2...n$

(a) Calculation of matrices of partial derivatives:

$$F_{k-1} = \frac{\partial f_{k-1}}{\partial x}|_{\hat{x}_{k-1}^+, u_k} \quad (15)$$

$$L_{k-1} = \frac{\partial f_{k-1}}{\partial w}|_{\hat{x}_{k-1}^+, u_k} \quad (16)$$

(b) A priori state estimate and estimation error covariance:

$$P_{e,k}^- = F_{k-1} P_{e,k-1}^+ F_{k-1}^T + Q_{e,k-1} \quad (17)$$

$$\hat{x}_k^- = f_{k-1}(\hat{x}_{k-1}^+, u_{k-1}) \quad (18)$$

(c) Calculation of matrices of partial derivatives:

$$H_k = \frac{\partial h_k}{\partial x}|_{\hat{x}_k^-, u_k} \quad (19)$$

$$M_k = \frac{\partial h_k}{\partial v}|_{\hat{x}_k^-, u_k} \quad (20)$$

(d) Updating the state estimation and estimation error covariance:

$$K_k = P_{e,k}^- H_k^T (H_k P_{e,k}^- H_k^T + R_{e,k})^{-1} \quad (21)$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k[y_k - h_k(\hat{x}_k^-)] \quad (22)$$

$$P_{e,k}^+ = (I - K_k H_k) P_{e,k}^- \quad (23)$$

where $x_k \in \mathbb{R}^n$ is the system state, $\hat{x}_0^+$ is the initial condition state, $P_{e,0}^+$ is the initial condition for the a *posteriori* estimation error covariance, $P_{e,k}^-$ is *a priori* estimation error covariance, K_k is the Kalman gain, $\hat{x}_k^-$ *a priori* estimated state, $\hat{x}_k^+$ *a posteriori* estimated state, $P_{e,k}^+$ is the *a posteriori* estimation error covariance. $R_{e,k}$ and $Q_{e,k}$ are the covariances of the model and measurement noises which in turn are knobs of the filter together with $\hat{x}_0^+$ and $P_{e,0}^+$.

Initial conditions and filter tuning: as pointed out before, in a real-life application the enteral feed is the unexpected disturbance entering to the process. Considering this disturbance, an enlarged state is defined as

$$x_k = [B_{G,k}\ I_{G,k}\ Q_k\ I_k\ P_{1,k}\ P_{2,k}\ D_k]^T \quad (24)$$

where the states are defined in discrete-time with a given sample time. Given the operating point of the glucose-insulin system, the EKF was initialized as follows, where variables are with proper units

$$\hat{x}_0 = [5\ 5\ 10.7655\ 20\ 22\ 111\ 0.7672]^T$$

The filter tuning was performed by a trial-and-error procedure by means of the mean quadratic error as performance

criterion. The weighting matrices allowing the most acceptable filter performance are presented below:

$$Q_{e,k} = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix} \tag{25}$$

$$R_{e,k} = 1 \tag{26}$$

$$P^{+}_{e,0} = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix} \tag{27}$$

4.2 ***Model Predictive Control design***

The design of the MPC was made taking into account the elements presented in (Van den Boom and Backx, 2010). The design considerations are presented as follows.

Prediction model: the mathematical model presented in Section 3 was linearized around the following operating point

$$x_{ss} = [5 \ 5 \ 10.7655 \ 20 \ 22 \ 111]^T \tag{28}$$

$$u_{ss} = [7.5933 \ 0]^T \tag{29}$$

where x_{ss} corresponds to the operating point of states and u_{ss} corresponds to the operating point of inputs. Then, the linearized model is discretized with a sample time of $T_s = 1min$. The linear discrete-time model of glucose-insulin system at the ICU is written as

$$\begin{aligned} x_{k-1} &= F_k x_k + G_k u_k + B_d d \\ y_k &= H_k x_k \end{aligned} \tag{30}$$

with F_k the state matrix, G_k is the input matrix, B_d is the matrix relating the unknown disturbance to the model, and H_k is the output matrix. The numerical values of the latter matrices are

$$F_k = \begin{bmatrix} 0.9758 & 0 & -0.007 & -0.00 & 0 & 0.0005 \\ 0.009 & 0.990 & -0.00 & -0.00 & 0 & 0 \\ 0 & 0 & 0.994 & 0.002 & 0 & 0 \\ 0 & 0 & 0.002 & 0.814 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.965 & 0 \\ 0 & 0 & 0 & 0 & 0.034 & 0.993 \end{bmatrix} \tag{31}$$

$$G_k = \begin{bmatrix} -0.0000 \\ -0.0000 \\ 0.0004 \\ 0.2870 \\ 0 \\ 0 \end{bmatrix} \tag{32}$$

$$Bd = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0.9828 \\ 0.0171 \end{bmatrix} \tag{33}$$

$$H_k = [0 \ 1 \ 0 \ 0 \ 0 \ 0] \tag{34}$$

The reader is advised about the notation abuse since x_k is used for both the nonlinear and linear models. In this sense, it worth to point out that the notation depends on the context. For instance, x_k in the linear model stands for the state in deviation variables.

Objective function: in the literature of MPC three cost functions appear typically, i.e., GPC (Generalized Predictive Control), LQPC (Linear Quadratic Predictive Control) and zone control (Van den Boom and Backx, 2010). In this case study the following LQPC cost function was chosen

$$J(\tilde{\Delta u}, x_0) = \hat{x}_N^T P_c \hat{x}_N + \sum_{k=0}^{N_p-1} \hat{x}_k^T Q_c \hat{x}_k + \sum_{k=0}^{N_c-1} \Delta u_k^T R_c \Delta u_k \tag{35}$$

Where N_p is the prediction horizon, N_c is the control horizon, and P_c, Q_c, and R_c are the tuning knobs of the controller weighting the state at the final horizon, the state in the time window, and the control effort, respectively. The following optimization problem is solved at each sampling time using the quadprog command from the optimization toolbox in Matlab.

$$\begin{aligned} &\underset{\tilde{\Delta u}}{min} \ J(\tilde{\Delta u}, x_0) \\ &s.t \\ &x_k \geq 0 \\ &0 \leqslant u_{ex,k} \leqslant 20 \ mU/min \end{aligned} \tag{36}$$

with J as in (35).

Constraints

State constraints: the only state constraint considered is that every state should be greater than zero.

$$x_k \geq 0$$

where $x_k = [B_{G,k} \ I_{G,k} \ Q_k \ I_k \ P_{1,k} \ P_{2,k}]^T$.

Input constraints: as it was shown before, the control input corresponds to u_{ex}, the exogenous insulin. In this sense, the smaller amount of insulin to be infused is zero and the largest amount is provided by the physical constraint of the catheter.

$$0 \leqslant u_{ex,k} \leqslant 20 mU/min$$

Initial conditions and controller tuning: the MPC was designed by setting a control horizon $N_c = 50$, and a prediction horizon $N_p = 100$. Regarding to the setting up of the weighting matrices, a greater weight was assigned for states $B_{G,k}$ and $I_{G,k}$ because the purpose of the MPC is to control glucose levels. The tuning of the matrices P_c, Q_c, and R_c was performed by a trial-and-error procedure such that an acceptable performance of the MPC is reached. The matrices are presented below

$$Q_c = \begin{bmatrix} 10000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (37)$$

$$P_c = \begin{bmatrix} 10000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (38)$$

$$R_c = 1 \quad (39)$$

5. THE COUPLING OF THE EKF AND THE MPC

The separation principle states that if a stable observer is combined with a stable controller, the resulting coupled system is stable and functional. In this case, an EKF is tuned to work with the MPC to perform an output feedback to control the blood glucose concentration of a patient at the ICU. The simulation of the MPC coupled with EKF was performed in the software Matlab® version 2016*a* using the *Runge Kutta* numerical method with a fixed step of 1 min during 1000 *min*. The state of the glucose-insulin system was estimated using the designed EKF from the measurement of the interstitial glucose ($I_{G,k}$). The behavior of the EKF is shown in Figure 4

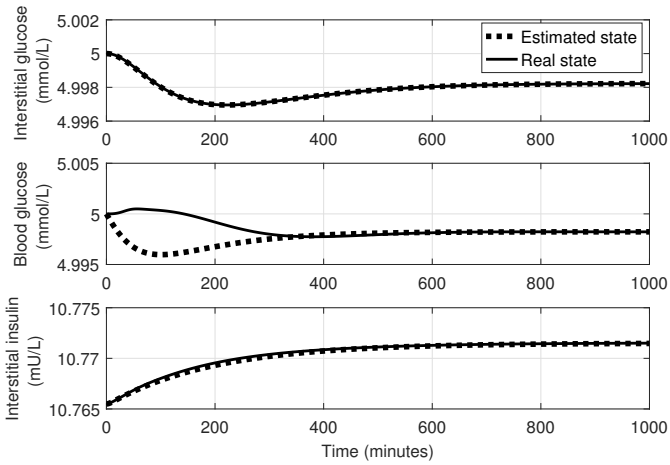
Once known the complete state of the system, the MPC is coupled to the EKF so that it will control the blood glucose, by just measuring the interstitial glucose. In a first approximation, the system was perturbed changing the enteral feeding from $D_k = 0.7672$ *mmol/min* to $D_k = 1.5344$ *mmol/min* at time $k = 250$ *min*. The results are reported in Figure 5.

As seen in Figure 5, by perturbing the system at time $k = 250$ *min* with the enteral feeding $D_k = 1.5344$, the blood glucose level reaches an overshoot of $B_{G,k} \approx 5.6$ *mmol/L* to which the MPC responds by infusing of insulin $u_{ex,k} \approx 18.5$ *mU/min*. It is important to make evident the existence of a steady-state error due mostly to the use of a linear controller in a highly nonlinear system.

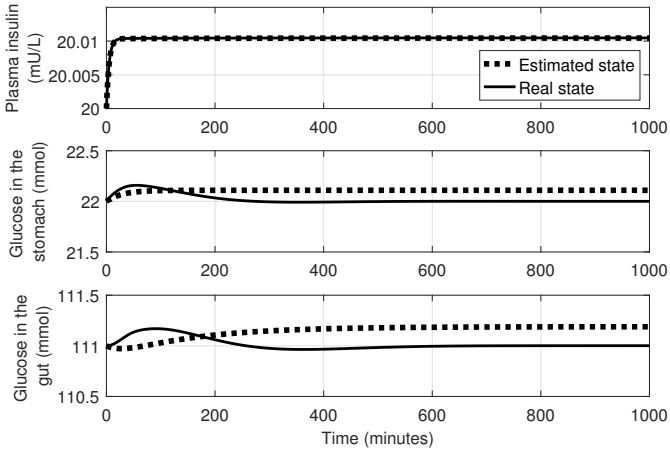
6. FINAL COMMENTS

First, according to the evidence from the simulations, it is concluded that the design of combined control and estimation schemes becomes a reachable technology to be used in health care applications as the presented. In this sense, the filter and controller should be carefully tuned together in order to guarantee the desired closed-loop behavior. To control hyperglycemia in patients at the ICU is important to weight the main state $B_{G,k}$ (Blood glucose). It is also important to note that the performance of the MPC is not as expected due to the nonlinearity of the real system.

In a future work, regarding to the state estimation, different nonlinear strategies such as the Unscented Kalman Filter (UKF) and Particle Filters (PF) with different configurations are desired to be tested. In this sense, better estimation strategies will be coupled with more reliable



(a) Estimation of $I_{G,k}$,$B_{G,k}$ and Q_k



(b) Estimation of $I_{G,k}$,$P_{1,k}$ and $P_{2,k}$

Fig. 4. States estimated by using an EKF

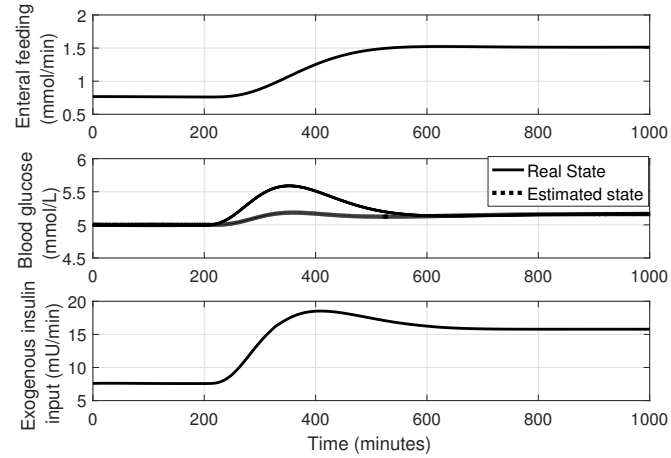


Fig. 5. Plasma glucose control when the system has been disturbed with enteral feeding $D_k = 1.5344$.

model-based controllers such as the nonlinear MPC. Since the literature about nonlinear MPC is vast, a careful review should be done in order to test the state-of-the

art of the strategy and hence to propose control schemes tailored to the specific case study.

REFERENCES

Aldworth, J., Al Bache, N., Hegelund, M.H., Linnenkamp, U., Magliano, D., Oomatia, F., Patterson, C., Peer, N., Pritulskiy, A., Saleh, M.M.A., Shelestova, E., Tamayo, T., Usher-Smith, J., Xiuying, Z., and Samrawit Yisahak (2015). *IDF Diabetes Atlas*. 7 edition.

Barrett, K.E. (2013). *Ganong's Review of Medical Physiology (24a)*. McGraw Hill Mexico.

Bequette, B.W. (2007). Analysis of algorithms for intensive care unit blood glucose control. *Journal of diabetes science and technology*, 1(6), 813–824.

Block, C.D., Vertommen, J., Manuel-y Keenoy, B., and Gaal, L.V. (2008). Minimally-invasive and non-invasive continuous glucose monitoring systems: Indications, advantages, limitations and clinical aspects. *Current diabetes reviews*, 4(3), 159–168.

Hospital Universitario de Bucaramanga S.A (2012). Insulin Administration Protocol (in Spanish).

King, C., Anderson, S.M., Breton, M., Clarke, W.L., and Kovatchev, B.P. (2007). Modeling of calibration effectiveness and blood-to-interstitial glucose dynamics as potential confounders of the accuracy of continuous glucose sensors during hyperinsulinemic clamp. *journal of diabetes science and technology*, 1(3), 317–322.

Lin, J., Razak, N.N., Pretty, C.G., Le Compte, A., Docherty, P., Parente, J.D., Shaw, G.M., Hann, C.E., and Chase, J.G. (2011). A physiological intensive control insulin-nutrition-glucose (icing) model validated in critically ill patients. *Computer methods and programs in biomedicine*, 102(2), 192–205.

Magni, L., Raimondo, D.M., Bossi, L., Dalla Man, C., De Nicolao, G., Kovatchev, B., and Cobelli, C. (2007). Model predictive control of type 1 diabetes: an in silico trial. *Journal of diabetes science and technology*, 1(6), 804–812.

Pascual, A. (2004). Ekf y ukf: two extentions of the kalman filter for nonlinear systems in an inverted pendulum (in spanish).

Schmidt, S.F. (1966). Applications of state space methods to navigation problems. *Advances in Control Systems*, 3, 293–340.

Simon, D. (2006). *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons.

Umpierrez, G.E., Isaacs, S.D., Bazargan, N., You, X., Thaler, L.M., and Kitabchi, A.E. (2002). Hyperglycemia: an independent marker of in-hospital mortality in patients with undiagnosed diabetes. *The Journal of Clinical Endocrinology & Metabolism*, 87(3), 978–982.

Van den Berghe, G., Wilmer, A., Hermans, G., Meersseman, W., Wouters, P.J., Milants, I., Van Wijngaerden, E., Bobbaers, H., Bouillon, R., et al. (2006). Intensive insulin therapy in the medical icu. *New England Journal of Medicine*, 354(5), 449.

Van den Boom, T.J. and Backx, T. (2010). Model predictive control. *DISC Course, Lecture Notes*, 16.

Villamarín, R.I. and Puentes, F.E. (2009). Intensive insulin therapy in patients with severe sepsis and septic shock (in Spanish). *Medical files*, 9(2), 165–173.

Appendix A. MATHEMATICAL MODEL PARAMETERS

Table A.1. System parameters

P	**Value**	**Units**	**Description**
B_G	*State*	$mmol/L$	Blood glucose
I_G	*State*	$mmol/L$	Interstitial glucose
Q	*State*	mU/L	Interstitial insulin
I	*State*	mU/L	Plasma insulin
P_1	*State*	$mmol$	Glucose in the stomach
P_2	*State*	$mmol$	Glucose in the gut
P_G	0.006	min^{-1}	Patient endogenous glucose removal
S_I	0.002	$L/mU/min$	From former model since it is identified online
α_G	0.0154	L/mU	Saturation of insulin-stimulated glucose
$P(t)$	*Fcn*	$mmol/min$	External nutrition
EG_{P_b}	1.16	$mmol/min$	Basal endogenous glucose production
CNS	0.3	$mmol/min$	Insulin independent central nervous system glucose uptake
V_G	13.3	L	Glucose distribution volume
V_I	3.15	L	Insulin distribution volume
α_I	0.0017	L/mU	Saturation of plasma insulin disappearance
n_C	0.003	min^{-1}	Parameter
n_I	0.003	min^{-1}	Transcapillary difussion rate
n_K	0.0542	min^{-1}	Kidney clearance
n_L	0.1578	min^{-1}	Patient specific liver clearance
u_{ex}	7.5933	mU/min	Exogenous insulin input
x_L	0.67	[]	First pass endogenous insulin hepatic uptake
u_{en}	*Fcn*	mU/min	Endogenous insulin production
d_1	0.0347	min^{-1}	Transport rate
d_2	0.0069	min^{-1}	Transport rate
$D(t)$	0.7672	$mmol/min$	Disturbance-Amount of dextrose from enteral feeding
P_{max}	6.11	$mmol/min$	Saturation value of P_2
$PN(t)$	0	$mmol/min$	Parenteral dextrose (intravenous)
k_1	45.7	mU/min	Base rate for endogenous insulin production
k_2	1.5	[]	Generic constant for exponential suppression
k_3	1000	[]	Generic constant for exponential suppression
β_1	0.0099	min^{-1}	Parameter
β_2	0.0099	min^{-1}	Parameter

Parameter Optimization of Sliding Mode Observer-based Controller for 2 DOF Stewart Platform ⋆

Sajjad Keshtkar * **Jaime Moreno** * **Alexander Poznyak** ‡

* *Instituto de ingeniería, Universidad Nacional Autónoma de México, (e-mail: diesel253m@gmail.com).*
† *Instituto de ingeniería, Universidad Nacional Autónoma de México, (e-mail: jmorenoP@iingen.unam.mx).*
‡ *Automatic Control Department, CINVESTAV - IPN, Mexico City 2508, Mexico, (e-mail: apoznyak@ctrl.cinvestav.mx)*

Abstract: This paper considers gain parameter election of a class of nonlinear controller-observer for a systems with uncertainty, using Paerto-set approach. The presented observer based on Super-twisting observer is used for velocity estimation/reconstruction signals based on the available plant input/output information and can be calculated on-line. The signal then is injected to the classical sliding mode controller to provide a feedback control in presence of unmodeled dynamics and perturbations. To guarantee the convergence of the proposed algorithm the gain parameters must be time-varying and depending on available current measurements. The optimal selection of the gains are proposed in this work. A simulation study on a 2 degrees of freedom Stewart platform is presented to show the effectiveness of the scheme.

Keywords: Sliding mode control, Super-Twisting observer, Pareto Optimization, Stewart Platform, Parallel Manipulator.

1. INTRODUCTION

1.1 Brief survey

Previous work (Keshtkar et al. (2016)) has considered the use of sliding mode controller-observer for high performance control and observation of a Stewart platform in presence of unmodeled dynamics and perturbations. The paper builds on this work and examines the parameter election of such a scheme within the broader context of optimization theory.

In the named work, the control based on sliding mode techniques is proposed for the case when not all states are measured directly but are estimated on-line by the second-order *sliding mode* (SM) observer. Such a design approach can reduce the cost of the controlled system, avoid the fragility of velocity sensors, and eliminate the difficulty of the sensors installation. Recently the design of such observers for nonlinear systems has received a great deal of attention: the standard Luenberger Nikolaos and Costas (1998); Xiaosong et al. (2010), the SM observers Edwards and Tanb (2006); Shtessel et al. (2014); Utkin et al. (2008), the extended Kalman filter Gobbo et al. (2001); Weiss and Moore (1980) and H_∞ observers Jung et al. (2006). Among them, adaptive sliding mode (SM) observers are particulary attractive technology due to their robustness against disturbances, parameter deviations and measurement noise and are widely used in different fields Shtessel et al. (2014). Some works Kima et al. (2004); Li et al. (2005); Zheng et al. (2000) propose the adaptive SM observer for sensorless induction motor drive. In Utkin et al. (2008) the adaptive SM observer - controller system was proposed for the induction machine control under unknown parameters and partial state variable information. In Fridman et al. (2008) the authors proposed a feedback linearization-based controller with a high-order SM observer running parallel and applied to a quadrotor unmanned aerial vehicle. The high-order SM observer works as an observer and estimator of the effect of the external disturbances such as wind and noise.

The problem treated in previous paper can be described as follows: estimate on-line unknown variables first and then, use these estimates to design a SM controller. However, the parameters of the controller as well as the observer are admitted to be time-varying and state-dependent. In this work the optimal selection of these gain parameters using the Pareto-set optimization method is presented. The task is to provide a simultaneous optimization two different functions by the same arguments. A standard technique for generating the Pareto set in multicriteria optimization problems is to minimize (convex) weighted sums of the different objectives for various different settings of the weights. The concept has applications in academic fields such as economics, engineering, and the life sciences (See Jornada and Leon (2016); Khoroshiltseva et al. (2016); Luo et al. (2016); Wan et al. (2016)).

1.2 Main contribution

This paper presents

⋆ The authors gratefully acknowledge the financial support from DGAPA-UNAM.

- An adaptive sliding mode controller, which uses the state estimated on-line by the adaptive Super-Twist observer;
- Optimization gain parameters of the named observer and controller by Pareto-set approach;
- Application of the studied approach to a new two DOF Stewart platform as a solar tracker

2. SYSTEM DESCRIPTION AND CONTROL DESIGN

Consider the mathematical model of 2 DOF Stewart platform

$$\begin{array}{c} \dot{X}_1(t) = X_2(t) \\ \dot{X}_2(t) = f(X,t) + g(X_1,t)\,u + \xi(t) \\ y(t) = X_1(t) \end{array} \tag{1}$$

- $X_1(t)$, $X_2(t) \in \mathbb{R}^2$ are the states of the system;

- $X(t) := \left(X_1^\top(t), X_2^\top(t)\right)^\top \in \mathbb{R}^4$;

- $u \in \mathbb{R}^2$ is a control to be designed,

- $y(t) \in \mathbb{R}^2$ is a measurable output at time $t \geq 0$,

- $\xi(t) \in \mathbb{R}^2$ is bounded unmeasurable term including external and internal perturbations/uncertainties: for all $t \geq 0$, $\|\xi(t)\| \leq \xi^+ < \infty$,

- $f(X,t) \in \mathbb{R}^2$

- $g(X_1,t) \in \mathbb{R}^{2\times 2}$

Remark 1. Notice that in the model (1) only the half of coordinates ($X_1(t)$) is available in time; $X_2(t)$ is supposed to be estimated online.

2.1 Observed sliding mode control design

To estimate the non-measurable coordinate $X_2(t)$ in (1) we apply a popular second-order sliding mode (super-twisting) observers Shtessel et al. (2014):

$$\begin{array}{c} \dot{\hat{X}}_1 = \hat{X}_2 + \lambda \left\| y - \hat{X}_1 \right\|^{1/2} \mathrm{Sign}\left(y - \hat{X}_1\right) \\ \dot{\hat{X}}_2 = f\left(\hat{X},t\right) + g(X_1,t)\,u + \alpha \mathrm{Sign}\left(y - \hat{X}_1\right) \end{array} \tag{2}$$

The parameters α and λ are constant in the original publications. The vector function $\mathrm{Sign}(z)$ is defined as follows

$$\begin{array}{c} \mathrm{Sign}(z) := (\mathrm{sign}(z_1), ..., \mathrm{sign}(z_n))^\intercal \\ \mathrm{sign}(z_i) := \begin{cases} 1 & \text{if } z_i > 0 \\ -1 & \text{if } z_i < 0 \\ \in [-1,1] & \text{if } z_i = 0 \end{cases} \end{array} \tag{3}$$

New variable $e := \left(X - \hat{X}\right)$ which characterizes the error of estimated state and is governed by the following equations:

$$\begin{array}{c} \dot{e}_1 = e_2 - \lambda \|e_1\|^{1/2} \mathrm{Sign}(e_1) \\ \dot{e}_2 = F_e\left(t, X_1, X_2, \hat{X}_2\right) - \alpha \mathrm{Sign}(e_1) \end{array} \tag{4}$$

where

$$F_e\left(t, X_1, X_2, \hat{X}_2\right) = \left[f(X,t) - f\left(\hat{X},t\right)\right] + \xi(X,t) \tag{5}$$

describes unmodeled dynamics and external perturbation effects. Obviously, it is unmeasurable.

The feedback control law $u\left(\hat{X},t\right)$ must derive the states of the systems to the desired values. i.e., $\lim\limits_{t\to\infty} \hat{X}_1 = X_1^*$, $\lim\limits_{t\to\infty} \hat{X}_2 = 0$. By defining the *"ideal"* sliding surface (Utkin (1992)(:

$$\sigma\left(\hat{X}\right) := \dot{\hat{X}}_1 + C\left(\hat{X}_1 - X_1^*\right) = \hat{X}_2 + C\left(\hat{X}_1 - X_1^*\right) = 0 \tag{6}$$

where $C = \mathrm{diag}(c_1, c_2)$ is a diagonal matrix with positive elements, the SM controller $u\left(\hat{X},t\right)$ structure can be defined as

$$u\left(\hat{X},t\right) = -k\left(\hat{X},t\right)\left[g(X_1,t)\right]^{-1} \mathrm{Sign}\left(\sigma\left(\hat{X}\right)\right) \tag{7}$$

where the state estimates $\hat{X}$ are generated by the observer (2) with varying parameters, that is,

$$\begin{array}{c} \dot{\hat{X}}_1 = \hat{X}_2 + \lambda\left(\hat{X},t\right)\left\|y - \hat{X}_1\right\|^{1/2} \mathrm{Sign}\left(y - \hat{X}_1\right) \\ \dot{\hat{X}}_2 = f\left(\hat{X},t\right) + g(X_1,t)\,u + \alpha\left(\hat{X},t\right) \mathrm{Sign}\left(y - \hat{X}_1\right) \end{array} \tag{8}$$

2.2 Gain parameters tuning

First by the physical reasons we may accept the following assumptions.

H1 The vector-function $f(X,t)$ and $g(X_1,t)$ are Lipschitzian with respect to the first argument on $t \geq 0$, i.e.,

$$\begin{array}{c} \left\|f(X,t) - f\left(\hat{X},t\right)\right\| \leq L_f \left\|X - \hat{X}\right\| = L_f \|e\| \\ \left\|g(X_1,t) - g\left(\hat{X}_1,t\right)\right\| \leq L_g \left\|X_1 - \hat{X}_1\right\| = L_g \|e_1\| \end{array}$$

H2 The matrix $g(X_1,t)$ function is invertible everywhere, namely

$$\left\|\left[g(X_1,t)\right]^{-1}\right\| \leq g_{-1}^+ < \infty, \ \left(\|g\|^2 := \mathrm{tr}\left\{gg^\intercal\right\}\right)$$

Notice that by (5)

$$\left\|F_e\left(t, X_1, X_2, \hat{X}_2\right)\right\|^2 \leq 2L_f \|e\|^2 + 2\left(\xi^+\right)^2 \tag{9}$$

The classical second-order scheme corresponds to the case $L_f = 0$. The situation, corresponding the inequality (9), has not been considered elsewhere. The adaptation parameters $k = k\left(\hat{X}, e_1, t\right)$ of the controller (7) and $\lambda\left(\hat{X},t\right)$, $\alpha\left(\hat{X},t\right)$ of the observer (8) are designed as

$$\left.\begin{array}{c} k = k\left(\hat{X}, e_1, t\right) := (p_0)^{-1}\left[\rho + \varkappa\left(\hat{X}, e_1, t\right) \right. \\ \left. + \gamma\sqrt{p_0/2}\right], \quad \rho \geq \sqrt{p_0/2} \\ \varkappa\left(\hat{X}, e_1, t\right) := \left\|f\left(\hat{X},t\right) + \alpha \mathrm{Sign}(e_1) + \right. \\ \left. C\left[\hat{X}_2 + \lambda \|e_1\|^{1/2} \mathrm{Sign}(e_1)\right]\right\| \end{array}\right\} \tag{10}$$

and

$$\left.\begin{array}{c} \lambda\left(\hat{X},t\right) := \begin{cases} (\eta_1(\gamma) + \sqrt{p_1/2})/p_1 \sqrt{\|e_1\|} \\ \text{if} \quad \sqrt{\|e_1\|} \geq \epsilon > 0 \\ \text{and} \\ (\eta_1(\gamma) + \sqrt{p_1/2})/p_1 \epsilon \\ \text{if} \quad \sqrt{\|e_1\|} < \epsilon \end{cases} \\ \alpha\left(\hat{X},t\right) := \alpha > 0 \end{array}\right\} \tag{11}$$

Theorem 1. Under the assumptions *H1-H2* the SM controller (7) with the adaptive gain parameters (10)-(11) guarantees the finite-time convergence of the closed-loop system trajectories to a μ-zone around the desired values, namely, for the "storage" function

$$V(t) := V\left(\sigma\left(\hat{X}(t)\right), e_1(t), e_2(t)\right) = \frac{p_0}{2}\left\|\sigma\left(\hat{X}(t)\right)\right\|^2 + \frac{p_1}{2}\|e_1(t)\|^2 + \frac{p_2}{2}\|e_2(t)\|^2 \qquad (12)$$
$$p_0,\ p_1,\ p_2\ > 0$$

we guarantee that it becomes less than μ^2 for all $t \geq t_f = \gamma^{-1}\sqrt{2W(0)}$ fulfilling

$$W(t) := \left[\sqrt{V(t)} - \mu\right]_+^2 = 0 \qquad (13)$$

where

$$[z]_+ := \begin{cases} z \text{ if } z \geq 0 \\ 0 \text{ if } z < 0 \end{cases}$$
$$\mu := \mu_0\left(\alpha, \gamma, e_2^+\right) + \mu_1(\epsilon)\, e_2^+$$
$$\mu_0\left(\alpha, \gamma, e_2^+\right) := \eta_0\left(\alpha, e_2^+\right) + \tfrac{10}{27}\gamma\epsilon\sqrt{\tfrac{p_1}{2}} \qquad (14)$$
$$\mu_1(\epsilon) := \left[\sqrt{\tfrac{p_2}{2}} + \epsilon\tfrac{10}{27}\left(p_1 + \sqrt{2L_f}p_2\right)\right]$$

and

$$\eta_0(\alpha) := p_2 e_2^+\left(\alpha\sqrt{n} + \sqrt{2}\xi^+ + \sqrt{2L_f}e_2^+\right) + \gamma\sqrt{\tfrac{p_2}{2}}$$
$$\eta_1(\gamma) := \gamma\sqrt{\tfrac{p_1}{2}} + \left(p_1 + \sqrt{2L_f}p_2\right)e_2^+,\ \gamma > 1$$
$$e_2^+ = 2c/\sqrt{b^2 + 4ac} + b \qquad (15)$$
$$a := \sqrt{2L_f}p_2,\ c := (\gamma - 1)\,\epsilon\tfrac{10}{27}\sqrt{\tfrac{p_1}{2}},\ \epsilon > 0$$
$$b := \epsilon\tfrac{10}{27}\left(p_1 + \sqrt{2L_f}p_2\right) + p_2\left(\alpha\sqrt{n} + \sqrt{2}\xi^+\right)$$

The initial values of the observer $\hat{X}_2(0)$ should be not so far from the real values $X_2(0)$ fulfilling $\left\|X_2(0) - \hat{X}_2(0)\right\| \leq e_2^+$ [1].

3. OPTIMIZATION OF THE CONTROLLER PARAMETERS

The control gain (10) contains three free parameters p_0, p_1, p_2 which should be selected in such a way that the resulting behavior would be as better as possible. Without loss of generality we may select the parameters p_0, p_1 and p_2 of the controller as follows

$$p_0 + p_1 + p_2 = 1$$

implying

$$p_2 = 1 - p_1 - p_0 \qquad (16)$$

Indeed, since

$$\arg\min_{u \in U_{adm}} V = \arg\min_{u \in U_{adm}} \frac{1}{p_0 + p_1 + p_2} V$$

the multiplication of the function V by any positive constant does not change the optimal control parameters. If we fix the value $p_0 \in (0,1)$, then in view of (16) we have only one free parameters $p_1 \in (0,1)$ which should be selected to fulfill the following optimality requirements:

1) to make the "workability zone" $e_2^+ = \dfrac{2}{\mathcal{F}_1(p_1)}$ as much as possible which corresponds to the minimization of the function

[1] The proof is described in Keshtkar et al. (2016)

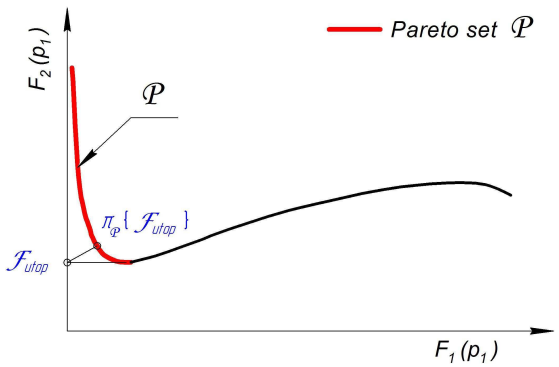


Fig. 1. The set of Pareto optimal points.

$$\mathcal{F}_1(p_1) := \sqrt{p_1}r_1 + \frac{1 - p_1 - p_0}{\sqrt{p_1}}r_2 + \sqrt{\left(\sqrt{p_1}r_1 + \frac{1 - p_1 - p_0}{\sqrt{p_1}}r_2\right)^2 + \frac{1 - p_1 - p_0}{p_1}r_3} \to \min_{p_1 \in (0,1)}$$

2) to make the convergence zone μ minimal possible one which corresponds to the minimization of the function

$$\mu = \mathcal{F}_2(p_1) := r_1 + \frac{r_2}{\mathcal{F}_1(p_1)} + \frac{r_3}{\mathcal{F}_1^2(p_1)} \to \min_{p_1 \in (0,1)}$$

One can see that we deal with the multi-functional optimization problem

$$\mathcal{F}_1(p_1) \to \min_{p_1 \in (0,1)}$$
$$\mathcal{F}_2(p_1) \to \min_{p_1 \in (0,1)}$$

and it is clear that simultaneous optimization two different functions by the same arguments p_1 is impossible. That's why we need to apply the, so-called, "*Pareto-set approach*". A standard technique for generating the Pareto set in multicriteria optimization problems is to minimize (convex) weighted sums of the different objectives for various different settings of the weights. *Pareto efficiency*, or *Pareto optimality*, is a state of allocation of resources in which it is impossible to make any one individual better off without making at least one individual worse off Ehrgott (2005).

The term is named after Vilfredo Pareto (1848–1923), an Italian engineer and economist who used the concept in his studies of economic efficiency and income distribution. Many real-world problems involve simultaneous optimization of several incommensurable and often competing objectives. Usually, there is no single optimal solution, but rather a set of alternative solutions. These solutions are optimal in the wider sense that no other solutions in the search space are superior to them when all objectives are considered. They are known as *Pareto-optimal solutions* Ehrgott (2005).

In our case the Pareto set looks as in the Fig.1.

It this figure

$\mathcal{P}$ is the Pareto set,

$\mathcal{F}_{utop} = \left(\min_{p_1 \in (0,1)} \mathcal{F}_1(p_1), \min_{p_1 \in (0,1)} \mathcal{F}_2(p_1)\right)$ is the utopia point,

$\pi_{\mathcal{P}}\{\mathcal{F}_{utop}\}$ is the projection of the utopia point $\mathcal{F}_{utop}$ to the Pareto set.

Definition. *The parameters p_1^* and p_2^*, corresponding to the point $\pi_{\mathcal{P}}\{\mathcal{F}_{utop}\}$ on the Pareto set, we will referred to as the* ***optimal parameters*** *of the control algorithm.*

4. NUMERICAL SIMULATIONS

The work described in previous sections will now be applied to a 2 DOF Stewart platform (2 rotations), which has been used as a solar tracker for photovoltaic panels. The main objective of the simulation is to control the rotations and the angular velocity of the moving platform of the parallel robot. The optimal parameters will be compared with the arbitrary tuned values obtained in previous paper.

The proposed mechanism (see Fig. 2) consists of two active linear actuators and one passive rod attached to a triangular base with spherical joints. Each actuator is made of upper and lower rods, connected by translational kinematic pair (screw mechanism). An anti-plunging strut, connected rigidly to the base and from the other end by Hooke's joint to the moving platform, inhibits the translational movements and one rotation, giving the system the desired two DOF.

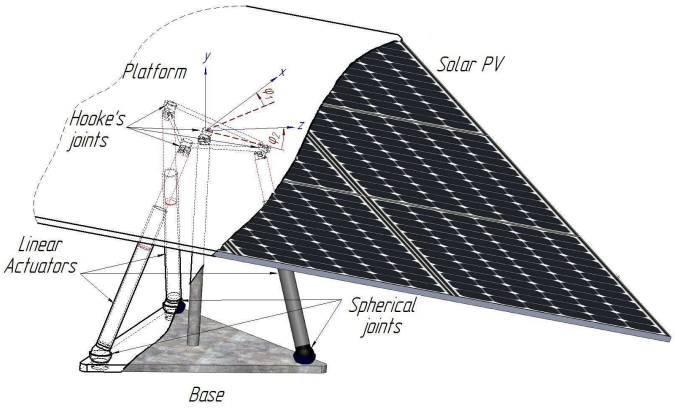


Fig. 2. Solar tracker on tripod type platform with two rotating DOF.

The orientation of the moving platform, and consequently the solar photovoltaic (PV) panel attached to it, regulates by changing the lengths of two actuators. This causes a corresponding change in the angle of the inclination of panel respect to the ground. In this case the variables $X(t) := (x_1, x_2, x_3, x_4) = (\varphi_1, \varphi_2, \dot{\varphi}_1, \dot{\varphi}_2)$ refer to the angular rotation and angular velocities of the moving platform. The trajectory of tracking is given either by the precalculated mathematical algorithms or by the measurement of light intensity in real time. This mechanism is specially important in solar tower applications to reach the permissable angle errors between the mirror and the central receiver located in the tower structure. For the model (1) we have:

$$\xi(t) = 1,5sin(x_1 + t)$$

$$f(X,t) = \begin{pmatrix} -\dfrac{x_3 x_4 \left(J_y c_{2x_2} - J_z s_{2x_2}\right)}{\left(J_y s_{x_2}^2 + J_z c_{x_2}^2\right)} \\ \dfrac{x_3^2 x_4 \left(J_y c_{2x_2} - J_z s_{2x_2}\right)}{J_x} \end{pmatrix} \tag{17}$$

$$g(X_1,t) = \begin{pmatrix} \dfrac{\sum \cos\gamma_{1,j}}{\left(J_y s_{x_2}^2 + J_z c_{x_2}^2\right)} & 0 \\ 0 & \dfrac{\sum \cos\gamma_{2,j}}{J_x} \end{pmatrix}$$

where J_i, $i = x, y, z$ are the main inertia moments of the moving platform and the angles γ_i $(0 < \gamma_i < \pi/2)$ are the inclination of the actuators respect to the fixed coordinate system for details see Keshtkar et al. (2016).

Remark 2. For all $X \in \mathbb{R}^4$

$$J_y s_{x_2}^2 + J_z c_{x_2}^2 \geq \min\{J_y, J_z\} > 0$$

The parameters of system and those that are participating in the controller realization, are given in Table 1.

Table 1. Numerical values of the parameters.

$p_0 = 0,2$	$\alpha = 2$	$\gamma = 1$	$J_z[\text{kgm}^2] = 2.5$
$p_{1opt} = 0.1040$	$e_2^+ = 4$	$h[\text{m}] = 0.4$	$x_1^*[\text{rad}] = 20$
$p_{2opt} = 0.6960$	$L_f = 2$	$R[\text{m}] = 0.4$	$x_2^*[\text{rad}] = 10$
$c_1 = 1.5$	$\xi^+ = 2$	$r[\text{m}] = 0.2$	$x_3^*[\text{rad/s}] = 0$
$c_2 = 1.6$	$n = 2$	$J_x[\text{kgm}^2] = 2$	$x_4^*[\text{rad/s}] = 0$
$\epsilon = 1$	$L_g = 1$	$J_y[\text{kgm}^2] = 2$	

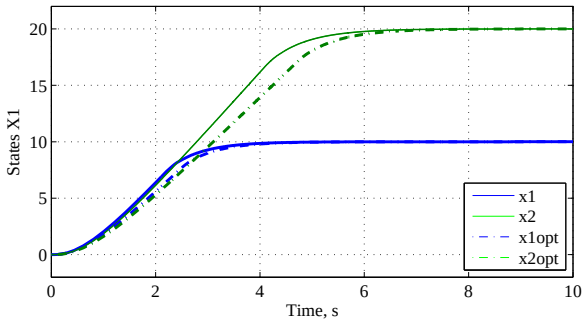


Fig. 3. Real and estimated of X_1.

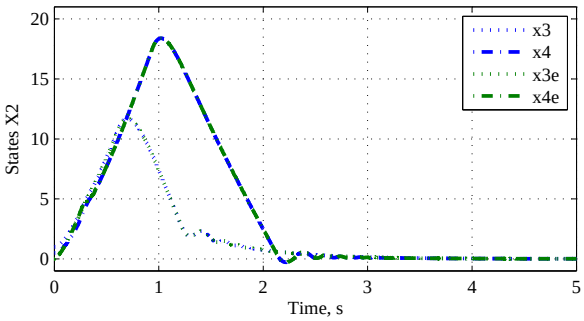


Fig. 4. Angular positions reached by conventional and optimal parameters.

From the figures 3 and 5 one can see the behavior of the real states X and the control U of the system obtained by both classical and optimal parameters of the gain. The angular position does not changes its trajectory when in

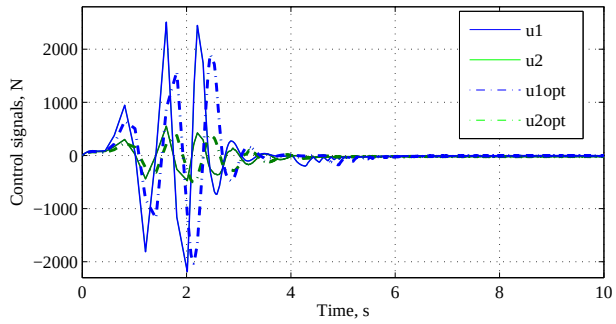


Fig. 5. Control signals representation.

the control signals we see a reduction in the control signal values.

Finally, Fig. 6 illustrates the time-variations of the adaptive control parameters $k = k(\hat{x}(t), t)$, $\lambda = \lambda(\hat{x}(t), t)$ providing the desired dynamics of the considered closed-loop system.

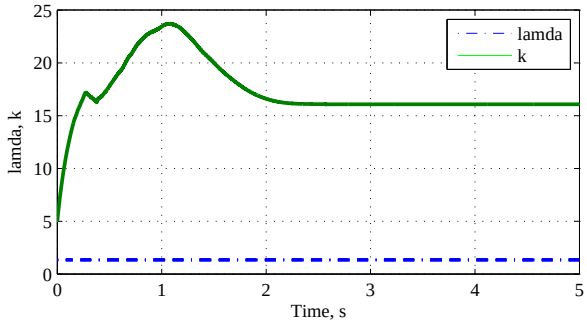


Fig. 6. Tuned parameters k and λ.

5. CONCLUSION

This paper has focused on reconstructing optimally the gain parameters in a feedback sliding mode controller-observer. Novel adaptive sliding mode observers based on Super-Twisting with the tuning gain is presented for this purpose. The proposed controller-observer robustly reconstruct the states in the presence of unknown bounded coupling strengths and a class of unknown bounded uncertainties/ nonlinearities/disturbances. The Pareto-set optimization approach is derived for exact selection of the gain parameters. A parallel manipulator with two DOF is used to demonstrate the novelty of the proposed approach via numerical simulations.

REFERENCES

Edwards, C. and Tanb, C.P. (2006). Sensor fault tolerant control using sliding mode observers. *Cont Eng Practice*, 14, 897–908.

Ehrgott, M. (2005). *Multicriteria Optimization*. Springer, Berlin, Heidelber, New York, 2 edition.

Fridman, L., Mokhtari, A., and Benallegu, A. (2008). High-order sliding-mode observer for a quadrotor uav. *Int. J. Robust Nonlinear Control*, 18, 427–440.

Gobbo, D.D., Napolitano, M., Famouri, P., and Innocenti, M. (2001). Experimental application of extended kalman filtering for sensor validation. *IEEE Trans on Control Systems Tech*, 9, 376–380.

Jornada, D. and Leon, V.J. (2016). Biobjective robust optimization over the efficient set for pareto set reduction. *European Journal of Operational Research*, 252(2), 573 – 586.

Jung, J., Huh, K., Fathy, H.K., and Stein, J.L. (2006). Optimal robust adaptive observer design for a class of nonlinear systems via an h-infinity approach. *American control conference*, In Proceedings of the 2006, 3637–3642.

Keshtkar, S., Keshtkar, J., and Poznyak, A. (2016). Adaptive sliding mode control for solar tracker orientation. *2016 American Control Conference*, (accepted).

Khoroshiltseva, M., Slanzi, D., and Poli, I. (2016). A pareto-based multi-objective optimization algorithm to design energy-efficient shading devices. *Applied Energy*.

Kima, S.M., Hanb, W.Y., and Kima, S.J. (2004). Design of a new adaptive sliding mode observer for sensorless induction motor drive. *Electric Power Systems Research*, 70, 16–22.

Li, J., Xu, L., and Zhang, Z. (2005). An adaptive sliding-mode observer for induction motor sensorless speed control. *IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS*, 41(4), 1039–1047.

Luo, Q., Wu, J., Yang, Y., Qian, J., and Wu, J. (2016). Multi-objective optimization of long-term groundwater monitoring network design using a probabilistic pareto genetic algorithm under uncertainty. *Journal of Hydrology*, 534, 352 – 363.

Nikolaos, K. and Costas, K. (1998). Nonlinear observer design using lyapunov auxiliary theorem. *Sys and Cont Letters*, 34(5), 241–247.

Shtessel, Y., Edwards, C., Fridman, L., and Levant, A. (2014). *Sliding Mode Control and Observation*. Birkhauser.

Utkin, V., Buss, M., and Rao, S. (2008). An adaptive sliding mode observer for induction machines. *ACC*, 1947–1951.

Utkin, V. (1992). *Sliding Modes in Control and Optimization*. Spronger Verlag.

Wan, L., Yuan, J., and Wei, L. (2016). Pareto optimization scheduling with two competing agents to minimize the number of tardy jobs and the maximum cost. *Applied Mathematics and Computation*, 273, 912 – 923.

Weiss, H. and Moore, J. (1980). Improved extended kalman filter design for passive tracking. *IEEE Trans. Autom. Contr.*, AC-25, 897–908.

Xiaosong, H., Fengchun, S., and Yuan, Z. (2010). Estimation of state of charge of a lithium-ion battery pack for electric vehicles using an adaptive luenberger observer.

Energies, 3, 1586–1603.
Zheng, Y., Fatta2, H.A.A., and Loparo, K.A. (2000). Nonlinear adaptive sliding mode observer-controller scheme for induction motors. *Int. J. Adapt. Control Signal Process*, 14, 245–273.

Parameter-Dependent Filter with Finite Time Boundedness Property for Continuous-Time LPV Systems ⋆

Hugo T. M. Kussaba * **Renato A. Borges** * **João Y. Ishihara** *,**

* *Department of Electrical Engineering, University of Brasília – UnB, 70910-900, Brasília, DF, Brazil (e-mail: kussaba@lara.unb.br,{raborges,ishihara}@ene.unb.br).*
** *Department of Electrical Engineering, University of California, Los Angeles (UCLA) – 90095, California, LA, United States of America*

AbstractIn this paper the problem of stable estimation for linear parameter varying (LPV) systems in finite-time setting is considered. In order to bound the states and the estimation errors, a parameter-dependent filter is proposed. Under the assumption that the parameters variation are sufficiently small, the filter existence and synthesis is characterized by linear matrix inequalities (LMI) conditions. A numerical example is provided to illustrate the proposed technique.

Keywords: Filter design, Disturbance signals, Uncertain linear systems, Linear parameter varying systems, Finite time stability, Finite time boundedness, Linear matrix inequality.

1. INTRODUCTION

Considering the challenges posed by the constant technological development, there is no doubt that the improvement of filtering techniques is of considerable importance. The use of dynamic filters to remove unwanted signal characteristics or to estimate system information from corrupted measurements is increasing within engineering. In many practical applications, this means suppressing interfering signals and reducing the effect of external noise in communication systems, electronic devices, industrial plants, among others. In fact, filter performance has been investigated under several scenarios like nonlinearities, delays, and parameter uncertainties. However, one can note that most works in the literature consider system performance only after the system had run a large amount of time, that is, filters and controllers are designed to achieve their goal only asymptotically. This is a strong theoretical limitation since for many practical applications it is important that the overall system achieves a desired state in a specified finite time. In order to fill this gap, many infinite time concepts like stability and controllability have been extended for finite time setting. In particular, a similar notion of ultimately bounded system Khalil (2002) in the finite time setting is the notion of Finite Time Stability, and in the presence of disturbances, the notion of Finite Time Boundedness (Amato et al., 2001).

The notion of Finite Time Boundedness states that the time-varying linear system

$$\dot{x}(t) = A(t)x(t) + G(t)w(t),\ \forall t \in [0,\ T] \tag{1}$$

subject to a disturbance w in a pre-specified class $\mathscr{W}$ is Finite Time Bounded (also abbreviated as FTB) with respect to $(c_1, c_2, T, R, \mathscr{W})$, with $c_2 > c_1$ and $R > 0$, if

$$x'(0)Rx(0) \leq c_1 \Rightarrow x'(t)Rx(t) \leq c_2,\ \forall t \in [0,\ T], \forall w \in \mathscr{W}.$$

In the particular case that the pre-specified class $\mathscr{W}$ is empty or $G = \mathbf{0}$, the system is said to be Finite Time Stable (FTS). Still in Amato et al. (2001), sufficient conditions for (1) being FTB are also derived in the form of a linear matrix inequality (LMI) feasibility problem. Those conditions are used for the synthesis of a state feedback controller which assures that the closed loop system is FTB. Further works in the area propose variations of the FTB definition, or of the structures of the controllers and the plant, or of the classes $\mathscr{W}$ of disturbances.

In this paper is considered a more general situation which the designer has to face an uncertain ambient—besides disturbances, the system itself has its parameters not known exactly. More specifically, it is supposed that the system is a continuous-time linear parameter varying (LPV) system. This setting is interesting because many general nonlinear systems can be converted into a LPV form (Toth, 2010). The examples range from flight and automative systems (Ganguli et al., 2002; Baslamisli et al., 2009) to anesthesia delivery (Lin et al., 2008) and diabetes control (Peña and Ghersin, 2010).

Considering the filter design problem, the goal is to guarantee that the estimation error is FTB with respect to $(c_1, c_2, T, R, \mathscr{W})$, as done in Luan et al. (2010) for stochastic systems; He and Liu (2011) for time-delay jump systems and Liu et al. (2012) for singular stochastic systems. In all these cases, the filters are designed using LMI conditions to ensure the FTB property. In principle, a very strong necessary and sufficient FTB filter design conditions could be developed based on the differential linear matrix inequality (DLMI) characterization (Amato et al., 2003, 2005, 2014) for FTB. However, this development has two drawbacks. First, it is important to salient that the DLMI approach for filtering is not trivial since the filter structure demands an analysis with an input signal, a much more challenging case than that considered by Amato et al. (2003). And second, it is noted that DLMI problems are generally computationally very expensive and in most situations even prohibitive.

⋆ Supported by the Brazilian agencies CAPES and CNPq

Therefore, it is interesting the search for solutions that uses less computational effort than the DLMIs. Thus, in this paper we aim achieving this goal by using only the LMI framework.

Under the assumption that the parameters of the plant are sufficiently slow time-varying, a new synthesis condition for a homogeneous polynomially parameter-dependent FTB filter for continuous-time LPV systems is derived in this paper. Those systems are an indexed collection of linear systems in which the indexing parameter is independent of the state (Shamma, 2012). Depending on the scenario, this indexing parameter can be seen as a parametric uncertainty of the model or a measurable parameter possibly read in real time, which can be used in the design of a controller or a filter that accounts for all possible variations of this parameter. Furthermore, the proposed approach also considers that for a particular case where the LMIs depend on a parameter α in the unit simplex, homogeneous polynomial structures can be used in the search for less conservative sets of design conditions, as done in Oliveira and Peres (2007).

This paper is organized as follows. Detailing of the problem and auxiliary lemmas are presented in Section 2. The main theorem, where LMI conditions are derived for the synthesis of a filter that solves the FTB problem for LPV systems, is proved in Section 3. A numerical example is given in Section 4 to illustrate the application of the technique. Finally, the conclusion is presented in Section 5.

In the sequel the following notation will be used: The symbol $(')$ indicates the transpose of a matrix; $P > 0$ means that P is symmetric positive definite. $\mathbb{R}$ represents the set of real numbers, $\mathbb{Z}_+ = \{0,1,2,\ldots\}$ the set of nonnegative integers. $\text{card}(\cdot)$ denotes the cardinality of a set. $\lambda_{max}(\cdot)$ and $\lambda_{min}(\cdot)$ indicate, respectively, the maximum and the minimum eigenvalue of the argument. The term $(\star)$ indicates symmetric terms in the LMIs and $\mathbf{I}$ and $\mathbf{0}$ are the identity and the zero matrices of suitable dimensions.

2. PROBLEM STATEMENT AND PRELIMINARY RESULTS

Consider a LPV system with $t \in [0,\ T]$ and

$$\begin{aligned} \dot{x}(t) &= A(\alpha(t))x(t) + B(\alpha(t))w(t), \\ y(t) &= C_y(\alpha(t))x(t) + D(\alpha(t))w(t), \\ z(t) &= C_z(\alpha(t))x(t), \end{aligned} \tag{2}$$

where $x(t) \in \mathbb{R}^n$ is the state space vector, $y(t) \in \mathbb{R}^q$ is the measured output, $z(t) \in \mathbb{R}^p$ is the signal to be estimated and $w(t) \in \mathbb{R}^r$ is the noise input with bounded L_2 norm. The parameter $\alpha(t)$ is assumed to be available online and is continuous with respect to its time dependence—which, to lighten the notation, will be omitted wherever there is no ambiguity.

All matrices are real, with appropriate dimensions and belongs to the polytope $\mathscr{P}$

$$\left[\begin{array}{c|c} A(\alpha) & B(\alpha) \\ \hline C_y(\alpha) & D(\alpha) \\ \hline C_z(\alpha) & - \end{array}\right] = \sum_{i=1}^{N} \alpha_i \left[\begin{array}{c|c} A_i & B_i \\ \hline C_{yi} & D_i \\ \hline C_{zi} & - \end{array}\right]. \tag{3}$$

For all $t \in [0,\ T]$, the system matrices are given by the convex combination of the known vertices of the polytope $\mathscr{P}$.

The vector of time-varying parameters $\alpha \in \mathbb{R}^N$ belongs for all $t \in [0,\ T]$ to the unit N-simplex Δ_N, that is:

$$\Delta_N = \left\{ \theta \in \mathbb{R}^N : \sum_{i=0}^{N} \theta_i = 1,\ \theta_i \geq 0 \right\}.$$

To account the information given by the parameter α, parameter-dependent matrices are used in the dynamics of a full order proper filter, precisely:

$$\begin{aligned} \dot{x}_f(t) &= A_f(\alpha)x_f(t) + B_f(\alpha)y(t), \\ z_f(t) &= C_f(\alpha)x_f(t), \end{aligned} \tag{4}$$

where $x_f(t) \in \mathbb{R}^n$ is the filter state and $z_f(t) \in \mathbb{R}^p$ the estimated signal. Coupling the filter to the plant, the equations that describe the augmented system dynamics are given by

$$\begin{aligned} \dot{\varsigma}(t) &= \bar{A}(\alpha)\varsigma(t) + \bar{B}(\alpha)w(t), \\ e(t) &= \bar{C}(\alpha)\varsigma(t), \end{aligned} \tag{5}$$

where $\varsigma(t) = [x(t)'\ \ x_f(t)']'$, $e(t) = z(t) - z_f(t)$, and

$$\bar{A}(\alpha) = \begin{bmatrix} A(\alpha) & \mathbf{0} \\ B_f(\alpha)C_y(\alpha) & A_f(\alpha) \end{bmatrix},\ \bar{B}(\alpha) = \begin{bmatrix} B(\alpha) \\ B_f(\alpha)D(\alpha) \end{bmatrix},$$
$$\bar{C}(\alpha) = \begin{bmatrix} C_z(\alpha) & -C_f(\alpha) \end{bmatrix}.$$

It is desirable that both the state of the plant and the error between it and the state of the filter are bounded during a finite time horizon. This motivates Definition 1.

Definition 1. Given three positive scalars c_1, c_2 and T, with $c_2 > c_1$, positive definite matrices $R_p \in \mathbb{R}^{n\times n}$ and $R_e \in \mathbb{R}^{n\times n}$ and the class of signals $\mathscr{W}_d$, the LPV system

$$\dot{\varsigma}(t) = \bar{A}(\alpha)\varsigma(t) + \bar{B}(\alpha)w(t) \tag{6}$$

is FTB with respect to $(c_1, c_2, \mathscr{W}_d, T, R_p, R_e)$, if

$$\begin{bmatrix} x(0) \\ x(0) - x_f(0) \end{bmatrix}' \begin{bmatrix} R_p & \mathbf{0} \\ \mathbf{0} & R_e \end{bmatrix} \begin{bmatrix} x(0) \\ x(0) - x_f(0) \end{bmatrix} \leq c_1$$

implies that

$$\begin{bmatrix} x(t) \\ x(t) - x_f(t) \end{bmatrix}' \begin{bmatrix} R_p & \mathbf{0} \\ \mathbf{0} & R_e \end{bmatrix} \begin{bmatrix} x(t) \\ x(t) - x_f(t) \end{bmatrix} \leq c_2$$

for all $w \in \mathscr{W}_d$ and for all $t \in [0,\ T]$.

Remark 1. It should be noted that the definition of a FTB system presented here is a specialization for LPV systems of the definition presented in Amato et al. (2001) with R chosen as:

$$R = \begin{bmatrix} R_p + R_e & -R_e \\ -R_e & R_e \end{bmatrix}. \tag{7}$$

The matrices R_p and R_e can be seen as weighting matrices that set the importance between bounding the states of the plant and the error between it and the states of the filter. In contrast to a classical scenario in which the Lyapunov stability of $x - x_f$ implies the Lyapunov stability of $z - z_f$ using a Luenberger observer as a filter, constraining the error $x - x_f$ in a region during a finite time does not necessarily imply that $z - z_f$ satisfies the same constraining. In fact, $z - z_f$ can be bounded independently of $x - x_f$, motivating the next definition.

Definition 2. Given a symmetric positive definite matrix Ω, the filter (4) is said to be Ω-bounded in finite time T if its estimation error $e(t)$ satisfies

$$e'(t)e(t) < \varsigma'(t)\Omega^{-1}\varsigma(t),\ \ \forall t \in [0,\ T]. \tag{8}$$

Taking into account the above discussion, the FTB filtering problem to be solved in this work is formally stated as follows.

Problem 1. Assuming that $\alpha \in \Delta_N$ is available online for every $t \in [0,\ T]$ and its variation is sufficiently small, find matrices $A_f(\alpha)$, $B_f(\alpha)$ and $C_f(\alpha)$ in (4), such that the augmented

system (5) is FTB with respect to $(c_1, c_2, \mathscr{W}_d, T, R_p, R_e)$ and the estimation error is Ω-bounded in finite time T.

The subsequent lemma, from Amato et al. (2001) with an extended proof in Borges et al. (2013) considering a wider class of noises, presents a sufficient condition to analyze if a LPV system is FTB and it is used in the solution of Problem 1.

Lemma 1. For a sufficiently slow varying parameter α, system (6) is FTB with respect to $(c_1, c_2, \mathscr{W}_d, T, R_p, R_e)$, if, for all $\alpha \in \Delta_N$, there exist positive definite symmetric matrices $Q_1(\alpha) \in \mathbb{R}^{2n\times 2n}$, $Q_2(\alpha) \in \mathbb{R}^{r\times r}$ and a positive scalar β such that

$$\begin{bmatrix} \bar{A}(\alpha)\tilde{Q}_1(\alpha) + \tilde{Q}_1(\alpha)\bar{A}'(\alpha) - \beta\tilde{Q}_1(\alpha) & \bar{B}(\alpha)Q_2(\alpha) \\ (\star) & -\beta Q_2(\alpha) \end{bmatrix} < \mathbf{0}, \tag{9}$$

$$\frac{c_1}{\lambda_{min}[Q_1(\alpha)]} + \frac{d}{\lambda_{min}[Q_2(\alpha)]} < \frac{c_2 e^{-\beta T}}{\lambda_{max}[Q_1(\alpha)]}, \tag{10}$$

in which $\tilde{Q}_1(\alpha) = R^{-1/2}Q_1(\alpha)R^{-1/2}$, with R given by (7).

3. MAIN RESULTS

The following theorem presents sufficient conditions, in terms of a parameter-dependent LMI, for the synthesis of matrices that solves Problem 1.

Theorem 1. Given a LPV continuous-time system (2), parameters $(c_1, c_2, d, T, R_p, R_e)$ and a fixed scalar parameter β, if, for each $\alpha \in \Delta_N$, there exist symmetric positive definite matrices $K \in \mathbb{R}^{n\times n}$, $W(\alpha) \in \mathbb{R}^{r\times r}$ and $Z(\alpha) \in \mathbb{R}^{n\times n}$; matrices $L(\alpha) \in \mathbb{R}^{n\times q}$, $M(\alpha) \in \mathbb{R}^{n\times n}$ and $F(\alpha) \in \mathbb{R}^{p\times n}$ and positive real scalars μ_1, μ_2 and μ_3, such that

$$\begin{bmatrix} \mathscr{M}_{11}(\alpha) & \mathscr{M}_{12}(\alpha) & KB(\alpha)+L(\alpha)D(\alpha) \\ (\star) & \mathscr{M}_{22}(\alpha) & Z(\alpha)B(\alpha) \\ (\star) & (\star) & -\beta W(\alpha) \end{bmatrix} < \mathbf{0},$$
$$\mathscr{M}_{11}(\alpha) = -\beta K - M(\alpha) - M'(\alpha),$$
$$\mathscr{M}_{12}(\alpha) = KA(\alpha) + L(\alpha)C_y(\alpha) + M(\alpha),$$
$$A'(\alpha)Z(\alpha) + Z(\alpha)A(\alpha) - \beta Z(\alpha), \tag{11a}$$

$$c_1\mu_1 + d\mu_3 < c_2 e^{-\beta T}\mu_2, \tag{11b}$$

$$W(\alpha) < \mu_3 \mathbf{I}, \tag{11c}$$

$$\mu_2 R_p < Z(\alpha) < \mu_1 R_p, \tag{11d}$$

$$\mu_2 R_e < K < \mu_1 R_e, \tag{11e}$$

$$\begin{bmatrix} K & \mathbf{0} & F'(\alpha) \\ (\star) & Z(\alpha) & C_z'(\alpha) - F'(\alpha) \\ (\star) & (\star) & \mathbf{I} \end{bmatrix} > \mathbf{0}, \tag{11f}$$

then for a sufficiently slow varying parameter α there exists a filter in the form (4), such that the augmented system (5) is FTB with respect to $(c_1, c_2, \mathscr{W}_d, T, R_p, R_e)$ and the filter is also Ω-bounded in finite time T for $\Omega = \Gamma\tilde{Q}_1\Gamma'$, $\Gamma = diag(\mathbf{I}, \Gamma_{22})$, with Γ_{22} non-singular. A realization of the filter is given by the matrices:

$$\begin{aligned} A_f(\alpha) &= -K^{-1}M(\alpha), \\ B_f(\alpha) &= -K^{-1}L(\alpha), \\ C_f(\alpha) &= F(\alpha)\Gamma_{22}^{-1}. \end{aligned} \tag{12}$$

Proof. As presented in Chilali and Gahinet (1996) in the context of pole placement, consider the partitioned matrices

$$\tilde{Q}_1(\alpha) = \begin{bmatrix} X(\alpha) & U'(\alpha) \\ U(\alpha) & \hat{X}(\alpha) \end{bmatrix}, \tilde{Q}_1^{-1}(\alpha) = \begin{bmatrix} Y(\alpha) & V'(\alpha) \\ V(\alpha) & \hat{Y}(\alpha) \end{bmatrix},$$
$$H(\alpha) = \begin{bmatrix} Y(\alpha) & \mathbf{I} \\ V(\alpha) & \mathbf{0} \end{bmatrix},$$

together with the following change of variables

$$M(\alpha) = -KA_f(\alpha)U(\alpha)Z(\alpha), \tag{13a}$$

$$L(\alpha) = -KB_f(\alpha), \tag{13b}$$

$$F(\alpha) = C_f(\alpha)\Gamma_{22}U(\alpha)Z(\alpha), \tag{13c}$$

$$W(\alpha) = Q_2^{-1}(\alpha), \tag{13d}$$

where $X(\alpha)$, $Y(\alpha)$ and $Q_2^{-1}(\alpha)$ are chosen such that

$$Z(\alpha) = X^{-1}(\alpha),$$
$$W(\alpha) = Q_2^{-1}(\alpha),$$
$$K = Y(\alpha) - Z(\alpha).$$

By multiplying the LMI (11a) on the left by $\bar{H}'(\alpha)$ and on the right by $\bar{H}(\alpha)$, and multiplying the result on the left by $\tilde{H}'(\alpha)$ and on the right by $\tilde{H}(\alpha)$, with

$$\bar{H}(\alpha) = \begin{bmatrix} N(\alpha) & \mathbf{0} \\ \star & \mathbf{I} \end{bmatrix}, \tilde{H}(\alpha) = \begin{bmatrix} H^{-1}(\alpha) & \mathbf{0} \\ \star & \mathbf{I} \end{bmatrix},$$
$$N(\alpha) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & X(\alpha) \end{bmatrix},$$

the LMI (9) is obtained. Moreover, it is easy to see that LMI (10) is satisfied if the conditions

$$c_1\mu_1 + d\mu_3 < c_2 e^{-\beta T}\mu_2, \tag{14}$$

$$Q_2^{-1}(\alpha) < \mu_3 \mathbf{I}, \tag{15}$$

$$\mu_2 \mathbf{I} < Q_1^{-1}(\alpha) < \mu_1 \mathbf{I}, \tag{16}$$

are guaranteed.

Inequalities (14) and (15) are LMIs (11b) and (11c), respectively. By multiplying inequality (16) on the left and on the right by $R^{1/2}$, with R given by (7), one has

$$\mu_2 R < \tilde{Q}_1^{-1}(\alpha) < \mu_1 R. \tag{17}$$

Knowing that the identity $\tilde{Q}_1(\alpha)\tilde{Q}_1^{-1}(\alpha) = \mathbf{I}$ gives the equations

$$X(\alpha)Y(\alpha) + U'(\alpha)V(\alpha) = \mathbf{I},$$
$$X(\alpha)V'(\alpha) + U'(\alpha)\hat{Y}(\alpha) = \mathbf{0},$$

one has for $U(\alpha) = X(\alpha)$ that

$$V(\alpha) = -K,$$
$$\hat{Y}(\alpha) = K.$$

Consequently, inequality (17) is satisfied if, and only if

$$\mu_2 R < \begin{bmatrix} K + Z(\alpha) & -K \\ -K & K \end{bmatrix} < \mu_1 R. \tag{18}$$

Left-multiplying LMI (18) by G' and right-multiplying by G, with

$$G = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix}$$

one can see that inequality (18) is equivalent to LMIs (11d) and (11e). At last, by multiplying LMI (11f) on the left by $\bar{H}'(\alpha)$ and on the right by $\bar{H}(\alpha)$, multiplying the result on the left by $\tilde{J}'(\alpha)$ and on the right by $\tilde{J}(\alpha)$, with

$$\tilde{J}(\alpha) = \begin{bmatrix} J^{-1}(\alpha)\Gamma_{22}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, J(\alpha) = \begin{bmatrix} \mathbf{I} & X(\alpha) \\ \mathbf{0} & X(\alpha) \end{bmatrix}$$

and then applying Schur complement in the resulting matrix, one has

$$e'(t)e(t) < \varsigma'(t)\Omega^{-1}\varsigma(t),$$

which guarantees the constraint in the estimation error.

By the choice of $U(\alpha)$, one has that $U(\alpha)Z(\alpha) = \mathbf{I}$ and that the filter matrices $A_f(\alpha)$, $B_f(\alpha)$ and $C_f(\alpha)$ can be recovered from the change of variables in (13).

Remark 2. It should be remarked that if β is not fixed, the proposed conditions are not longer linear for a fixed α. Actually, they are not even bilinear due to (11b). Aside from that, a binary search for a suitable β can be guided by balancing (11a) and (11b) and should not be a computational burden since β is just a scalar variable.

In the definition of Problem 1, Ω is a design parameter that must be adjusted for a suitable weighting between the size of output estimation error and the size of the filter and plant states. It should be observed that there is no loss of generality to write $\Omega = \Gamma\tilde{Q}_1\Gamma'$ and to consider the parameter Γ as invertible (Ω, Γ, $\tilde{Q}_1$ are invertible). The choice of $\Gamma = \mathrm{diag}(\mathbf{I}, \Gamma_{22})$ allows to directly adjust the matrix C_f with a scale factor given by Γ_{22}. Consequently, the quality of the realization $\{A_f, B_f, C_f\}$ of the filter can be improved without deteriorating the estimate error.

Theorem 1 leads to a LMI feasibility problem that must be satisfied for all parameters $\alpha \in \Delta_N$. Although this is an infinite dimension problem in the parameter α, the fact that it lies in the unit N-simplex can be used to find sufficient LMI conditions written only in terms of the vertices of the polytope (Bliman et al., 2006).

In fact, using the relaxation proposed in Oliveira and Peres (2007) one can write the parameter-dependent LMIs in Theorem 1 as LMIs that are independent of the parameter α. As the level of relaxation increases, it is possible to achieve less conservative sets of conditions and tending to necessary and sufficient conditions.

For this purpose, the next Definition 3 generalizes the linear dependence on the parameter α to a homogeneous polynomial dependence.

Definition 3. A matrix $M(\alpha)$ is homogeneous polynomially parameter-dependent (HPPD) on $\alpha \in \Delta_N$ with degree g if it can be expressed as

$$M(\alpha) = \sum_{k \in \mathscr{S}_g} \alpha_1^{k_1} \alpha_2^{k_2} \cdots \alpha_N^{k_N} M_k, \tag{19}$$

with

$$\mathscr{S}_g = \left\{ k \in \left(\mathbb{Z}_+\right)^N : \sum_{i=1}^{N} k_i = g \right\}.$$

M_k are the matrices coefficients of the monomials of $M(\alpha)$, where

$$\mathrm{card}\left(\mathscr{S}_g\right) = \frac{(N+g-1)!}{g!\,(N-1)!}.$$

The set of HPPD of $\alpha \in \Delta_N$ with degree g matrices is denoted by $\mathbb{H}_{(g)}$ and the subset corresponding to the matrices with order $m \times n$ is denoted by $\mathbb{H}_{(g)}^{m\times n}$.

The relaxation proposed in Oliveira and Peres (2007) can be used in the parameter-dependent LMIs of Theorem 1 by forcing a homogeneous polynomial structure in the LMI variables $L(\alpha)$, $M(\alpha)$ and $F(\alpha)$, turning them into HPPD matrices. The relaxed condition is given by the LMIs stemming from the matrices coefficients of the HPPD matrices, and the LMI variables are the matrix coefficients of the monomials of the HPPD matrices. Whilst this procedure is systematic, it can become very complex as the degree g of the HPPD matrices increases. Nevertheless, the specialized parser ROLMIP [1] can be used to automatically carry this relaxation (Agulhari et al., 2012).

[1] Available for download at `http://www.dt.fee.unicamp.br/~agulhari/rolmip/rolmip.htm`.

It is important to note that for the particular case that $g = 0$, the recuperation of the filter from the LMI variables using (12) is free from the parameter α, and consequently, the assumption that α can be read in real time is no longer required; the filter is robust in the sense that the parameter can be considered uncertain. The reason why is used degrees greater than zero is that a sequence of less conservative LMI relaxations may be obtained in the conditions of Theorem 1 by increasing the degree g, as will be clearer in the next theorems.

Theorem 2. For given $\bar{g}$ and $\bar{\mu}_2$, let $c_2^*(\bar{g})$ be the optimal solution of

$$\begin{aligned} &\min \quad c_2 \\ &\text{such that (11) holds with } g = \bar{g} \text{ and } \mu_2 = \bar{\mu}_2. \end{aligned}$$

Then, $c_2^*(\bar{g}+1) \leq c_2^*(\bar{g})$.

Proof. If there exist scalars μ_1,$\bar{\mu}_2$ and μ_3; matrix $K \in \mathbb{R}^{n\times n}$; and matrices W, Z, L, M, $F \in \mathbb{H}_{(g)}$ such that (11) holds, then μ_1, $\bar{\mu}_2$, μ_3, K and the following matrices

$$\left(\sum_{i=1}^{N}\alpha_i\right)W, \left(\sum_{i=1}^{N}\alpha_i\right)Z, \left(\sum_{i=1}^{N}\alpha_i\right)L, \left(\sum_{i=1}^{N}\alpha_i\right)M, \left(\sum_{i=1}^{N}\alpha_i\right)F,$$

belonging to $\mathbb{H}_{(g+1)}$, are a particular solution to (11), since $\alpha \in \Delta_N$. Hence, the minimization of c_2 subject to (11) for $\bar{g}+1$ produces at least the same optimal value obtained with $\bar{g}$, which implies that $c_2^*(\bar{g}+1) \leq c_2^*(\bar{g})$.

Theorem 3. For given $\bar{g}$, $\bar{\mu}_1$ and $\bar{\mu}_3$, let $c_1^*(\bar{g})$ and $d^*(\bar{g})$ be the optimal solution of problems

$$\begin{aligned} &\max \quad c_1 \\ &\text{such that (11) holds with } g = \bar{g} \text{ and } \mu_1 = \bar{\mu}_1, \end{aligned}$$

$$\begin{aligned} &\max \quad d \\ &\text{such that (11) holds with } g = \bar{g} \text{ and } \mu_3 = \bar{\mu}_3, \end{aligned}$$

respectively. Then $c_1^*(\bar{g}) \leq c_1^*(\bar{g}+1)$ and $d^*(\bar{g}) \leq d(\bar{g}+1)$.

Proof. Similar to Theorem 2.

The optimization problems in Theorem 2 and Theorem 3 can be seen as optimum filtering problems within FTB context. For example, to design a filter which rejects the maximum possible types of disturbances, one may try to maximize d.

The computational complexity of the LMIs is estimated by the number of scalar variables V and the number of LMI scalar rows L. For Theorem (1),

$$V = n(p+q+n)\,\mathrm{card}\left(\mathscr{S}_g\right) + n(n+1) + \frac{q(q+1)}{2} + 3, \tag{20}$$

$$L = (2n+r)\,\mathrm{card}\left(\mathscr{S}_{g+f+1}\right) + (4n+p+r)\,\mathrm{card}\left(\mathscr{S}_{g+f}\right) + n + 1. \tag{21}$$

By increasing the degree g, the number of decision variables is also increased and in consequence, the complexity of the LMIs also raises. However, by using an extension of Pólya's theorem (Oliveira and Peres, 2005, 2007), and based on the fact that the time-varying parameters α belong to the unit N-simplex, the conditions of Theorem (1) may also be improved using a sufficiently large positive integer f with no increase in the number of variables for a given degree g by multiplying the LMIs (11) by the factor $\left(\sum_{i=1}^{N}\alpha_i\right)^f$.

Table 1. Minimum upper bounds of c_2 and maximum lower bounds of c_1 and d for different values of g and f.

Degree g	Index f	c_2	c_1	d
0	0	3.5381	-	0.5027
1	0	3.2441	-	0.5307
1	1	3.1724	0.0042	0.5307
1	2	3.1720	0.0042	0.5307
2	0	3.1511	0.0118	0.5308
2	1	3.0856	0.0164	0.5308
2	2	3.0856	0.0164	0.5308
3	0	3.0856	0.0164	0.5308
3	1	3.0656	0.0175	0.5309

4. NUMERICAL EXAMPLE

The numerical example was performed using the solver SeDuMi (Sturm, 1999) and the parsers YALMIP (Löfberg, 2004) and ROLMIP (Agulhari et al., 2012) within Matlab environment.

Consider the system (2) with matrices in polytope (3) with the following vertices

$$A_1 = \begin{bmatrix} -1.0 & 2.0 \\ -3.0 & -2.0 \end{bmatrix}, A_2 = \begin{bmatrix} -1.0 & 2.0 \\ -3.0 & -1.0 \end{bmatrix},$$
$$A_3 = \begin{bmatrix} -2.0 & 2.0 \\ -3.0 & -2.0 \end{bmatrix}, A_4 = \begin{bmatrix} -2.0 & 2.0 \\ -3.0 & -1.0 \end{bmatrix},$$
$$B_1 = \begin{bmatrix} -0.5 \\ 0.1 \end{bmatrix}, B_2 = \begin{bmatrix} -0.1 \\ 0.1 \end{bmatrix}, B_3 = \begin{bmatrix} -0.5 \\ 0.5 \end{bmatrix}, B_4 = \begin{bmatrix} -0.1 \\ 0.5 \end{bmatrix},$$
$$C_{y1} = [1.0 \ 0.5], C_{y2} = [1.2 \ 0.5],$$
$$C_{y3} = [1.0 \ 0.6], C_{y4} = [1.2 \ 0.6],$$
$$C_{z1} = [0.6 \ 1.0], C_{z2} = [1.0 \ 1.0],$$
$$C_{z3} = [0.6 \ 1.2], C_{z4} = [1.0 \ 1.2],$$
$$D_1 = D_2 = 0.1, D_3 = D_4 = 0.2$$

and the slowly varying parameter

$$\alpha(t) = \left(\tfrac{1}{2}\sin^2(\omega t), \tfrac{1}{2}\sin^2(\omega t), \tfrac{1}{2}\cos^2(\omega t), \tfrac{1}{2}\cos^2(\omega t)\right), \quad (22)$$

with ω sufficiently small. It is easy to check that (22) belongs to Δ_4 for all $t \geq 0$.

Theorems 2 and 3 along with Pólya's relaxation are applied with $\bar{\mu}_1 = 1$, $\bar{\mu}_2 = 1$, $\bar{\mu}_3 = 1$ and $\beta = 0.6$ in order to investigate the effect of increasing g and f in the search of minimum upper bounds of c_2 attained by the conditions of Theorem 2 and also in the search of maximum lower bounds of c_1 and d attained by the conditions of Theorem 3. The chosen FTB parameters for this example are $c_1 = 0.1$, $c_2 = 2$, $d = 1$, $T = 1.5$, $R_e = 4\mathbf{I}$ and $R_p = 4\mathbf{I}$. The results of the optimization problem are summarized in Table 1 (when the parameter is the value being optimized the corresponding value of the parameter should be ignored).

As can be seen in Table 1, the conditions of Theorem 3 are not able to provide a robust filter nor a LPV filter with a $c_1 > 0$. Also, by using degree $g = 3$ and $f = 1$ it was possible to obtain an upper bound to c_2 approximately 13.35% smaller than the robust filter corresponding to $g = 0$ and $f = 0$. Finally, as illustrated by the maximum lower bounds obtained by d, it may happen that the gain obtained increasing g and f is not appreciable. In this case, it would be better use the robust filter corresponding to $g = 0$ if existent.

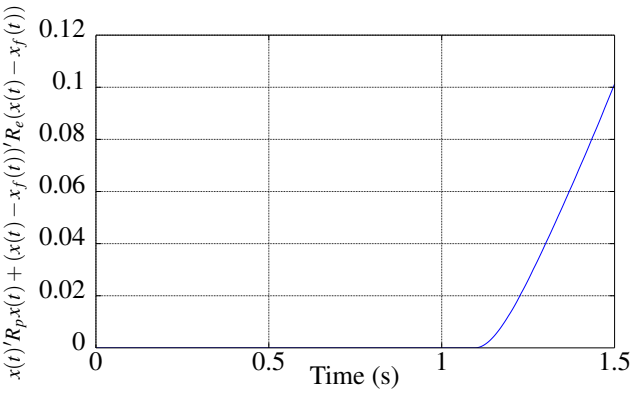


Figure 1. Sum of the weighted quadratic norm of the states of the plant and observation error.

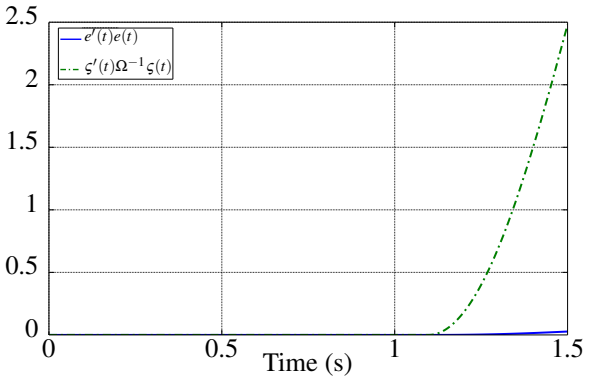


Figure 2. Time simulation comparing the norm of the error with the bound imposed by Ω.

Consider now system (2) with the varying parameter (22) with $\omega = 0.001$. We wish to check if the FTB filter obtained by Theorem 2 with $\bar{\mu}_2 = 1$, $g = 3$ and $f = 1$ satisfies the FTB condition with respect to $(0.1, 3.06, \mathscr{W}_1, 1.5, 4\mathbf{I}, 4\mathbf{I})$ and the Ω-bound condition defined in (8) considering a disturbance given by the step function

$$w(t) = \begin{cases} 0, & \text{if } t \leq 1.1s, \\ 0.9, & \text{if } t > 1.1s, \end{cases}$$

which represents the worst type of signal belonging to the class $\mathscr{W}_1$. Since this filter is obtained using the assumption that α is a slowly varying parameter, one must verify if $\dot{\alpha}$ is really sufficiently small by time-domain simulations. Considering zero initial conditions, a time-simulation was performed in the time interval $t \in [0, 1.5s]$.

As shown in Figure 1, the designed filter satisfies the FTB condition and

$$\max_{t\in[0,1.5s]} \left\{ \begin{bmatrix} x(t) \\ x(t) - x_f(t) \end{bmatrix}' \begin{bmatrix} R_p & \mathbf{0} \\ \mathbf{0} & R_e \end{bmatrix} \begin{bmatrix} x(t) \\ x(t) - x_f(t) \end{bmatrix} \right\} = 0.06,$$

which is approximately 2% of the value of $c_2 = 3.0653$. Moreover, as can be seen in Figure 2, the estimation error of the filter also satisfies the Ω-bound condition defined in (8). The tracking error is shown in Figure 3. Although the choice of Γ_{22} did not nullify the error between z and z_f, it ensured a maximum error of 0.0764. The difficulty of having a null estimation error in a finite time horizon is due to the small time that the filter has to dynamically estimate the output z.

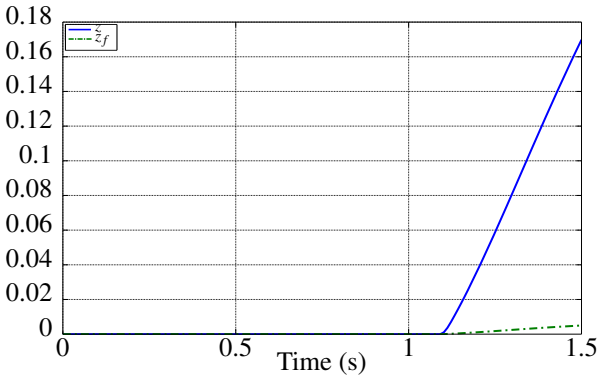


Figure 3. Time simulation of the outputs z and z_f.

5. CONCLUSION

This paper considers the problem of filter design for LPV continuous-time systems. The filter was obtained under the assumption that the parameter of the LPV system is sufficiently slow time-varying. If this assumption is satisfied the obtained filter guarantees that the augmented system is bounded during a finite time horizon under the presence of bounded disturbances. The design conditions are represented by a LMI feasibility problem, which can be relaxed via homogeneous polynomials techniques and Pólya's theorem. It was shown that a sequence of less conservative conditions may be obtained by increasing the degree g of the HPPD matrices or increasing the positive integer f based on Pólya's theorem. To handle the algebraic manipulation involved in the construction of the relaxation of the parameter-dependent LMIs, the specialized parser ROLMIP was used, easing the work.

REFERENCES

Agulhari, C.M., de Oliveira, R.C.L.F., and Peres, P.L.D. (2012). Robust LMI Parser: a computational package to construct LMI conditions for uncertain systems. In *XIX Brazilian Conference on Automation (CBA 2012)*, 2298–2305. Campina Grande, PB, Brazil.

Amato, F., Ariola, M., and Cosentino, C. (2005). Finite-time control of linear time-varying systems via output feedback. In *Proceedings of the 2005 American Control Conference*, 4722–4726. Portland, OR, USA.

Amato, F., Ariola, M., Cosentino, C., Abdallah, C.T., and Dorato, P. (2003). Necessary and sufficient conditions for finite-time stability of linear systems. In *Proceedings of the 2003 American Control Conference*, volume 5, 4452 – 4456.

Amato, F., Ariola, M., and Dorato, P. (2001). Finite-time control of linear systems subject to parametric uncertainties and disturbances. *Automatica*, 37(9), 1459–1563.

Amato, F., Ambrosino, R., Ariola, M., Cosentino, C., and De Tommasi, G. (2014). *Finite-Time Stability and Control*. Springer, London, UK.

Baslamisli, S.Ç., Köse, İ.E., and Anlaş, G. (2009). Gain-scheduled integrated active steering and differential control for vehicle handling improvement. *Vehicle System Dynamics*, 47(1), 99–119.

Bliman, P.A., Oliveira, R.C.L.F., Montagner, V.F., and Peres, P.L.D. (2006). Existence of homogeneous polynomial solutions for parameter-dependent linear matrix inequalities with parameters in the simplex. In *Proceedings of the 45th IEEE Conference on Decision and Control*, 1486–1491. San Diego, CA, USA.

Borges, R.A., Ishihara, J.Y., Rocha, I.G., Risso, L.O., and Kussaba, H.T.M. (2013). Finite time robust filtering for time-varying uncertain polytopic linear systems. In *Proceedings of the 2013 European Control Conference*, 1854–1859. Zurich, Switzerland.

Chilali, M. and Gahinet, P. (1996). $\mathscr{H}_\infty$ design with pole placement constraints: an LMI approach. *IEEE Transactions on Automatic Control*, 41(3), 358–367.

Ganguli, S., Marcos, A., and Balas, G. (2002). Reconfigurable LPV control design for Boeing 747-100/200 longitudinal axis. In *Proceedings of the 2002 American Control Conference*, volume 5, 3612–3617. IEEE.

He, S. and Liu, F. (2011). Robust L_2-L_∞ filtering of time-delay jump systems with respect to the finite-time interval. *Mathematical Problems in Engineering*. `http://www.hindawi.com/journals/mpe/2011/839648/`.

Khalil, H.K. (2002). *Nonlinear Systems*. Prentice Hall, Upper Saddle River, NJ, 3rd edition.

Lin, H.H., Beck, C., and Bloom, M. (2008). Multivariable LPV control of anesthesia delivery during surgery. In *Proceedings of the 2008 American Control Conference*, 825–831. IEEE.

Liu, C., Zhang, Y., and Sun, H. (2012). Finite-time H_∞ filtering for singular stochastic systems. *Journal of Applied Mathematics*. `http://www.hindawi.com/journals/jam/2012/615790/`.

Löfberg, J. (2004). YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the 2004 IEEE International Symposium on Computer Aided Control Systems Design*, 284–289. Taipei, Taiwan. `http://control.ee.ethz.ch/~joloef/yalmip.php`.

Luan, X., Liu, F., and Shi, P. (2010). Finite-time filtering for non-linear stochastic systems with partially known transition jump rates. *IET Control Theory & Applications*, 4(5), 735–745.

Oliveira, R.C.L.F. and Peres, P.L.D. (2005). Stability of polytopes of matrices via affine parameter-dependent Lyapunov functions: Asymptotically exact LMI conditions. *Linear Algebra and Its Applications*, 405, 209–228.

Oliveira, R.C.L.F. and Peres, P.L.D. (2007). Parameter-dependent LMIs in robust analysis: Characterization of homogeneous polynomially parameter-dependent solutions via LMI relaxations. *IEEE Transactions on Automatic Control*, 52(7), 1334–1340.

Peña, R.S.S. and Ghersin, A.S. (2010). LPV control of glucose for diabetes type I. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, 680–683. IEEE.

Shamma, J.S. (2012). An overview of LPV systems. In *Control of Linear Parameter Varying Systems with Applications*, 3–26. Springer.

Sturm, J.F. (1999). Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1–4), 625–653. `http://sedumi.ie.lehigh.edu/`.

Toth, R. (2010). *Modeling and Identification of Linear Parameter-Varying Systems*. Lecture Notes in Control and Information Sciences. Springer.

Percolation Theory Approach to Transient Stability Analysis of Power Systems

Alejandra Machuca-Moreno * **Eduardo Mojica-Nava** **

* *Universidad Nacional de Colombia, Bogota, Colombia. (e-mail:lamachucam@unal.edu.co)*
** *Universidad Nacional de Colombia, Bogota, Colombia. (e-mail: eamojican@unal.edu.co)*

Abstract: This paper proposes a methodology for the stability assessment of a power grid based on percolation theory. Once it is proved that the rotor angle stability depends on the initial condition and the topological properties of the network, tools provided by complex network and percolation theory are used to identify the most vulnerable buses from a transient stability perspective in order to locate the best places to install control to prevent the spread of lack of synchronism through the grid. Finally, the proposed approach is tested on the transmission network system of IEEE 14.

Keywords: Smart Power Grid, Transient Stability, Complex Networks, Percolation Theory.

1. INTRODUCTION

Power system is the set of elements that generate, transmit and distribute electrical energy to consuming agents such as residential buildings, factories, street lighting, and so on. Most of power generators are electromechanical systems, they use the energy of water or fossil fuels to generate mechanical energy, which is then converted into electrical energy (A.R. Bergen (1999)). Normal operating conditions or *steady state* satisfy the following conditions:

- All machines rotate at the same velocity and they are never accelerated.
- The voltage and currents generated have the same frequency.

The first condition is called *frequency stability* while the second is known as *voltage stability*. When all generators rotate at the same velocity, the relative difference between their rotor angles remains constant. *Rotor angle stability* is the ability of the power system to maintain this synchronism.
Transient Stability is the name given to the maintenance of rotor angle stability when the system is subjected to large disturbances (Kundur et al. (2004)).

There are several methods for transient stability analysis of power systems such as:

- Automatic Learning: Based on machine learning techniques.(Wehenkel (1998))
- Direct methods: Based on obtaining Lyapunov functions for simple models of the systems. (Varaiya et al. (1985))
- Simulations in time domain: Solving differential equations using numerical methods. (M.Pavella (1994))

In practical situations, the most common way to check transient stability is running vast time-domain simulations for those that are considered the important fault scenarios. However, power systems are large scale systems, and power engineers need to predict those scenarios based on guesses; sometimes, criteria like bigger capacity are used. However, they are still guesswork made by humans, and are prone to mistakes. (M.Pavella (1994))

Currently, the power grid is continuously growing thanks to new uses of electrical energy and demand vegetative growth. Also, the envisioned future power generation will rely on renewable energy sources and they are highly stochastic, there will be an increasing number of transient disturbances acting on the power system. Considering these future conditions, it is going to be more difficult to guess the important fault scenarios.

This paper proposes an approach of transient stability analysis in the power grid based on topological information by identifying the most vulnerable generators of the grid using percolation theory. The main objective of identifying the vulnerable nodes is to place controls on them in order to prevent the loss of synchronism in the other generators of the grid.

2. REVIEW OF TRANSIENT STABILITY ANALYSIS

The equation that governs rotor motion of a synchronous machine is known as the swing equation, and is based on the elementary principle in dynamics which states that accelerating torque is the product of the moment of inertia of the rotor times its angular acceleration. It can be written for the synchronous generator in the form

$$\frac{H}{180f}\frac{d^2\delta}{dt^2} = P_a = P_m - P_e - D\frac{d\delta}{dt}, \tag{1}$$

where H is the inertia constant of the machine, f is the frequency of the system, δ is the rotor angle, P_a is the accelerating power, P_m is the mechanical power, P_e is the

electrical power and D is the damping coefficent of the machine (J.Machoswki (2008)).

Constants coefficients H and D depend on the size and capability of the machine, f is a constraint of the system, and P_m depends on the external mechanical power (water, oil, etc.), and for this analysis it is taken as a constant.

This equation shows that every time the mechanical and the electrical power are unequal the second derivative of the rotor angle is proportional to that difference; meaning that there is an acceleration in the machine.

Considering P_m as a constant, the main variable to analyze is the electric power; for simplified models this power is described as

$$P_e = P_{max} \cdot sin\delta = \frac{|E_1'||E_2'|}{X} \cdot sin\delta \tag{2}$$

For (2) the generator is supplying power through a transmission system. Voltage $|E_1'|$ is the transient internal voltage of the generator and $|E_2'|$ is the voltage at the receiving end and X is the transfer reactance between $|E_1'|$ and $|E_2'|$ (J.Grainger (1994)).

Most systems have more than one synchronous machine; for a system with n generators, the active output of the i^{th} generator is given by

$$\begin{aligned} P_e &= Re\{E_i I_i^*\} \\ &= Re\left\{ E_i \sum_{j=1}^{n} Y_{ij}^* E_j^* \right\} \\ &= E_i^2 G_{ii} + \sum_{j=1}^{n} E_i E_j \left[B_{ij} sin(\delta_i - \delta_j) + G_{ij} cos(\delta_i - \delta_j) \right] \end{aligned} \tag{3}$$

$|E_i|$ and $|E_j|$ are the internal transient voltages, and B_{ij} and G_{ij} are the susceptance and conductance between generators i and j (P.Kundur (1994)).

Equation (3) demonstrates that rotor angle stability depends highly on the initial conditions of the system and its transfer admittance, which is heavily reliant on the topological structure of the power network.

3. PERCOLATION THEORY AND NETWORK RESILIENCE

Percolation theory describes the behavior of connected clusters in a graph, in order to understand this concept assume that some liquid is poured on top of a porous material, and wonder if the liquid will be able to reach the bottom of it (Broadbent and Hammersley (1957)). Percolation is a pervasive concept that arises not only from atomic and molecular solids in physics, it also comes from social, technological, and natural systems (Strogatz (2001); ben Avraham and Glasser (2007); S et al. (2000)), information diffusion in online social networks (Barrat et al. (2008)), resistor networks (Kirkpatrick (1973)), forest fires (Drossel and Schwabl (1992)), among others.

Consider a network whose edges are present with probability p; percolation studies the emergence of paths that percolate through the grid. For great values of p a lot edges are present, then big clusters of nodes connected by edges can form. The creation of a giant component during the percolation process is known as a percolation transition (Albert and lszl Barabsi (2002)).

Imagine taking a network and removing some fraction of its vertices or edges, what effect this will have on the performance of the network?, this is a practical interest that Percolation Theory can help us to analyze. It can help us to understand the macroscopic behavior of networks in relation to the microscopic states of its components (Newman (2010)).

4. TRANSIENT STABILITY APPROACH FROM A PERCOLATION THEORY PERSPECTIVE

First, the power grid must by modelled as a complex network. The regional transmission system was simulated but the local distribution system will be excluded from the analysis. The vertices of a complex network represent the generation centrals, and load or switching substations; meanwhile, the edges are modelled as the high voltage transmission lines. The topology is not hard to determine, a power network has a spatial and geographic interpretation, and its the same given to its graph model.

The admittance matrix of a power network has the same formulation of the Laplacian of a graph: the diagonal elements of the admittance matrix are equal to the degree matrix and the non-diagonal elements are the same to the adjacency matrix.

For a graph $\mathcal{G}$ formally defined as $\mathcal{G} = (V, E, \omega)$, where V is a finite set of vertices, E is a set of edges, and ω is the weight between vertices (Mesbahi and Egerstedt (2010)). The Laplacian $L(\mathcal{G}_w)$ is defined as

$$[\Delta(\mathcal{G}_w)]_{ii} = \sum_{\{j|(v_j,v_i)\ \epsilon\ E(\mathcal{G})\}} w_{ij} \tag{4}$$

$$[A(\mathcal{G}_w)]_{ij} = \begin{cases} w_{ij} & \text{if } v_i v_j \ \epsilon\ E(\mathcal{G}) \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

$$L(\mathcal{G}_w) = \Delta(\mathcal{G}_w) - A(\mathcal{G}_w) \tag{6}$$

This paper considers the following considerations for the modelling of the power grid.

- It only considers the high voltage system and it is assumed balanced.
- The impedances between bus and neutral are neglected.
- All transmission line and transformers are modelled as weighted edges, the weight is equal to the impedance between nodes.
- All parallel lines are modelled as an equivalent single line.
- Shunt impedances of lines are considered.
- Loads are modelled as constant admittances.
- All generator have the same constant of inertia (13,08 s) and damping constant (2 s).

This last consideration might seem mistaken, but, to analyse the effect of topological properties of the network on its transient stability, this two variables must be uniform in the grid.

Figures 1 shows the physical topology graph of IEEE 14 and test case.

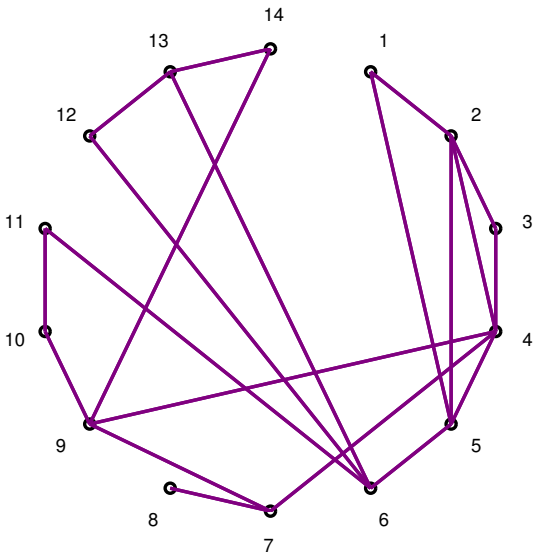


Fig. 1. Modeled graph of IEEE-14 bus test case

For the transient stability simulations it is used the Power System Analysis Toolbox - PSAT. For this specific case, all faults are balanced three phase faults on the buses, not on the high voltage lines; which means there are not changes of topology before or after the fault. In addition, all faults had a duration of 2 seconds.

For each system, a fault is simulated on every bus associated to a generator, as expected the generator suffered some kind of perturbation not only in its rotor angle but also in its frequency. Because of the duration of the fault, other synchronous machines begin to suffer this disturbances. It means every time a generator losses synchronism this perturbation percolates through the grid to reach other machines; creating a cluster of unsynchronised units. This process might be stopped or retarded by placing a control in the most vulnerable generator. Figures 2 and 3 show the frequency variation of each generator of the system for a Fault in machines 1 and 6 respectively.

Acording to the Colombian Network Code (re Regulacion de Energia y Gas CREG (1995)), all generator units must have a setting of relay tripping for low or high frequency every time that variable gets out of the range between 57.5 - 63 Hz. Then, when the frequency of the generator is 10% to the way on this limits (59.75 Hz - 60.3 Hz), the generator counts as a part of the cluster.

Figure 4 shows the size of the cluster formed by the machines whose frequency is not on the speciied limits.

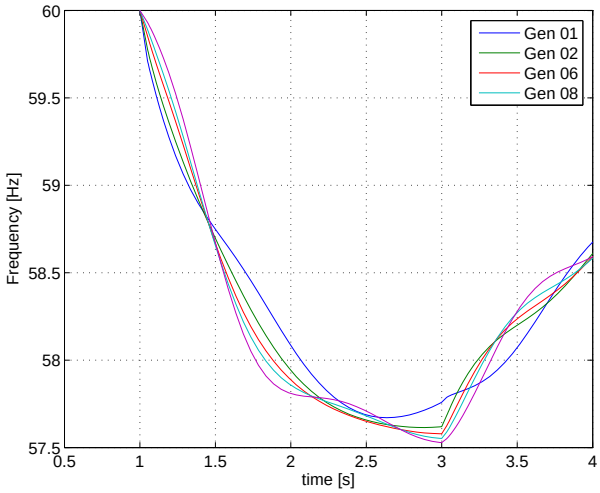


Fig. 2. Frequency variation due to fault on bus 1

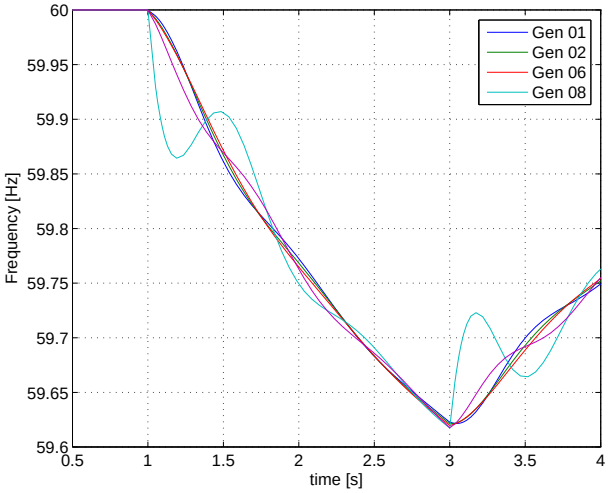


Fig. 3. Frequency variation due to fault on bus 6

When a fault is made in the bus associated to bus 1 or 2, it occurs a discrete percolation, all other generators seem to lost synchronism very fast. But, for machine 8 even when it lost synchronism right after the fault, there are several cycles before the others machines start to get out its state of synchronism. Finally, Generator 6 it starts to accelerate long afterwards the fault, but when it does all other generators begin to lost synchronism in which seems to be a delayed discrete percolation. For this test case, the bus 2 has been selected as the vulnerable node; Then, the same simulation is performed a second time with a overdamped generator 2; the damping coefficient for this machine is no longer 2, its 30 in order to emulate the controllers placed.

In Figure 5 Generator 1 and 2 dynamics does not seem to change, however, generator 6 never loss synchronism and machine 8 acceleration is some cycles delayed. The main objective of delay the percolation transition is been achieved. This gives some cycles to the smart grid to

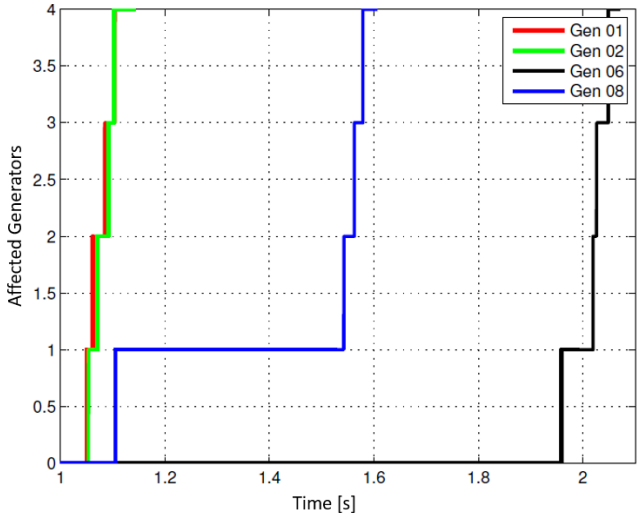


Fig. 4. Unsynchronized generators in IEEE-14 bus test case

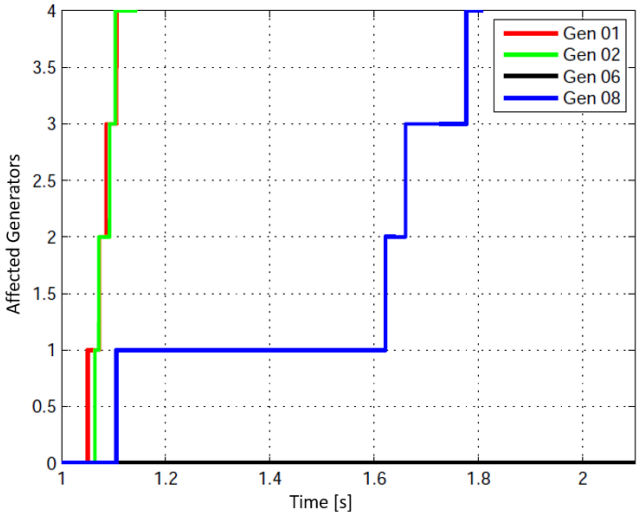


Fig. 5. Unsynchronized generators in IEEE-14 bus test case using an overdamping in machine 2

operate and prevent or decrease the potential damage of a blackout.

5. CONCLUSIONS

This paper provides an approach of network transient stability analysis using percolation and complex networks theory.

Dynamic equations demonstrates that transient stability relies heavily on the networks topological properties. In addition, the power grid characteristics show that it can be easily modelled as a complex network allowing us to use percolation theory analysis tools on it.

For a selected system, a balanced three phase fault was simulated in the generator and it was observed the frequency variation of all the generators in the network.

IEEE 14 bus system was tested and the results show how the lost of synchronism spreads through the grid; by identifying the bus associated to a generator which fault causes the most accelerated spreading and placing a control device on it can help the grid to delay and sometimes delete the percolation process improving the resilience of the grid.

REFERENCES

Albert, R. and lszl Barabsi, A. (2002). Statistical mechanics of complex networks. *Rev. Mod. Phys.*

A.R. Bergen, V. (1999). *Power Systems Analysis.* Tom Robbins, USA.

Barrat, A., Barthlemy, M., and Vespignani, A. (2008). *Dynamical Processes on Complex Networks.* Cambridge University Press, New York, NY, USA, 1st edition.

ben Avraham, D. and Glasser, M.L. (2007). Diffusion-limited one-species reactions in the bethe lattice. *Journal of Physics: Condensed Matter*, 19(6), 065107.

Broadbent, S.R. and Hammersley, J.M. (1957). Percolation processes. I. Crystals and Mazes. *Proceedings of the Cambridge Philosophical Society*, 53, 629–641.

Drossel, B. and Schwabl, F. (1992). Self-organized critical forest-fire model. *Phys. Rev. Lett.*, 69, 1629–1632.

J.Grainger, W. (1994). *Power System Analysis. United States of America.* McGraw-Hill Education, USA.

J.Machoswki, J.W.Bialek, J. (2008). *Power System Dynamics: Stability and Control.* John Wiley y Sons, United Kingdom.

Kirkpatrick, S. (1973). Percolation and conduction. *Rev. Mod. Phys.*, 45, 574–588.

Kundur, P., Paserba, J., Ajjarapu, V., Andersson, G., Bose, A., Canizares, C., Hatziargyriou, N., Hill, D., Stankovic, A., Taylor, C., Cutsem, T.V., and Vittal, V. (2004). Definition and classification of power system stability ieee/cigre joint task force on stability terms and definitions. *IEEE Transactions on Power Systems*, 19(3), 1387–1401.

Mesbahi, M. and Egerstedt, M. (2010). *Graph theoretic methods in multiagent networks.* Princeton series in applied mathematics. Princeton University Press, Princeton (N.J.).

M.Pavella, P. (1994). *Transient Stability of Power Systems: Theory and Practice.* John Wiley y Sons, United Kingdom.

Newman, M. (2010). *Networks: An Introduction.* Oxford University Press, Inc., New York, NY, USA.

P.Kundur (1994). *Power System Analysis. United States of America.* McGraw-Hill Education, USA.

re Regulacion de Energia y Gas CREG, C. (1995). *Res. 025 de 1995 - Codigo de Redes.* Ministerio de Minas y Energia - Republica de Colombia.

S, S., G, W., L., D.A., N, J., and D, S. (2000). Social percolation models. *PHYSICA. A*, 277, 239–247.

Strogatz, S.H. (2001). Exploring complex networks. *Nature*, 410(6825), 268–276.

Varaiya, P., Wu, F.F., and Chen, R.L. (1985). Direct methods for transient stability analysis of power systems: Recent results. *Proceedings of the IEEE*, 73(12), 1703–1715.

Wehenkel, L. (1998). *Automatic Learning Techniques in Power Systems.* Kluwer Academic Publishers Norwell, USA.

Performance evaluation of MPC for Waste Heat Recovery applications using organic Rankine cycle systems ⋆

Andres Hernandez [*,**] **Clara Ionescu** [*] **Sergei Gusev** [***]
Adriano Desideri [**] **Martijn van den Broek** [***] **Vincent Lemort** [**]
Robin De Keyser [*]

[*] *Department of Electrical energy, Systems and Automation, Ghent University, Belgium (e-mail: Andres.Hernandez@UGent.be).*
[**] *Thermodynamics laboratory, Energy system research unit, University of Liege, Belgium.*
[***] *Department of Flow, Heat and Combustion Mechanics, Ghent University, Belgium.*

Abstract: Organic Rankine Cycle (ORC) technology has demonstrated to be a suitable tool for recovering waste heat at low temperatures. The fluctuating nature of the waste heat source (varying temperature and mass flow) makes of waste heat recovery applications a challenging task. In this contribution Model Predictive Control (MPC) and more classical PID-like controllers are investigated, where special attention is paid to the analysis of the control performance for heat source profiles coming from different applications. A dynamic model of a real regenerative ORC unit equipped with a single screw expander developed in the Modelica language is used to test and compare the PID and MPC based control strategies. Results show that for low amplitude variations PID and MPC can perform equally good, but in case of large variations MPC is a more effective control strategy as it allows a safer and more efficient operation, operating close to the boundary conditions where production is maximized.

Keywords: Model Predictive control, Renewable energy systems, Process control, organic Rankine Cycle.

1. INTRODUCTION

Reducing the world-wide industrial energy consumption is a major concern in order to ensure guarantee a sustainable development. Despite all efforts to achieve a more efficient production, waste heat losses are still an important concern. An attractive technology able to recover heat at low temperatures is the Organic Rankine Cycle (ORC) system.

ORC power units stand out for their reliability and cost-effectiveness Verneau (1979),Angelino et al. (1984). Replacing the water by an organic compound opened new challenges, regarding the cycle design, selection of the fluid, modeling, simulation and control design Sun and Li. (2011); Colonna and Van Putten. (2007). Such thermodynamic units are designed to operate around certain steady-state conditions, however due to the highly fluctuating nature of the heat source, they are forced to operate at part-load conditions. Control design plays an essential role to enable a safe and optimal performance of the ORC unit. Safe operation is achieved by an accurate regulation of the superheating, since it is already recognized that low values for superheating maximize the cycle efficiency Hernandez et al. (2014) and avoid the formation of liquid droplets at expander inlet that can damage the expansion machine Wei et al. (2007).

Most of the current studies are restricted to guarantee safety conditions by regulating the superheating (see Grelet et al. (2015) and the references therein), but little attention has been paid to the performance of the power unit in terms of energy production. In order to maximize the output power the evaporating temperature is usually considered as the most relevant controlled variable Quoilin et al. (2011). In Feru et al. (2014), the modeling and control of a waste heat recovery system for a Euro-VI heavy-duty truck engine was achieved through the use of a switching model predictive control strategy to guarantee safe operation of the WHR system and to maximize output power. Also in the automotive field, the problem of maximizing the power produced by an ORC waste heat recovery system on board a diesel-electric railcar is tackled using dynamic real-time optimization Peralez et al. (2015). In Hernandez et al. (2015), an experimental study is conducted using an 11 kW_{el} pilot plant, showing that the constrained Model Predictive Control (MPC) outperforms PID based strategies, as it allows to accurately regulate the evaporating temperature with a lower control effort while keeping the superheating in a safer operating range.

In this study we investigate the performance of MPC and PID based strategies to optimally recover waste heat through ORC technology. Using existent components from the ThermoCycle library Quoilin et al. (2014), a dynamic model of an ORC system is built for simulation purposes. The model dynamics

⋆ The results presented in this paper have been obtained within the frame of the IWT SBO-110006 project The Next Generation Organic Rankine Cycles (www.orcnext.be), funded by the Institute for the Promotion and Innovation by Science and Technology in Flanders, Belgium. This financial support is gratefully acknowledged.

are coherent with those observed on real systems as presented in Desideri et al. (2014), where the dynamic models is experimentally validated using a low-capacity (11 kWe) waste heat recovery unit equipped with a single screw expander. Using the developed dynamic model, insight on the system dynamics and optimal operation is achieved, resulting in the development of a real-time optimizer to compute the optimal evaporating temperature which maximizes the power generation. The controller's task is to track the optimal set-point generated by the optimizer while ensuring a minimum superheating value for safely operation.

The paper is structured as follows. Section 2 introduces the architecture and main characteristics of the ORC system. Next, in section 3 the Extended Prediction Self-Adaptive (EPSAC) approach to MPC used in this study is briefly described. A low-order model suitable for prediction is then developed using parametric identification as described in section 4. The control structure, design and tuning of the proposed PID and MPC based strategies is described in section 5, followed by the simulation results in section 6. Finally a conclusion section summarizes the main outcome of this contribution.

2. PROCESS DESCRIPTION

This section describes the architecture and main characteristics of the ORC system used for evaluating the performance of the developed control strategies.

2.1 The Organic Rankine Cycle System

In order to assess the performance of the different developed control strategies, a dynamic model of the ORC system presented in Fig. 1 has been developed in the Modelica language using existent components from the Thermo Cycle library Quoilin et al. (2014). The developed model is then exported into Simulink/Matlab environment by means of the Functional Mock-Up Interface (FMI) open standard.

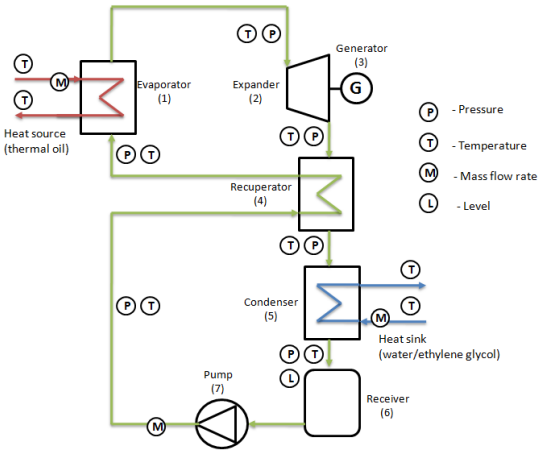


Fig. 1. Schematic layout of the pilot plant available at Ghent University, campus Kortrijk (Belgium)

The system based on a regenerative cycle and solkatherm (SES36) as working fluid, shows a nominal power of 11 kWe. The expander is originally a single screw compressor adapted to run in expander mode. It drives an asynchronous generator connected to the electric grid through a four-quadrant inverter, which allows varying the generator rotational speed (N_{exp}). During the simulations performed in this paper, the generator rotational speed is kept constant at 3000 rpm to emulate an installation directly connected to the grid. The circulating pump (N_{pp}) is a vertical variable speed 14-stage centrifugal pump with a maximum pressure of 14 bar and 2.2 kWe nominal power.

Starting from the bottom of the scheme it is possible to recognize the liquid receiver (b) installed at the outlet of the condenser (a) where the fluid is collected in saturated liquid condition. From the receiver outlet, the fluid is pumped (c) through the regenerator (d) cold side, and the evaporator (e), where it is heated up to superheated vapor, reaching its maximum temperature at the evaporator outlet. The fluid, after being expanded in the volumetric machine (f), enters the regenerator hot side, and then it flows into the condenser (a) to close the cycle.

In order to assess the performance of the different developed control strategies a dynamic model of the ORC system (Fig. 1) has been developed in the Modelica language using existent components from the ThermoCycle library Quoilin et al. (2014). The developed model is then exported into Simulink/Matlab® environment by means of the Functional Mock-Up Interface (FMI) open standard.

2.2 Conditions for optimal operation of an ORC unit

In order to optimally operate an ORC unit, two main conditions need to be satisfied: **i)** keep the cycle in a safe condition during operation and **ii)** maximize the net output power. Safe operation of the ORC unit is important as it allows a longer life expectancy in all components. In this concern, an accurate regulation of superheating (ΔT_{sh}), is the main priority since a minimum value of superheating has to be guaranteed in order to avoid a wet expansion (i.e., formation of liquid droplets at the expander inlet that can damage the expansion machine). The superheating is defined as:

$$\Delta T_{sh} = T_{exp,in} - T_{sat,ev} \tag{1}$$

where $T_{exp,in}$ is the temperature measured at the inlet of the expander and $T_{sat,ev}$ the evaporating temperature, corresponding to the temperature at which the fluid undergoes the phase transition from saturated liquid to saturated vapor at the fixed evaporating pressure $p_{sat,ev}$.

In order to maximize the output power the evaporating temperature represents the most relevant control variable Quoilin et al. (2011), which needs to be adapted depending on the heat source conditions Hernandez et al. (2015). The main terms to assess the performance of the ORC system are the net output power and the cycle efficiency, which are defined as:

$$\dot{W}_{el,net} = \dot{W}_{exp} - \dot{W}_{pump} \tag{2}$$

$$\eta_{cycle} = \frac{\dot{W}_{el,net}}{\dot{Q}_{in,ORC}} \tag{3}$$

where $\dot{W}_{exp}$ is the expander electrical power, $\dot{W}_{pump}$ is the pump electrical power and $\dot{Q}_{in,ORC}$ is the thermal power supplied to the ORC system in the evaporator.

2.3 Optimal evaporating temperature

Previous studies have demonstrated the existence of an optimal evaporating temperature which maximizes the output power for a given heat source conditions, where a real-time optimizer (RTO) can be built using a steady-state model Quoilin et al.

(2011) or by means of extremum-seeking algorithm Hernandez et al. (2016). In this paper the first approach is chosen, thus leading to the following correlation used in the RTO:

$$T_{sat,opt} = -290.915 + 183.33 * log_{10}(T_{hf}) + 10.636 * \dot{m}_{hf} \quad (4)$$

Equation (4) is valid in the range of $0.5 \leq \dot{m}_{hf} \leq 1.5\,\mathrm{kg/s}$ and $90 \leq T_{hf} \leq 125\,^{\circ}\mathrm{C}$ given a constant saturation temperature in the condenser of $p_{sat,cd} = 1.4\,\mathrm{bar}$.

3. MODEL PREDICTIVE CONTROL

A brief introduction to EPSAC algorithm is presented in this section. For a detailed description the reader is referred to De Keyser (2003); Hernandez et al. (2015).

3.1 Computing the Predictions

Using EPSAC algorithm, the measured process output can be represented as:

$$y(t) = x(t) + n(t) \quad (5)$$

where $x(t)$ is the model output which represents the effect of the control input $u(t)$ and $n(t)$ represents the effect of the disturbances and modeling errors, all at discrete-time index t. Model output $x(t)$ can be described by the generic system dynamic model:

$$x(t) = f[x(t-1), x(t-2), \ldots, u(t-1), u(t-2), \ldots] \quad (6)$$

Notice that $x(t)$ represents here the model output, not the state vector. Also important is the fact that f can be either a *linear* or a *nonlinear* function.

Furthermore, the disturbance $n(t)$ can be modeled as colored noise through a filter with the transfer function:

$$n(t) = \frac{C(q^{-1})}{D(q^{-1})} e(t) \quad (7)$$

with $e(t)$ uncorrelated (white) noise with zero-mean and C, D monic polynomials in the backward shift operator q^{-1}. The disturbance model must be designed to achieve robustness of the control loop against unmeasured disturbances and modeling errors Maciejowski. (2002).

A fundamental step in the EPSAC methodology consists of the prediction. Using the generic process model (5), the predicted values of the output are:

$$y(t+k|t) = x(t+k|t) + n(t+k|t) \quad (8)$$

$x(t+k|t)$ and $n(t+k|t)$ can be predicted by recursion of the process model (6) and by using filtering techniques on the noise model (7), respectively De Keyser (2003).

3.2 Computing the optimal control action

A key element in linear MPC is the use of base (or free) and optimizing (or forced) response concepts Maciejowski. (2002). In EPSAC, the future response can be expressed as:

$$y(t+k|t) = y_{base}(t+k|t) + y_{optimize}(t+k|t) \quad (9)$$

The two contributing factors have the following origin:

- $y_{base}(t+k|t)$ is the effect of the past inputs, the a priori defined future base control sequence $u_{base}(t+k|t)$ and the predicted disturbance $n(t+k|t)$.
- $y_{optimize}(t+k|t)$ is the effect of the additions $\delta u(t+k|t)$ that are optimized and added to $u_{base}(t+k|t)$, according to $\delta u(t+k|t) = u(t+k|t) - u_{base}(t+k|t)$. The effect of these additions is the discrete time convolution of $\Delta U = \{\delta u(t|t), \ldots, \delta u(t+N_u-1|t)\}$ with the impulse response coefficients of the system (G matrix), where N_u is the chosen control horizon.

The control ΔU is the solution to the following constrained optimization problem:

$$\begin{aligned} \Delta U = & arg \min_{\Delta U \in \mathbb{R}^{N_u}} \sum_{k=N_1}^{N_2} [r(t+k|t) - y(t+k|t)]^2 \\ & subject\ to\ |M.\Delta U \leq N \end{aligned} \quad (10)$$

where N_1 and N_2 are the minimum and maximum prediction horizons, N_u is the control horizon, $r(t+k|t)$ is a future set-point or reference sequence. The various process input and output constraints can all be expressed in terms of ΔU, resulting in matrices M, N. Since (10) is quadratic with linear constraints in decision variables ΔU, then the minimization problem can be solved by a quadratic programming (QP) algorithm Maciejowski. (2002).

4. SYSTEM IDENTIFICATION

A trade-off between complexity of the model and prediction accuracy has to be made, in order to ensure the correct performance of the MPC strategy. In this work we have chosen a pragmatic approach by performing a parametric identification based on experimental data recorded in the available setup.

The model has been identified from the manipulated variable, pump speed (N_{pp}) to the evaporating temperature ($T_{sat,ev}$) and superheating (ΔT_{sh}). The identification has been performed using a multisine excitation signal and the prediction error method (pem) Ljung (2007). The sampling time $T_s = 1\,\mathrm{s}$ has been chosen according to the fastest dynamics of the system.

It is important to notice that in practice it is also possible to measure the temperature and mass flow rate of the heat source (T_{hf}) and ($\dot{m}_{hf}$), making possible to use them as measured disturbances. Therefore, models from these variables to evaporating temperature ($T_{sat,ev}$) and superheating (ΔT_{sh}) are also built. The nominal operating conditions of the system are presented in table 1.

Table 1. Nominal operating conditions considered for the Identification Procedure

Parameter	*Description*	*Value*	*Unit*
N_{pp}	Pump rotational speed	1680	rpm
N_{exp}	Expander rotational speed	3000	rpm
$T_{sat,ev}$	Evaporating temperature	100	$^{\circ}C$
ΔT_{sh}	Superheating	20	$^{\circ}C$
T_{hf}	Temperature hot fluid	120	$^{\circ}C$
$\dot{m}_{hf}$	Mass flow rate hot fluid	1.0	kg/s
T_{cf}	Temperature cold fluid	15	$^{\circ}C$
$\dot{m}_{cf}$	Mass flow rate cold fluid	3.0	kg/s
$\dot{W}_{el,net}$	Net output power	11	kW
η_{cycle}	Cycle efficiency	6	%

The identified model is presented in (11) in the form of discrete-time transfer functions using the backwards shift operator q^{-1}.

$$\frac{\Delta T_{sh}(t)}{N_{pp}(t)} = \frac{-0.063q^{-1}+0.059q^{-2}}{1-2.44q^{-1}+1.955q^{-2}-0.51q^{-3}} \tag{11a}$$

$$\frac{\Delta T_{sh}(t)}{T_{hf}(t)} = \frac{0.47q^{-1}}{1-0.51q^{-1}} \tag{11b}$$

$$\frac{\Delta T_{sh}(t)}{m_{hf}(t)} = \frac{-2.98q^{-1}+4.29q^{-2}-1.31q^{-3}}{1-1.35q^{-1}-0.11q^{-2}+0.46q^{-3}} \tag{11c}$$

$$\frac{T_{sat}(t)}{N_{pp}(t)} = \frac{0.066q^{-1}-0.063q^{-2}}{1-2.42q^{-1}+1.91q^{-2}-0.49q^{-3}} \tag{11d}$$

$$\frac{T_{sat}(t)}{T_{hf}(t)} = \frac{0.0017q^{-11}-0.0017q^{-12}}{1-3.6q^{-1}+4.88q^{-2}-2.95q^{-3}+0.67q^{-4}} \tag{11e}$$

$$\frac{T_{sat}(t)}{m_{hf}(t)} = \frac{2.43q^{-1}-6.16q^{-2}+5.33q^{-3}-1.6q^{-4}}{1-2.93q^{-1}+3.12q^{-2}-1.42q^{-3}+0.23q^{-4}} \tag{11f}$$

5. CONTROL STRUCTURE AND TUNING

In this section the control structure and tuning procedure of the proposed strategies are discussed.

Three control strategies are developed in order to control the ORC unit, two based on PID and one based on MPC. In all strategies we make use of the real-time optimizer (RTO) to compute, as a function of the varying heat source conditions, the optimal evaporating temperature $T_{sat,opt}$ which will be used as reference to the controller, as illustrated in Fig. 2.

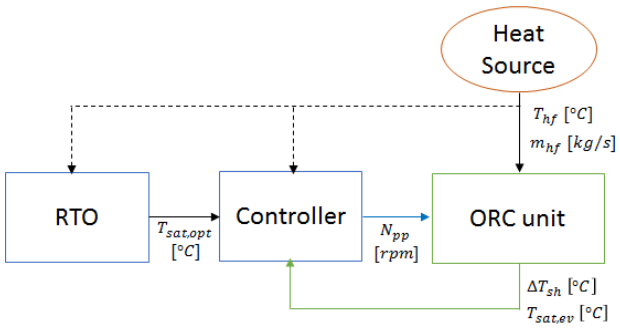


Fig. 2. Control structure of the proposed closed loop including the real-time optimizer (RTO).

An important element on the control design are the physical constraints and the safety conditions of the system. For which, the controller is required to respect the input (pump speed) and output (superheating) constraints, summarized in table 2.

Table 2. Operation constraints of the ORC unit

Variable	*max*	*min*	Δ
Pump Speed N_{pp}	1320 rpm	2100 rpm	60 rpm/s
Superheating ΔT_{sh}	–	10 °C	-

5.1 PI strategy

This strategy is based on a PI controller which is used to track the optimal evaporating temperature $T_{sat,opt}$. The PI controller is tuned using the transfer function which relates the speed in the pump to the evaporating temperature (11) for the following design specifications: settling time $T_{set} = 60$ s, overshoot percent $OS\% = 0$ and robustness $R_o = 0.7$, obtaining the PI parameters:

$$PI_{T_{sat,ev}} = K_p\left(1+\frac{1}{T_i s}\right) = 0.189\left(1+\frac{1}{1.7813\,s}\right) \tag{12}$$

During the implementation phase the clamping anti-reset windup scheme is used to clip the control action into the permissible range of the pump (table 2).

5.2 Switching PIs

In order to improve the performance of the PI strategy, essentially to what refers to safety conditions, an override control (here called switching PI strategy) is implemented. In this strategy the $PI_{T_{sat,ev}}$ controller is used to follow the optimal evaporating temperature set-point unless that superheating value goes below a threshold value $\Delta T_{sh} < 10\,°C$, in which case another PI controller for superheating $PI_{DT_{sh}}$, is used with set-point at $\Delta T_{sh,ref} = 10\,°C$, in order to bring the system back to a safe stage.

While the PI controller for the evaporating temperature is the same used on the basic PI strategy (12), the PI controller for superheating is tuned using the transfer function which relates the speed of the pump to the superheating found in (11), for the following design specifications: settling time $T_{set} = 60s$, overshoot percent $OS\% = 0$ and robustness $R_o = 0.7$, the following PI parameters are obtained:

$$PI_{\Delta T_{sh}} = K_p\left(1+\frac{1}{T_i s}\right) = -1.1\left(1+\frac{1}{0.98\,s}\right) \tag{13}$$

5.3 MPC-EPSAC

In this strategy a constrained MPC-EPSAC controller is implemented to track the optimal evaporating temperature $T_{sat,opt}$, while ensuring superheating will remain above 10 °C.

In MPC, a balance between acceptable control effort and acceptable control error can be obtained via many tuning parameters (e.g., the reference trajectory design parameter α; the prediction horizon N_2 and the control horizon design parameter N_u). Closed loop performance is designed using the N_2 parameter, whereas larger values provide a more conservative and robust control. The control horizon N_u is used to structure the future control scenario, reducing the degrees of freedom from N_2 to N_u. Structuring leads to simplified calculations and has generally a positive effect on robustness. The design parameter α in the reference trajectory can vary in the range of: $0 \leq \alpha \leq 1$. A value of α closer to 1 means a smoother variation of the set-point and hence a less aggressive control action.

A trade-off between closed loop speed and robustness has been obtained for $N_2 = 15$, $N_u = 1$ and $\alpha = 0.5$. The main goal is to achieve a response without overshoot $OS\% = 0$ and settling time of about 60 s. Another important element in the design of the controller is the choice of the disturbance model (7), during this study the 'default' filter $C(q^{-1}) = 1$ and $D(q^{-1}) = 1 - q^{-1}$ has been chosen leading to zero steady-state error Maciejowski. (2002). Notice that this filter choice acts like the integral action for PID controllers.

6. SIMULATION RESULTS AND DISCUSSION

The present study focuses on investigating which are the advantages of using advanced controllers such as MPC compared to PID-like strategies for the optimal operation of an ORC system in waste heat recovery applications. The control strategy task is to accurately regulate the evaporating temperature (given by the RTO), in order to maximize the energy production, while avoiding formation of liquid droplets that could damage the expander by ensuring an small amount of superheating ΔT_{sh}. Thus in this study we will focus on answering two questions:

- which are the heat source conditions which represent the main challenge for any controller? and

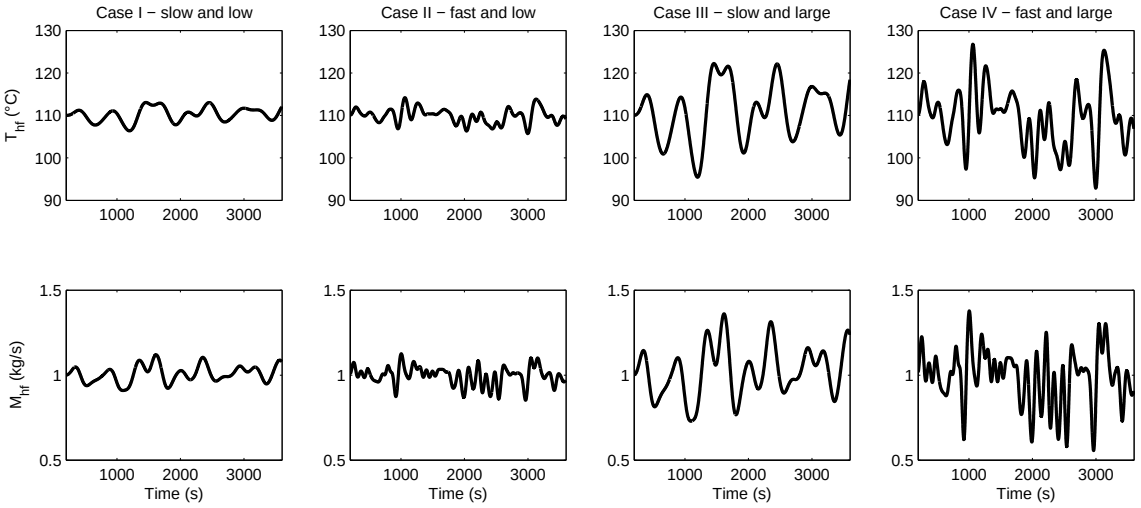


Fig. 3. Heat source profiles due to temperature T_{hf} and mass flow rate m_{hf} variations. Case I: slow and low amplitude variations. Case II: fast and low amplitude variations. Case III: slow and large amplitude variations. Case IV: fast and large amplitude variations.

- which controller produces the highest output power while keeping the safety conditions?

In order to answer those questions, we introduce four different scenarios depicted in figure 3, in which the controllers will be tested. The heat source variations are due to the combination of slow, fast, low or large transitions in temperature T_{hf} and mass-flow-rate m_{hf}:

- Case I: slow and low amplitude transitions. $T_{hf} = \pm 5\,^{\circ}\mathrm{C}$ and $\dot{m}_{hf} = \pm 0.1\,\mathrm{kg/s}$
- Case II: fast and low amplitude transitions. $T_{hf} = \pm 5\,^{\circ}\mathrm{C}$ and $\dot{m}_{hf} = \pm 0.1\,\mathrm{kg/s}$
- Case III: slow and large amplitude transitions. $T_{hf} = \pm 15\,^{\circ}\mathrm{C}$ and $\dot{m}_{hf} = \pm 0.3\,\mathrm{kg/s}$
- Case IV: fast and large amplitude transitions. $T_{hf} = \pm 18\,^{\circ}\mathrm{C}$ and $\dot{m}_{hf} = \pm 0.4\,\mathrm{kg/s}$

The three strategies tested for cases I and II (not shown here) result on good closed-loop performance, i.e., the difference in terms of tracking capabilities and control effort is negligible. During those heat source conditions the controllers are able to track correctly the quick transitions, meaning that the controllers have a high enough bandwidth and superheating remains into the desired limits.

Large amplitude variations in the heat source cause sudden drops in the superheating value, as depicted in Fig. 4 at time instant 1350 s. The switching mechanism avoids the superheating to decrease dramatically compared to the PI strategy, nevertheless, it still undergoes the threshold value of 10 °C. For the case of MPC, the most important element to highlight is the fact that this control strategy always respects the hard-output-constraint of $\Delta T_{sh} > 10\,^{\circ}\mathrm{C}$. Because it uses a model for prediction, it is able to better compensate possible sudden drops in the superheating, thus resulting in a higher net output power and higher life expectancy of the actuator, both due to the smoother control effort (i.e., lower pump speed N_{pp} variations).

Previous observations are more evident when analyzing the results for case IV, (i.e., fast and large amplitude variations), where sudden drops in the superheating value are observed

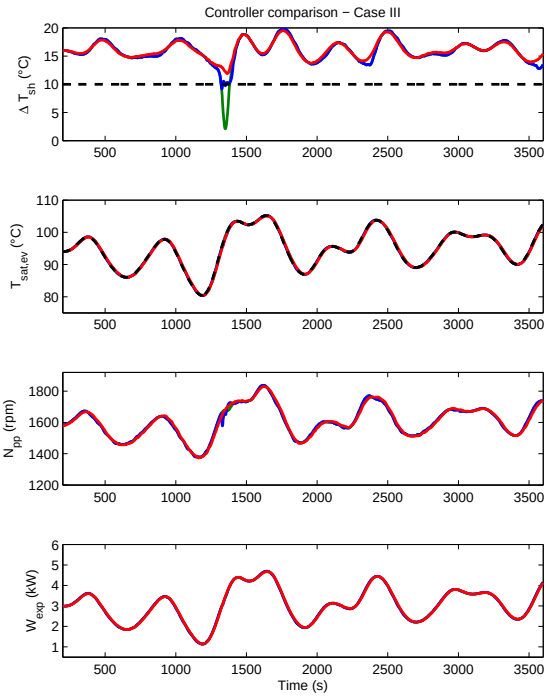


Fig. 4. Controllers comparison for Case III. Solid green- PI strategy; solid blue- Switching PI and solid red- represents MPC-EPSAC strategy.

at time instants 1038 s, 1350 s, 2080 s, 2664 s and 3080 s as depicted in Fig. 5.

The simulation results obtained suggest that MPC outperforms the PI based strategies for the case of large amplitude variations in the heat source. Hence, resulting in a desirable strategy regarding safety conditions. In a real industrial context, using a single PI would be an unsafe and therefore unusable strategy. Instead, the switching PI (in solid blue line) regulate

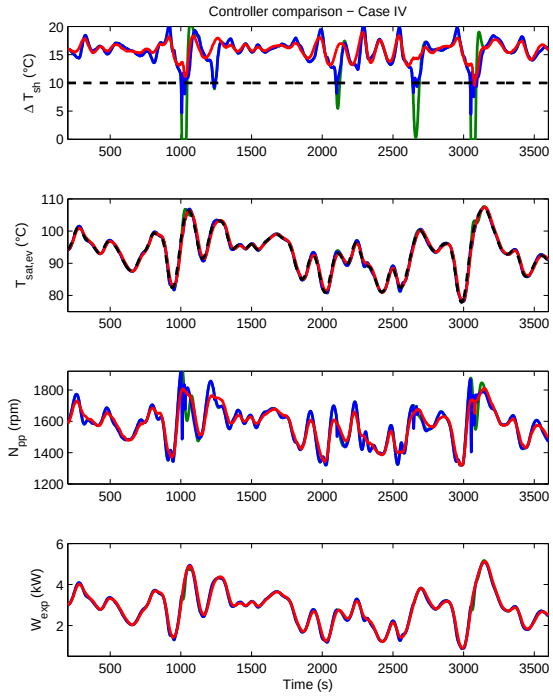


Fig. 5. Controllers comparison for Case IV. Solid green- PI strategy; solid blue- Switching PI and solid red- represents MPC-EPSAC strategy.

both evaporating temperature and superheating, thus enabling a safer operation while keeping smooth transition between the controllers. However, there is no direct control on how much can superheating decrease, as observed in for cases III and IV where superheating values close to 5 °C were observed.

7. CONCLUSION

In the present contribution three different PI and MPC based strategies have been designed and tested in simulation, with the goal of optimizing the working conditions of an ORC unit for WHR applications.

Results suggest that large amplitude variations in the waste heat source (e.g. cases III and IV), represent the most challenging situation for control design. Hence, implementation of advanced controllers such as MPC is highly recommended as it generates the same or higher net electrical output power compared to PID-based strategies while offering a safer operation. This is achieved by more accurately regulating the optimal evaporating temperature generated by the optimizer, while keeping the superheating at safe values, resulting also in a higher efficiency of the system. On the other hand, for the case of low amplitude variations (e.g. cases I and II), PID-like strategies might offer a satisfactory performance.

Future work includes adding more degrees of freedom by manipulating the expander speed, or by acting on the mass flow rate and/or temperature of the heat sink.

REFERENCES

Angelino, G., Gaia, M., and Macchi., E. (1984). A review of italian activity in the field of organic rankine cycles. In *proceedings of the International VDI-Seminar*, 465–482.

Colonna, P. and Van Putten., H. (2007). Dynamic modeling of steam power cycles.: Part i modeling paradigm and validation. *Applied Thermal Engineering*, 27(2-3), 467 – 480.

De Keyser, R. (2003). *Model based Predictive Control for Linear Systems*, chapter invited in UNESCO Encyclopaedia of Life Support Systems (EoLSS). Article contribution 6.43.16.1, Oxford, 35 pages.

Desideri, A., van den Broek, M., Gusev, S., Lemort, V., and Quoilin, S. (2014). Experimental campaign and modeling of a low-capacity waste heat recovery system based on a single screw expander. In *22nd International Compressor Engineering Conference*. Purdue, USA.

Feru, E., Willems, F., de Jager, B., and Steinbuch, M. (2014). Modeling and control of a parallelwaste heat recovery system for euro-vi heavy-duty diesel engines. *energies*, 7, 6571–6592.

Grelet, V., Dufour, P., Nadri, M., Lemort, V., and Reichel, T. (2015). Explicit multi-model predictive control of a waste heat rankine based system for heavy duty trucks. In *In IEEE Conference on Decision and Control, Osaka, Japan*.

Hernandez, A., Desideri, A., Ionescu, C., De Keyser, R., Lemort, V., and Quoilin, S. (2016). Real-time optimization of organic rankine cycle systems by extremum-seeking control. *Energies*, 9(5), 334.

Hernandez, A., Desideri, A., Ionescu, C., Quoilin, S., Lemort, V., and De Keyser, R. (2014). Increasing the efficiency of organic rankine cycle technology by means of multivariable predictive control. In *Proc. of the 19th World IFAC Congress (IFAC 2014)*.

Hernandez, A., Desideri, A., Ionescu, C., Quoilin, S., Lemort, V., and De Keyser., R. (2015). Experimental study of predictive control strategies for optimal operation of organic rankine cycle systems. In *Proceedings of the European Control Conference (ECC15)*. Linz, Austria.

Ljung, L. (2007). *System identification: theory for the user*. Prentice-Hall.

Maciejowski., J. (2002). *Predictive Control: With Constraints*. Pearson Education. Prentice Hall.

Peralez, J., Tona, P., Nadri, M., Dufour, P., and Sciarretta, A. (2015). Optimal control for an organic rankine cycle on board a diesel-electric railcar. *Journal of Process Control*, 33, 1 – 13.

Quoilin, S., Aumann, R., Grill, A., Schuster, A., and Lemort., V. (2011). Dynamic modeling and optimal control strategy for waste heat recovery organic rankine cycles. *Applied Energy*, Vol. 88, 2183–2190.

Quoilin, S., Desideri, A., Wronski, J., Bell, I., and Lemort., V. (2014). Thermocycle: A modelica library for the simulation of thermodynamic systems. In *Proceedings of the 10th International Modelica Conference*. Lund, Sweden.

Sun, J. and Li., W. (2011). Operation optimization of an organic rankine cycle (orc) heat recovery power plant. *J. Applied Thermal Engineering*, Vol. 31, 2032–2041.

Verneau, A. (1979). Waste heat recovery by organic fluid rankine cycle. In *In Proceedings from the First Industrial Energy Technology Conference*, 940–952. Houston, TX.

Wei, D., Lu, X., Lu, Z., and Gu., J. (2007). Performance analysis and optimization of organic rankine cycle (orc) for waste heat recovery. *J. Energy Conversion and Management*, Vol. 48, 1113–1119.

PID Optimal Controller with Filtered Derivative Part for Unstable First Order Plus Time Delay Systems

J. A. Cárdenas-Valderrama*, J. F. Marquez-Rubio*, O. Jiménez-Ramírez*

**Instituto Politécnico Nacional, ESIME Unidad Culhuacán, Sección de Estudios de Posgrado e Investigación, Av. Santa Ana, 1000, México DF, 04430, México, (e-mail: cardenas.v@live.com.mx, jfcomr23@yahoo.com.mx, cuauhpitzote@hotmail.com)*

Abstract: A typical problem in the derivative part of the PID controllers is its practical implementation. This work avoids to assume high values for the filter coefficient instead, a stability analysis of a proposed filtered PID controller to the case of systems with time delay, particularly the Unstable First Order plus Time Delay (UFOPTD) systems is presented. This analysis becomes interesting since analytical results have not been presented in literature. The aim of the obtained results is to illustrate how the stability conditions are modified when different values of the filter coefficient are chosen. In fact, the parametric stabilization region is obtained based in previous literature results. As a second goal of the present research is to obtain the optimal parameters of the proposed filtered PID controller such that the energy of control input and system states be minimized. An optimization methodology based on the second method of Lyapunov is obtained.

Keywords: PID controller, Stabilization, Time-delay, Unstable Processes, Optimization.

1. INTRODUCTION

Time-delay of control inputs is a common phenomenon in diverse application fields, including chemical processes, hydraulic systems and servomechanisms. Recently, some systems are remotely controlled generating a time delay between the controlled system and the control stage. The control problem of time delay systems arises mainly by the induced transcendental term in the characteristic equation, which gives as result a characteristic equation having an infinite number of poles. Due to the complexity of the problem, several researchers have devoted its efforts for designing control strategies that provide an adequate performance of the system; see for instance Seshagiri, et al., (2007). To the case of open-loop stable process, the well-known Smith Predictor Compensator (SPC) has been used as a traditional control structure Seshagiri, et al., (2007). Considering a similar approach, some works have been reported in order to deal with unstable processes, see for instance, Liu et al., (2005), Seshagiri and Chidambaram (2005). As a first attempt to analyse a generalized class of delayed systems, the case of first order delayed system has been widely studied by using the basis provided by the SPC, see for instance Seshagiri, et al., (2007), Michiels et al., (2002), Marquez et al., (2012). With a simple and different perspective, some authors have regarded to analyse the case when the system is controlled by a Proportional (P), Proportional-Integral (PI) and Proportional-Integral-Derivative (PID). Nesimioglu and Soylemez Nesimioglu and controllers. Hwang and Hwang (2004) used the D-partition technique to estimate the stabilization limits of PID compensation, showing that an UFOPTD system can be stabilized if $\theta < \theta_{un}$, where θ is the time-delay and θ_{un} is the unstable constant-time. Silva and Bhattacharyya (2005) provided a complete parametrization of the stabilizing P and PI controllers in the case $\theta < \theta_{un}$ and the stabilizing PID controllers for the case $\theta < 2\theta_{un}$. Recently, Marquez et al., (2014) consider the stabilization of UFOPTD system by using a traditional Proportional-Derivative (PD) controller, showing that the derivative term minimizes the effects of the time-delay. In Lee et al., (2010) some results about the stabilization of a delayed system with one unstable pole and several stable poles by using P, PI, PD and PID controllers are provided. An important issue in the derivative part of PID controllers is its practical implementation. The solution of using a filter to the derivative action has been widely proposed in the literature. However, just high values of the filter coefficient is suggested in order to recuperate the inherent derivative action of the controller. The task of this work is to analyze the stability properties when a filtered PID controller is considered to the case of UFOPTD system and showing how the stability conditions of the closed-loop system can be modified for different values of the filter coefficient. As we mentioned before, the control strategies previously cited only provides stability results and there are not result related to the performance of the controlled variable. It is known that when a control strategy is applied to a system it is desired to obtain optimal performance of the system with respect to specific variables. Also, in this work a simple and effective methodology to obtain the optimal gains k_p, k_i and k_d gains is proposed in order to minimize the energy at the control input and the system states. The proposed optimization methodology is based on an approximation of the delay term as well as on the second method of Lyapunov. It is important to note that the proposed optimization strategy assumes that the stabilizing region of the control parameters is known, and such stabilizing regions are obtained from previous literature results such as Lee et

al., (2010) and Marquez et al., (2014). This work is organized as follows. Section 2 presents the problem formulation. After this, in Section 3 some preliminary results are given. Then, Section 4 presents the main results of this work. In Section 5 a numerical example is given and finally in Section 6 some conclusions are provided.

2. PROBLEM FORMULATION

Consider an UFOPTD system given by,

$$\bar{G}(s) = \frac{Y(s)}{U(s)} = \frac{b}{s-a}e^{-\theta s}, \qquad (1)$$

a PID controller with filtered derivative part of the form,

$$\bar{C}(s) = \bar{k}_p + \frac{\bar{k}_i}{s} + \frac{\bar{k}_d s}{\frac{s}{N}+1} \quad or,$$

$$\bar{C}(s) = \frac{\left(N\bar{k}_d + \bar{k}_p\right)s^2 + \left(N\bar{k}_p + \bar{k}_i\right)s + N\bar{k}_i}{s(s+N)}, \qquad (2)$$

where N is known as coefficient filter, and the control scheme shown in Fig 1. If the parameters $\bar{k}_p$, $\bar{k}_i$ and $\bar{k}_d$ are known in the case of the typical form of PID controller (without filter at derivative term), a traditional and heuristic way to implement in practice the controller given in (2) is setting N as $N \gg 0$. Notice that for $N \gg 0$ in (2) the properties of the non-filtered PID controller are recovered. However there is not a guideline to set the filter coefficient N or an explanation if the closed-loop system remains stable when N decreases. This work considers this problem to the proposed filtered PID controller, it is provided the allowable value of the coefficient filter such that the closed loop is stable and it is shown how the stability properties are modified due to the values of N. Consider a system given by (1) and a PID controller given by (2). The open loop transfer function given by,

$$\bar{C}(s)\bar{G}(s) = \frac{\left(\left(N\bar{k}_d + \bar{k}_p\right)s^2 + \left(N\bar{k}_p + \bar{k}_i\right)s + N\bar{k}_i\right)be^{-\theta s}}{s(s+N)(s-a)}, \qquad (3)$$

can be separated as follows,

$$C(s)G(s) = \left(k_p + \frac{k_i}{s} + k_d s\right)\frac{b}{(s-a)(s+N)}e^{-\theta s}, \qquad (4)$$

where $k_p = N\bar{k}_p + k_i$, $k_d = N\bar{k}_d + \bar{k}_p$, $k_i = N\bar{k}_i$ and

$$C(s) = \left(k_p + \frac{k_i}{s} + k_d s\right) \qquad (5)$$

$$G(s) = \frac{b}{(s-a)(s+N)}e^{-\theta s} \qquad (6)$$

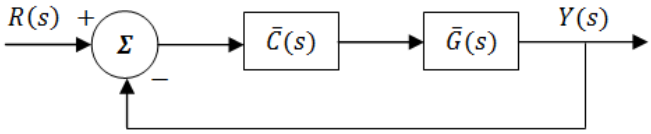


Fig 1. A control scheme UFOPTD system.

Notice that the closed-loop stability properties of the system (1) with the controller (2) can be obtained from the closed-loop stability properties of system (6) controlled by (5). In fact the stability properties of the closed-loop system (5)-(6) are presented in Lee et al., (2010).

3. PRELIMINAR RESULTS

Consider a system given by (6) and a PID controller by (5). The following result establishes the closed-loop stability conditions.

[Lee et al., (2010)] **Lemma 1.** A necessary condition for a PID controller to stabilize the system given by (6) is,

$$\theta < \sqrt{\left(\frac{1}{a}\right)^2 + \left(\frac{1}{N}\right)^2} + \frac{1}{a} - \frac{1}{N} \qquad (7)$$

If this condition is satisfied then the range of $\frac{k_d}{k_p}$ values for which a solution exists to the PID stabilization problem are given by,

$$\theta - \frac{1}{a} + \frac{1}{N} < \frac{k_d}{k_p} < \sqrt{\left(\frac{1}{a}\right)^2 + \left(\frac{1}{N}\right)^2}, \qquad (8)$$

and $\frac{k_i}{k_p}$ should be such that,

$$atg(\omega) + atg\left(\frac{k_d}{k_p}a\omega - \frac{k_i}{k_p}\frac{1}{a\omega}\right) - atg\left(\frac{a\omega}{N}\right) - \theta a\omega > 0,$$

for some $\omega > 0$, and

$$\frac{Na}{b}\sqrt{\frac{(1+\omega_{c1}^2)\left(1+\frac{a^2\omega_{c1}^2}{N^2}\right)}{1+\left(\frac{k_d}{k_p}a\omega_{c1} - \frac{k_i}{k_p}\frac{1}{a\omega_{c1}}\right)^2}} < k_p < \frac{Na}{b}\sqrt{\frac{(1+\omega_{c2}^2)\left(1+\frac{a^2\omega_{c2}^2}{N^2}\right)}{1+\left(\frac{k_d}{k_p}a\omega_{c2} - \frac{k_i}{k_p}\frac{1}{a\omega_{c2}}\right)^2}}, \qquad (9)$$

with $\omega_{c1} < \omega_{c2}$ being the first two phase crossover solved from,

$$atg(\omega) + atg\left(\frac{k_d}{k_p}a\omega - \frac{k_i}{k_p}\frac{1}{a\omega}\right) - atg\left(\frac{a\omega}{N}\right) - \theta a\omega = 0.$$

4. MAIN RESULTS

In this section the main results of this work are depicted. The following result provides the stability conditions such that the PID controller given by (2) stabilizes the system given by (1). The main aim of the following result is to illustrate how the stability conditions are modified with respect to different values of the filter coefficient N.

Theorem 2. Consider a system given by (1), a PID controller of the form (2) and the control feedback (Fig 1). A necessary condition for a filtered PID controller stabilizes the closed-loop system is:

i) $\theta < \frac{2}{a}$ with $N \gg a$ (the case of non filtered PID, [Lee et al., (2010)])

ii) $\theta < \frac{1}{a}$ with $N \ll a$

iii) $\frac{1}{a} < \theta < \frac{2}{a}$ with $N > \frac{2-2a\theta}{a\theta^2-2\theta}$

Proof. i) From the necessary condition (7) given in Lemma 1, it should be noticed that with $N \gg a$, the term $\frac{1}{N} \to 0$ which leads to the condition,

$$\theta < \sqrt{\left(\frac{1}{a}\right)^2} + \frac{1}{a} = \frac{2}{a}$$

The condition (7) gives the necessary condition to stabilize the delayed system with one unstable pole and one stable, in our case the stable pole position is given by the value of the filter coefficient N. Therefore, high values of N ($N \gg a$), (7) is expressed as $\theta < \frac{2}{a}$.

Proof. ii) From the necessary condition (7) given by Lemma 1, if $N \ll a$ is considered i.e, $N = xa$ with $x \ll a$. Since the term $\left(\frac{1}{a}\right)^2 \ll \left(\frac{1}{xa}\right)^2$ then the term $\left(\frac{1}{a}\right)^2$ can be remove from the expression and we obtain,

$$\theta < \sqrt{\left(\frac{1}{xa}\right)^2} + \frac{1}{a} - \frac{1}{xa} = \frac{1}{a}$$

Proof. iii) From the necessary condition (7) given in Lemma 1and the inequality for filter coefficient N we obtain,

$$N > \frac{2-2a\theta}{a\theta^2-2\theta}$$

This result only is valid for the range $\frac{1}{a} < \theta < \frac{2}{a}$.

In what follows a simple and effective methodology to obtain the optimal k_p, k_i and k_d control parameters in order to minimize the energy of the input control and states system. It is important to note that the following optimization strategy assumes that the stabilizing region of the control parameters is known and such regions can be computed following Lemma 1 (Lee et al 2010). In order to obtain a rational representation on the complex variable "s" of the delay term the first step of the proposed methodology considers a second order Pade approximation, which can be expressed as,

$$e^{-\theta s} = \frac{s^2 - \frac{6}{\theta}s + \frac{12}{\theta^2}}{s^2 + \frac{6}{\theta}s + \frac{12}{\theta^2}} \qquad (10)$$

Then, by substituting the approximation (10) into the process given by (1), a rational representation on the complex variable "s" of the plant is obtained,

$$G(s) = \frac{b\left(s^2 - \frac{6}{\theta}s + \frac{12}{\theta^2}\right)}{(s-a)(s+N)\left(s^2 - \frac{6}{\theta}s + \frac{12}{\theta^2}\right)} \qquad (11)$$

Taking into account the approximated system (11), the proposed filtered PID (2), and the closed loop system shown in Figure 1, a closed-loop state space representation of the form,

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u$$
$$y = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}u, \qquad (12)$$

is obtained with,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -\frac{12}{\theta^2} & -\frac{6}{\theta} & b & 0 & 0 \\ 0 & -\frac{12\,k_d}{\theta^2} & a - b\,k_d & k_i & k_p - Nk_d \\ 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{12}{\theta} & -b & 0 & -N \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ k_d \\ 0 \\ 1 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & -\frac{6}{\theta} & b & 0 & 0 \end{bmatrix}, \quad D = [0] \qquad (13)$$

Now, based on the second method of Lyapunov an optimization process to obtain the optimal k_p, k_i and k_d control parameters is derived. The performance index defined to evaluate the behaviour of the system is given by,

$$J = \int_0^\infty \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{Q}\boldsymbol{x}dt = \boldsymbol{x}^{\boldsymbol{T}}(0)\boldsymbol{P}\boldsymbol{x}(0), \qquad (14)$$

where:

$\boldsymbol{x}$ is the states vector.

$\boldsymbol{P}$ is a Define Positive Matrix.

$\boldsymbol{Q}$ is a Define Positive Matrix.

$\boldsymbol{A}$ is a nxn Matrix.

J is the performance index.

The main objective of the optimization process is to minimize the behavior index (14) assuring closed-loop stability. In this way, the solution of the following equation is required,

$$\boldsymbol{A^T P + PA} = -\boldsymbol{Q} \qquad (15)$$

From the second method of Lyapunov it is known that if the equation (15) has a unique solution, the closed-loop system given by (12) is stable. Equation (15) should be solved by proposing $\boldsymbol{Q}$ and solving to $\boldsymbol{P}$. This step requires that one of the control gains (for instance k_p and k_d, k_p and k_i or k_i and k_d) are given from the stable region computed with Lemma 1 (Lee et al. 2010). Notice that the resultant matrix $\boldsymbol{P}$ contains the control parameters of the control. Once that $\boldsymbol{P}$ is obtained, $\boldsymbol{P}$ is replaced into the performance index (14). Then, in order to minimize the performance index (14), the derivative of the performance index (14) is computed and the optimal gain (for instance k_i, k_d *or* k_p) is solved from,

$$\frac{dJ}{dx} = 0 \qquad (16)$$

5. SIMULATION RESULTS

Example 3. Consider a UFOPTD system given by (1) with his unstable pole $a = 1$. From the results of the Theorem 2., θ values are in the range given by,

$$1 < \theta < 2$$

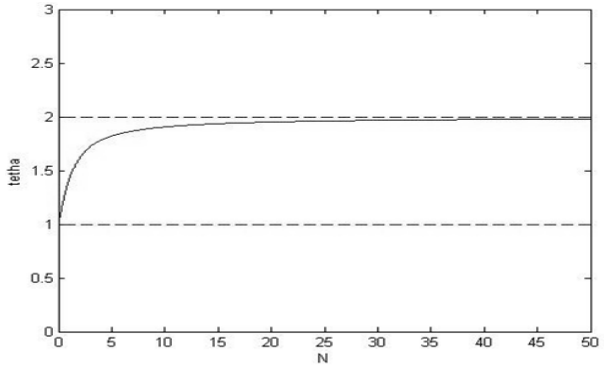


Fig. 2. Permissible region for N.

Fig 2. is a plot of θ-N that shows the maximum value that could be θ for a set of values of N. It can be seen that for small values of N, the maximum value of θ is close to 1 and for big values of N, the maximum value of θ converges to 2. Notice that for $N = 50$, the maximum value of θ.is close to 2.

Example 4. Consider a UFOPTD system given by,

$$\bar{G}(s) = \frac{1}{s-1}e^{-1.3s},$$

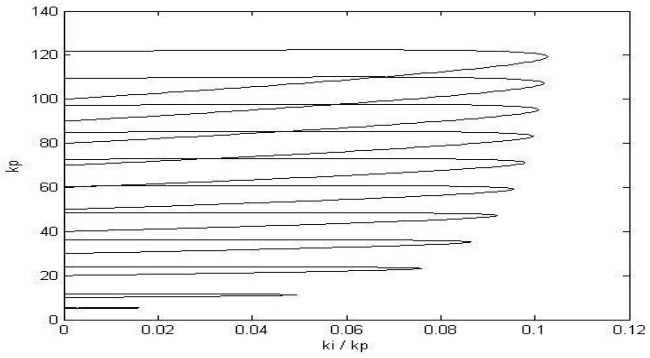


Fig 3. Stability regions for 5 < N < 100.

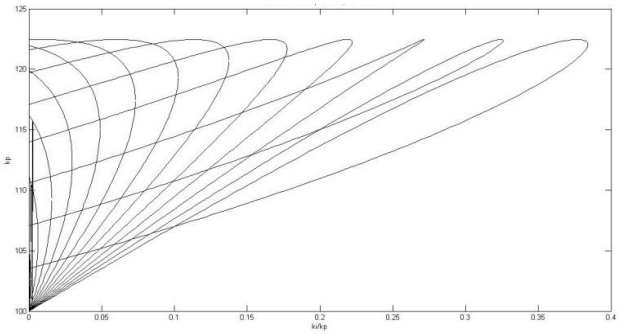


Fig 4. Stability regions for $0.35 < \frac{k_d}{k_p} < 0.95$.

and a PID controller by (2). We have rewritten the system and PID controller in the form given by (6)-(5). Following the Theorem 2, we can see the condition $\theta < \frac{2}{a}$ is satisfied due to $1.3 < 2$. From the (7), the range of N values is,

$$N > 0.65934066$$

Following the Lemma 1, the range of $\frac{k_d}{k_p}$ values is,

$$0.3 + \frac{1}{N} < \frac{k_d}{k_p} < \sqrt{1 + \left(\frac{1}{N}\right)^2},$$

$$atg(\omega) + atg\left(\frac{k_d}{k_p}\omega - \frac{k_i}{k_p}\frac{1}{\omega}\right) - atg\left(\frac{\omega}{N}\right) - 1.3\omega > 0,$$

For positive values of ω such that $\frac{k_i}{k_p} \geq 0$, and

$$1.3\sqrt{\frac{\left(1+\omega_{c1}^2\right)\left(1+\frac{\omega_{c1}^2}{N^2}\right)}{1+\left(\frac{k_d}{k_p}\omega_{c1}-\frac{k_i}{k_p}\frac{1}{\omega_{c1}}\right)^2}} < k_p < 1.3\sqrt{\frac{\left(1+\omega_{c2}^2\right)\left(1+\frac{\omega_{c2}^2}{N^2}\right)}{1+\left(\frac{k_d}{k_p}\omega_{c2}-\frac{k_i}{k_p}\frac{1}{\omega_{c2}}\right)^2}}.$$

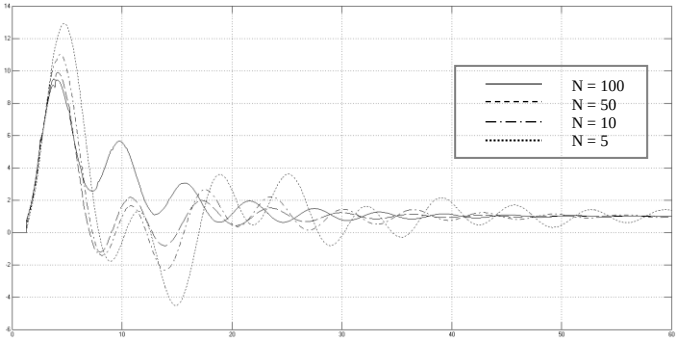


Fig 5. Output system for PID optimal parameters.

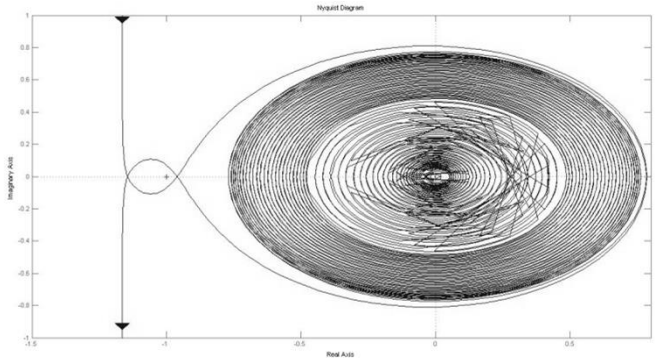


Fig 6. Nyquist plot of the closed-loop.

Table 1. comparison between approximate J and real J.

N	$\frac{k_d}{k_p}$	$J_{approximate}$	J_{real}	error rate
100	0.85	199.5011	200.9235	0.7079
	0.75	124.9385	127.1771	1.7602
	0.65	119.3475	121.4983	1.7702
	0.55	163.8896	166.2439	1.4162
	0.45	399.749	401.1684	0.3538
	0.35	7453.3	5753.6545	22.8039
50	0.85	202.3155	203.7388	0.6986
	0.75	128.8108	131.0875	1.7368
	0.65	125.9473	128.6289	2.0848
	0.55	178.3854	180.8468	1.3610
	0.45	472.3632	471.8013	0.1190
	0.35	16275	9984.886	38.6489
10	0.85	222.4695	224.292	0.8126
	0.75	170.5749	173.5998	1.7425
	0.65	205.3699	207.3216	0.9414
	0.55	433.4463	435.4711	0.4650
	0.45	4921.2	4346.442	11.6792
5	0.95	491.8966	470.7528	4.2984
	0.85	267.4086	270.0312	0.9712
	0.75	284.9237	287.5156	0.9015
	0.65	557.7871	560.7171	0.5225
1	1.35	72018	64714	10.1419

There are infinite stability regions due filter constant N and for each $\frac{k_d}{k_p}$ value of the range, there is a stability region of $k_p - \frac{k_i}{k_p}$. In this work we take some values for N and $\frac{k_d}{k_p}$ in order to show some stability regions of $k_p - \frac{k_i}{k_p}$ as well as compute some approximated performance indexes.

We compute J for some values of N and $\frac{k_d}{k_p}$ with (10), (11) and (13), and we compare it against real J obtained by simulation.

Table 1 shows approximate J, real J and the error rate for some values of N and $\frac{k_d}{k_p}$. Notice that the minimum J occurs when $N = 100$ and we conclude that a big value of N (with respect to the unstable constant a) to obtain minimum J.

Note that we use a small $\frac{k_d}{k_p}$ value, the error rate increases due to the use of a Pade approximation for time delay. We obtain the minimum J when we use $\frac{k_d}{k_p} = 0.75$. Thus, we take $N = 100$ - $\frac{k_d}{k_p}$ =0.65, $N = 50$ - $\frac{k_d}{k_p}$ =0.65, $N = 10$ - $\frac{k_d}{k_p}$ =0.75 and $N = 5$ - $\frac{k_d}{k_p}$ =0.85, and we show the output variable in Fig 5.

The satisfied closed-loop stability condition on the Nyquist plot is shown in Fig 6. for optimal values of k_i, k_p and k_d with $N = 100$. This plot shows one anticlockwise encircle to the critical point (-1,0) due to the one unstable pole.

6. CONCLUSIONS

In this paper, the stabilization of UFOPTD systems is investigated. The stability conditions by the PID controller with filtered derivative part are established to different N values.

The analysis provides exact stability region in terms of control parameters and indicates that stability can be achieved as long as $\theta < \frac{2}{a}$ and we have illustrated it through an example and its simulation. Comparison between approximate J and real J shows that Pade approximation is a good representation for time delay. To obtain the best J, it is necessarily choose a big N value and fixing an intermediate value of the $\frac{k_d}{k_p}$ range.

REFERENCES

C. Hwang, J. Hwang (2004). *Stabilization of first-order plus dead-time unstable processes using PID controllers*, IEE Proc. Control Theory Appl., Vol. 151, pp. 89-94.

Lee S. C., Wang Q. G. and Xiang C. (2010*). Stabilization of all-pole unstable delay processes by simple controllers*, Journal of Process Control, Vol. 20, pp. 235-239. 2010.

Liu T., Zhang W. and Gu D. (2005). *Analytical design of two-degree of freedom control scheme for open-loop unstable processes with time delay*, J. Process Control, Vol. 15, pp. 559-572.

Marquez-Rubio J. F., Del-Muro-Cuéllar B. Velazco-Villa M. and Novella-Rodríguez D. F. (2012). *Observer PID Stabilization Strategy for Unstable First-Order Linear Systems with Large Time Delay*, Industrial & Engineering Chemistry Research, Vol. 51, pp. 8477-8487.

Marquez-Rubio J. F., Del-Muro-Cuéllar B. and Alvarez-Ramírez j. (2014). *Stabilization Region of PD Controller for Unstable First Order Process with Time Delay*, International Journal of Control, Automation and Systems, Vol. 12(2), pp. 747-761.

Michiels W., Engelborghs K., Vansevnant P. and Roose D. (2002). *Continuous pole placement for delay equations*, Automatica, Vol. 38, pp. 747-761.

Nesimioglu B. S. and Soylemez M. T. (2010). *A simple derivation of all stabilizing proportional controller for*

first order time-delay systems, Asian J. Contrl, Vol. 14, pp. 1-7.

Normey-Rico J. E. and Camacho E. F. (2009). *Unified approach for robust dead-time compensator design*, J. Process Control, Vol. 19, pp. 38-47.

Seshagiri R. A. and Chidambaram M. (2005), *Enhanced Smith predictor for unstable processes with time delay*, Ind. Eng. Chem. Res., Vol. 44, pp. 8291-8299.

Seshagiri R. A. , Rao V. S. R. and Chidambadam M. (2007). *Simple analytical design of modified Smith Predictor with improved performance for unstable first-order plus time delay (UFOPTD) processes*, Ind. Eng. Chem. Res., Vol. 46, pp. 4561-4571.

Silva G. J. and Bhattacharyya S. P., *PID controller for time-delay systems*, Birkhuser. Boston, 2005.

Proposal of Two Degree of Freedom Structure for Hydro Governors

Francisco Javier Triveño Vargas * Hernan Jaldin Florero **

* Universidade de Araraquara, Rua Maria Antônia de Camargo, 170
CEP: 14800-370 – Araraquara, SP, Brazil
(e-mail:trivenoj@yahoo.com.br).
** CNDC Comite Nacional de Despacho de Carga, Calle Colombia, 0749
Cochabamba, Bolivia
(e-mail: h.jaldin.f@gmail.com)

Abstract: In this work the linear and nonlinear model of an hydraulic turbine are presented, it is described the model of an infinity bus generator to be used in non-linear simulations. In sequence, hydraulic mechanic governor and the Two Degree of Freedom (TDOF) proposal structure for speed governors are presented. The design and analysis of regulators are done in the frequency domain using requirements of gain and phase margin, both evaluated with Nichols Chart. Finally, the nonlinear simulations that include the excitation model and power system stabilizer are presented. It should be emphasized that this proposal is part of Control of Electric Generators with Matlab-Simulink® course from Bolivia.

Keywords: Hidrogenerators, Two Degrees-of-Freedom, Voltage Control, Speed Control.

NOMENCLATURE

Symbol	Definition	Unit
δ	rotor angle of synchronous generator	rad
ω_B	rotor speed deviation	rad/s
ω	frequency	rad/s
$T_{d_o}^{'}$	open circuit d-axis time constant	s
$T_{q_o}^{'}$	open circuit q-axis time constant	s
T_w	water starting time constant	s
T_R	reset time	s
H	net head	m
L	length of penstock pipe	m
H_r	rated hydraulic head	m
U	water velocity	m/s
A	piping area	m^2
Q_r	water-flow rate at rated load	m^3/s
$\overline{g}_{FL}$	gate opening full load	p.u
$\overline{g}_{NL}$	gate opening no load	p.u
ω_m	machine speed	p.u
V_t	generator terminal voltage	p.u
x_d	d-axis synchronous reactance	p.u
$x_d^{'}$	d-axis transient reactance	p.u
x_q	q-axis synchronous reactance	p.u
$x_q^{'}$	q-axis transient reactance	p.u
S_m	generator slip	p.u
T_e	electrical torque	p.u
T_m	mechanical torque	p.u
P_m	turbine mechanical power	p.u
E_{fd}	excitation system voltage	p.u
R_p	permanent droop	dimensionless
R_T	temporary droop	dimensionless
j	imaginary number	dimensionless
ρ, σ	weighting factors	dimensionless
κ	steady state gain	dimensionless
$\hat{q}(s), h(s), k(s)$	controller gains	dimensionless
D	damping coefficient	dimensionless
h	inertia constant	dimensionless

⋆ Araraquara University and CNDC.

1. INTRODUCTION

In April of 2015 was administered the course Modelling, Simulation and Control of Mechatronic and Aerospace Systems with Matlab-Simulink®and Mathematica™ executed between BOCIER (Comisión de Integración Eléctrica Regional Bolivia) and the National Committee of Load Dispatch from Bolivia (CNDC), located in the city of Cochabamba Vargas and Paglione (2015). This point marks the beginning of a series of improvement courses with the participation of engineers from different companies of the electrical industry in Bolivia. Other result of this initiative it was the developed of the course Electric Generators Control Using Matlab-Simulink® the same was executed in January of this year in the CNDC environments also.

The main reason to prepare these activities is that today's Power Systems are extraordinarily complex, so the service must be provided with high reliability and quality. Therefore, it is necessary to ensure the system's ability to supply loads continuously and voltage and frequency characteristics within the contractual restrictions. This means that the voltage and frequency must be maintained within permissible tolerance values to have nicely customers Machowski et al. (2008).

When the system is subjected to disturbances, it must be able to operate satisfactorily and to provide the required load demand successfully. Must support disturbances, such as a short circuit in a transmission line or output a generator Farmer (2001). The response to a disturbance can compromise considerable part of the equipment. For example the failure of a critical element, followed by its isolation by protection relays cause variations in power flows affecting the mainly voltage and rotor speeds of machine. Voltage variations trigger voltage regulators (AVR and PSS among others); and speed variations trigger speed regulators Kundur (1993).

Considering the comments above, the main concepts and mathematical development of speed regulators for hydroelectric, steam and gas turbines were presented. The simulation and project of lead-leg and PID controllers were included. Additionally, the controller of two degrees of freedom TDOF Wolovich (1994) was proposed as an alternative for speed regulator of hydroelectric turbines. The initial results obtained are the reason for submitting this work.

The main advantage of TDOF controller is that its design is pragmatical, once it depends only of two parameters (a third one can be added to corrects the steady state error). The feedback gains affect the stability of the system and the pre-filter affects the response characteristics. The controller exhibits robustness to plant uncertainties and if the sensitivity and complementary sensitivity functions are calculated Zhou and Doyle (1999), the controller reach disturbances rejection and noise attenuation as shown in Vargas et al. (2015). The strategy has been successfully implemented for the control position problem in a SCARA robot Vargas et al. (2004) and propose the structure to control and stability augmentation of executive aircraft also Vargas et al. (2015). Its focus is proposing an alternative solution to speed governor of hydroelectric turbines.

In section 3 the mathematical models of hydroelectric turbine are described. The synchronous generator model is presented in section 4. The main characteristics of speed governors are described in section 5, TDOF governor and its algorithm are described also. The designs of governors are presented in section 6.The linear analysis results are presented in section 7. Nonlinear simulation results are presented in section 6. Finally the conclusions are presented.

2. HYDROELECTRIC POWER PLANTS

A hydraulic generation plant converts mechanical energy (stored water) into electrical energy through a hydraulic turbine. Fig. 1 illustrates the components of simplified hydroelectric power generation unit.

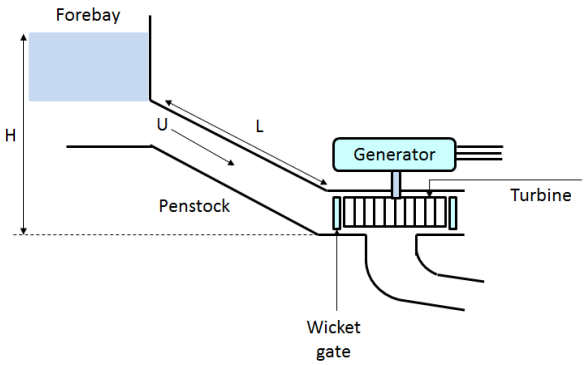


Fig. 1. Simplified hydroelectric power plant

From the reservoir, the water is transported from the area called forebay to the turbine through the water column. The water column comprises the entire structure used for transport, part of this structure are the penstock and surge tanks. Inertias of the water column and elasticity structures contribute to the water hammer effect which affects the performance of the turbine. Wicket gates are adjustable and pivot around the periphery of the turbine to control the amount of water admitted by the same.

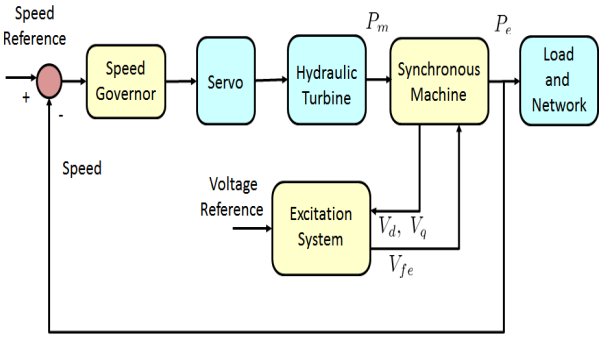


Fig. 2. Block diagram of hydroelectric power plants

These gates are adjusted by servo actuators controlled by the governor. Fig. 2 illustrates the simplified block diagram that shown the basic elements of the power generation process in a hydraulic plant.

3. HYDRAULIC TURBINE MODELS

A hydroelectric generation system presents a highly nonlinear behavior, for this reason appropriate mathematical models are essential for its understanding. Models can be classified into linear and nonlinear where can or not consider the effect of elasticity of the water column Naghizadeh et al. (2012).

3.1 Linear Model

The linear model is obtained from the basic equations of the turbine and the characteristics of penstock Kundur (1993), in these equations are made some simplifications considering the operation points of turbine. The ideal lossless turbine-penstock system is given by:

$$\frac{\Delta \overline{P}_m}{\Delta \overline{G}} = \frac{1 - T_w s}{1 + 0.5 T_w s} \tag{1}$$

Typical values of T_w vary between 0.5 and 4.0 seconds Kundur (1993).

3.2 Nonlinear Model

Traditionally the control design is done using linear models, however, the nonlinear model should be used for more elaborated simulations Vargas et al. (2015). Some examples corresponds to islanding, and load rejection, in these models, hydrodynamics and mechanical-electric dynamics are considered. Nonlinear model can be represented by the block diagram illustrated in Fig. 3. where $F(s) = \frac{1}{sT_w}$. This model take accounts the effects of varying flow on the effective water starting time, also incorporate the change in gains due to off nominal head variations (H_0), the offset for speed-no-load flow ($\overline{U}_{NL}$), and the effective reduction of gate stroke (A_t) given by:

$$A_t = \frac{1}{(\overline{g}_{FL} - \overline{g}_{NL})}$$

It is important emphasize that all per unit values ($\overline{\cdot}$) in the governor/turbine models are normally based on the rated power output MW of the turbine and not on the generator model MVA. The values of model are presented in Table 1 Kundur (1993).

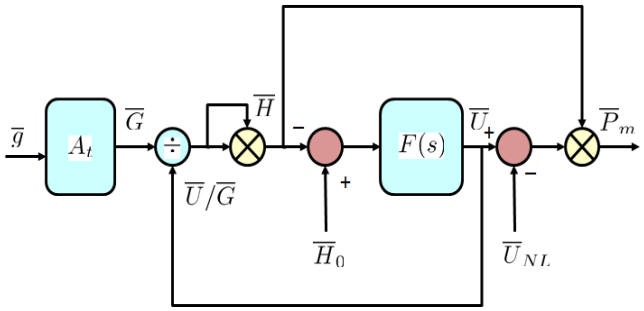


Fig. 3. Nonlinear hydraulic turbine-penstock model

Table 1. Hydraulic power plant values

Generator rating 140 MVA	Turbine rating 127.4 MW
$L = 250$	$A = 11.15$
$H_r = 165$	$Q_r = 80$
$\overline{g}_{FL} = 1.1$	$\overline{g}_{NL} = 0.1$

4. SYNCHRONOUS GENERATOR MODEL

The system considered is illustrated in Fig. 4, where one port is connected to the generator terminals and the second port is connected to a voltage source $E_b\angle 0$ Padiyar (2008).

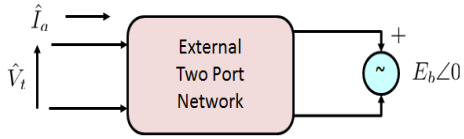


Fig. 4. External two port network

The main synchronous generator equations are:

$$\frac{d\delta}{dt} = \omega_B(S_m - S_{mo}) \tag{2}$$

$$\frac{dS_m}{dt} = \frac{1}{2h}\left[-D(S_m - S_{mo}) + T_m - T_e\right] \tag{3}$$

$$\frac{dE_q^{'}}{dt} = \frac{1}{T_{do}^{'}}\left[-E_q^{'} + (x_d - x_d^{'})i_d + E_{fd}\right] \tag{4}$$

$$\frac{dE_d^{'}}{dt} = \frac{1}{T_{qo}^{'}}\left[-E_d^{'} + (x_q - x_q^{'})i_q\right] \tag{5}$$

The electrical torque T_e is given by:

$$T_e = E_d^{'}i_d + E_q^{'}i_q + (x_d^{'} - x_q^{'})i_q i_d \tag{6}$$

Considering a lossless network, the stator algebraic equations and the network equations are expressed by:

$$E_q^{'} + x_d^{'}i_d - R_a i_q = V_q \tag{7}$$

$$E_d^{'} - x_q^{'}i_q - R_a i_d = V_d \tag{8}$$

$$V_q = -x_e i_d + E_b\cos(\delta) \tag{9}$$

$$V_d = x_e i_q - E_b\sin(\delta) \tag{10}$$

Solving the Equations (7)-(10) and doing $R_a = 0$, currents are obtained as:

$$i_d = \frac{E_b\cos(\delta) - E_q^{'}}{x_e + x_d^{'}} \tag{11}$$

$$i_q = \frac{E_b\sin(\delta) + E_d^{'}}{x_e + x_q^{'}} \tag{12}$$

For more details and values, review Padiyar (2008) and Kundur (1993).

5. MODELLING OF GOVERNORS

The primary control function involves feedback of speed error to control the position of the gate. In order to ensure a satisfactory and stable parallel operation of multiple units, the speed controller is provided of droop characteristic. The purpose of droop is ensuring equitable load sharing between the generating units. It is important to mention that the relationship between the mechanical power and the opening of the gate of hydroelectric turbines is initially inverse due to the inertia of water. Therefore, governor needs to provide a transient droop in speed controls to limit the overshoot of turbine gate servomotor during a transient condition Naghizadeh et al. (2012).

5.1 Hydraulic mechanical governor

These governors perform their functions through the use of mechanical and hydraulic components Kundur (1993). The typical structure is illustrated in Fig. 5.

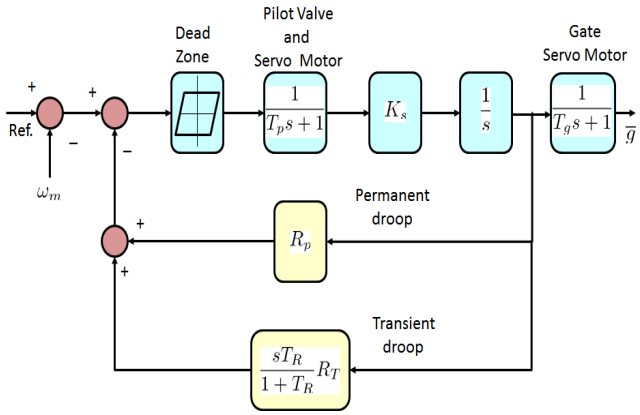


Fig. 5. Hydraulic mechanical governor

The transfer function of transient droop compensation is given by:

$$G_c(s) = R_T\frac{sT_R}{1 + sT_R} \tag{13}$$

The steady state performance can be determined by the permanent droop R_p. This droop is typically set at the range of 0.03 to 0.06. The parameters of hydraulic-mechanical governor that can be tuned to control the dynamic performance of the generating unit are T_R, R_T, and K_s. For stable operation under islanding conditions (worst case), the choice for the transient droop R_T and reset time T_R is given by:

$$R_T = [2.3 - (T_w - 1.0)0.15]\frac{T_w}{2h} \tag{14}$$

$$T_R = [5.0 - (T_w - 1.0)0.50]\,T_w \tag{15}$$

The above settings provide good performance during the most severe isolated conditions and slow response during loading in normal interconnected operation, also K_s should be set as high as possible Kundur (1993).

5.2 Two Degree of Freedom governor proposal

The structure of the TDOF controller is presented in Figure 6.

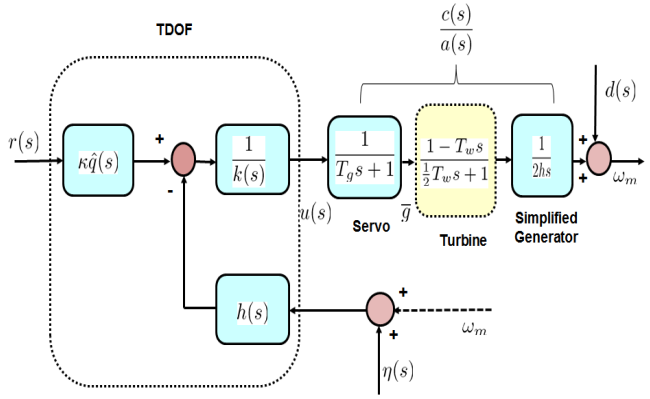


Fig. 6. TDOF controller structure

The gate servo motor, turbine and simplified generator model to be controlled are represented by the minimal, strictly proper rational transfer function given by:

$$G(s) = \frac{c(s)}{a(s)} \tag{16}$$

where polynomials $a(s)$ and $c(s)$ are co-prime, with degrees n and m $(m < n)$. The inputs $d(s)$ and $\eta(s)$ corresponds to disturbance and sensor noise signal. The polynomials $h(s)$, $k(s)$ and $\hat{q}(s)$ of degrees $n-1$ [1] are the controller functions.

The closed-loop transfer function from $r(s)$ to $\omega_m(s)$, without disturbance $d(t)$ and sensor noise signal $\eta(t)$ is:

$$\frac{\omega_m(s)}{r(s)} = \kappa\frac{c(s)\hat{q}(s)}{a(s)k(s)+c(s)h(s)} = \kappa\frac{c(s)\hat{q}(s)}{\delta(s)} \tag{17}$$

where $\kappa \in \mathcal{R}$ and $\delta(s)$ represents the closed-loop polynomial.

The controller guarantees a good output regulation, by positioning per design the roots of $\delta(s)$ far enough into the left half s-plane. However, the design could increase the system bandwidth sufficiently to produce hydraulic turbine control effort $u(s)$ saturation; in our case the commands cannot exceed the physical design limits. Complementarily, one way of obtaining a desirable output regulation, without requiring an excessive control effort signal is to design the turbine controller minimizing the LQR Wolovich (1994) performance index, which is:

$$J_{LQR} = \int_0^\infty \{\rho(r(t)-\omega_m(t))^2 + u(t)^2\}dt \tag{18}$$

The excessive output excursions and the control effort required to prevent such excursions can be obtained minimizing the Equation (18). The adjustable weighting factor $\rho \in \mathcal{R}^+$ can be used to obtain appropriate trade-offs between these two conflicting goals. The time solution of (18) corresponds to matrix gains calculated by the Riccati equation and the equivalent solution in the frequency domain is known as Spectral Factorization Wolovich (1994).

[1] The order of polynomials is based on Diophantine equation

5.3 Spectral Factorization

Since the polynomials $a(s)$ and $c(s)$ have real coefficients, that is:

$$\begin{aligned} a(j\omega)a(-j\omega) &= |a(j\omega)|^2 \geq 0 \\ c(j\omega)c(-j\omega) &= |c(j\omega)|^2 \geq 0 \;\forall \text{ real } \omega \end{aligned} \tag{19}$$

Therefore, it is possible to define:

$$\Delta(s) = a(s)a(-s) + \rho c(s)c(-s) \tag{20}$$

where $\Delta(s)$ is an even polynomial given by:

$$\Delta(s) = (-1)^n s^{2n} + \Delta_{2n-2}s^{2n-2} + \ldots + \Delta_2 s^2 + \Delta_0 \tag{21}$$

The $2n$ roots of $\Delta(s)$ are obtained with ρ varying from 0 to ∞ and represent a special root locus which is termed as root-square locus, i.e. if λ_j is a root of $\Delta(s)$ then $-\lambda_j$ is also a root of $\Delta(s)$. Consequently $\Delta(s)$ can be expressed by a spectral factorization as:

$$\Delta(s) = [\Delta(s)]^+[\Delta(s)]^- = \delta^{F^*}(s)\delta^{F^*}(-s) \tag{22}$$

where n stable roots of $[\Delta(s)]^+$ allows to obtain $\delta^{F^*}(s)$.

By duality, spectral factorization also can be used to obtain the n stable roots of $\delta^{H^*}(s)$, which is defined as the left half-plane spectral factor of:

$$\begin{aligned} \bar{\Delta}(s) &= a(s)a(-s) + \sigma c(s)c(-s) \\ &= [\bar{\Delta}(s)]^+[\bar{\Delta}(s)]^- \\ &= \delta^{H^*}(s)\delta^{H^*}(-s) \end{aligned} \tag{23}$$

where each choice of $\sigma \in \mathcal{R}^+$ in Equation (23) implies a corresponding $\delta^{H^*}(s)$, consequently, the pre-filter $\hat{q}(s)$ is calculated as function of n stable roots of $\delta^{H^*}(s)$. Then, for

$$k(s) = s^{n-1} + k_{n-2}s^{n-2} + \ldots + k_1 s + k_0 \tag{24}$$

and

$$h(s) = h_{n-1}s^{n-1} + \ldots + h_1 s + h_0 \tag{25}$$

it is possible to solve the Diophantine equation.

$$a(s)k(s) + c(s)h(s) = \delta^{F^*}(s)\hat{q}(s) \tag{26}$$

The Equation (26) can be written as:

$$\mathcal{S}\mathcal{V} = \bar{\delta} \tag{27}$$

From Equation (27) are obtained the real coefficients of gains $h(s)$ and $k(s)$ Vargas et al. (2015).

6. DESIGN OF GOVERNORS

The transfer function used to design is given by:

$$\frac{1}{R_p}\left(\frac{1}{1+T_g s}\right)\left(\frac{1-T_w s}{1+0.5T_w s}\right)\left(\frac{1}{2hs}\right) \tag{28}$$

where the values used are presented in Table 2:

Table 2. Values of time constants and parameters

$T_g = 0.5$	$T_p = 0.00$	$T_w = 1$	$h = 3.5$	$R_p = 0.05$

For the values of Table 2, the design transfer function is:

$$\frac{-20s + 20}{1.75s^3 + 7s^2 + 7s} \tag{29}$$

The poles and zeros of Equation (29) are presented in Table 3.

Table 3. poles and zeros of design model

poles	-2.0000	-2.0000	0
zeros			1

6.1 Design of Hydraulic Mechanical

Considering the governor structure illustrated in the Fig. 5 (dead-band effects are not modelled), the compensation transient droop transfer function is:

$$G_c(s) = \frac{1 + T_R s}{1 + (\frac{R_T}{R_p})T_R s} \tag{30}$$

The values of R_T and T_R are calculated using the Equations (14) and (15) and are equal to 0.4 and 6 respectively.

6.2 Design of TDOF Governor

The design transfer function $\frac{c(s)}{a(s)}$ is given by Equation (29). Therefore, using Equations (22) and (23), the values of root-square locus for weighting factors $\rho = 1$ and $\sigma = 0.5$ are detailed in Table 4.

Table 4. The values of root-square locus

$\rho = 1$	$\sigma = 0.5$
$-2.7719 + 2.0371j$	$-2.4730 + 1.5970j$
$-2.7719 - 2.0371j$	$-2.4730 - 1.5970j$
$2.7719 + 2.0371j$	$2.4730 + 1.5970j$
$2.7719 - 2.0371j$	$2.4730 - 1.5970j$
$-0.9658 + 0.0000j$	$-0.9325 + 0.0000j$
$0.9658 + 0.0000j$	$0.9325 + 0.0000j$

The weighting factor ρ makes possible fulfilling with the closed loop stability requirements (fast pole location without actuator saturation) through the calculation of $\Delta(s)$ to obtain the polynomial $\delta^{F^*}(s)$ as:

$$\delta^{F^*}(s) = s^3 + 6.5096s^2 + 17.1873s + 11.4286 \tag{31}$$

at the same time, the weighting factor σ helps to fulfilling with performance requirements through the calculation of $\bar{\Delta}(s)$ to obtain the polynomial $\delta^{H^*}(s)$ as:

$$\delta^{H^*}(s) = s^3 + 5.8587s^2 + 13.2784s + 8.0812 \tag{32}$$

Considering Equation (32), the pre-filter $\hat{q}(s)$ is calculated as:

$$\hat{q}(s) = (s + real(2.4730 + 1.5970j)^2 \tag{33}$$

Finally, using the Diophantine Equation (27), the 2^{th} order polynomials $h(s)$ and $k(s)$ presented in Table 5 are obtained.

The gain κ was selected equal to 0.58, this value allows to close the steady-state error near of zero.

Table 5. $TDOF$ Controller gains

$\hat{q}(s)$	$h(s)$	$k(s)$
s^2	$-1.5339s^2$	s^2
$4.9460s$	$2.0608s$	$2.5461s$
6.1158	3.4947	27.1885

7. LINEAR ANALYSIS RESULTS

The results of the design and analysis of linear simulations are presented in the following.

7.1 Hydraulic mechanical governor results

Figure 7 illustrates the gain margin versus frequency. This result shows that the crossover frequency is lower than 4 radians per second (would be considered slow response).

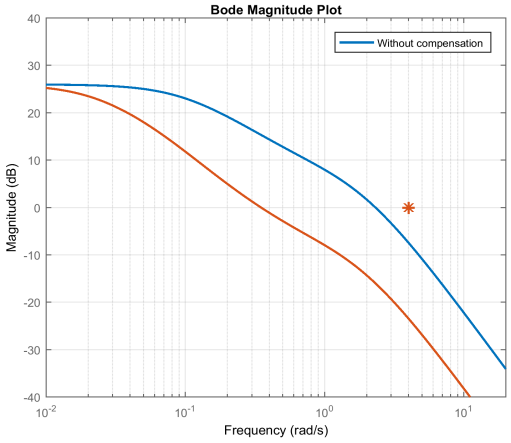


Fig. 7. Gain vs. frequency for hydraulic mechanical governor

Figure 8 illustrates the Nichols plot, where the horizontal and vertical axis of rhomboid represents the phase and gain margin respectively. The values of rhomboid corresponds to 45 degrees and 6 dB.

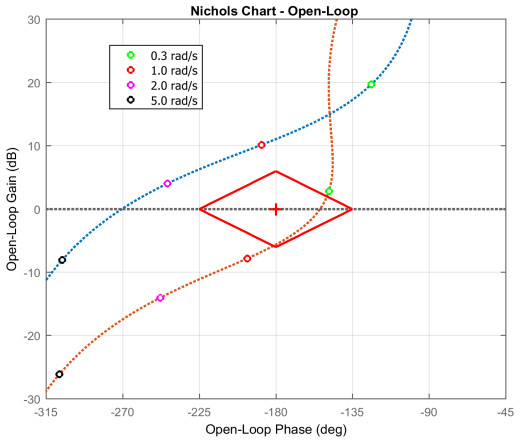


Fig. 8. Gain vs. phase for hydraulic mechanical governor

Our requirements corresponds to 45 degrees of phase margin and 6 dB of gain margin Kundur (1993).

7.2 Two Degree of Freedom results

Figure 9 illustrates the gain versus frequency response. This result shows that the crossover frequency in this case is near of 4 radians per second, this case is considered appropriate.

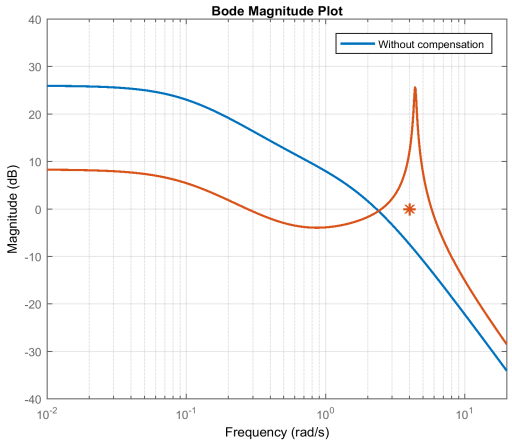


Fig. 9. Gain vs. frequency for TDOF governor

Figure 10 illustrates the Nichols diagram. The profile is due to composition of zeros and poles of open loop transfer function formed by the feedback gain, turbine and generator Alavi and Saif (2013).

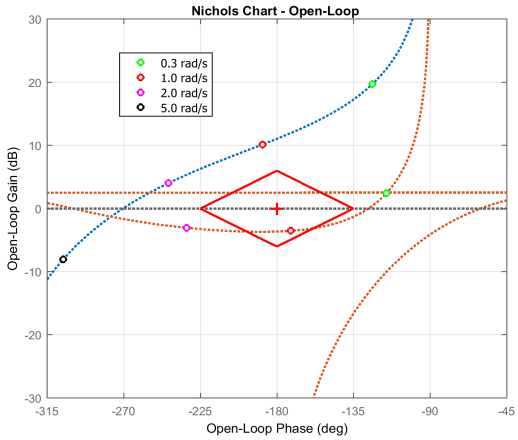


Fig. 10. Gain vs. phase for TDOF governor

For comparison purposes, the Table 6 presents the values of phase and gain margins of both governors (based in Nichols charts illustrated in Figures 8 and 10).

Table 6. Phase and gain margins

Margin	Without Comp.	Gain/Phase Comp.	TDOF
Gain [dB]	-9.85	5.6	4
Phase [^{o}]	-88.61	26	53

From the table it is possible to verify that TDOF governor has a better phase margin and hydraulic mechanical governor has a better gain margin, therefore both do not comply with the expected values.

Finally, Fig. 11 illustrates the simulation result obtained, using a step input in the speed for both governors.

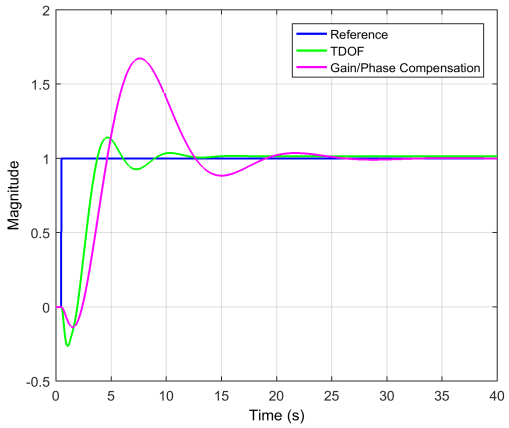


Fig. 11. Step input response

In this results it is evident that the mechanical hydraulic governor presents a greater overshoot.

8. NONLINEAR SIMULATION RESULTS

To verify the possible implementation of proposed governor, nonlinear simulations were executed under $\Delta\omega_m$ disturbances with amplitude of 20 radianos by second at 20 seconds. Obtained results for both governors are illustrated in Figures 12, 13, 14 and 15.

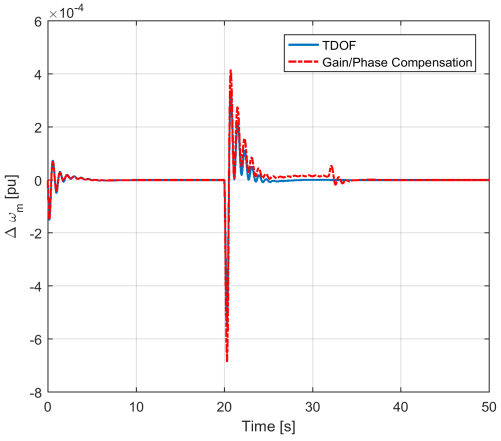


Fig. 12. Variation of speed deviation

These results allows to verify that the speed variation and the power angle reached after the disturbance is smaller for the Two Degree of Freedom structure, also, the terminal voltage is smaller compared to first governor. Finally, the proposed algorithm does not saturate the electrical torque.

9. CONCLUSIONS

This work proposes the use of the structure of two degrees of freedom as governor of hydro generators. As traditionally done, for project was used a linear model, parallel comparison with other proposed controller in the literature was done. The controllers obtained as well as their linear analysis and comparisons

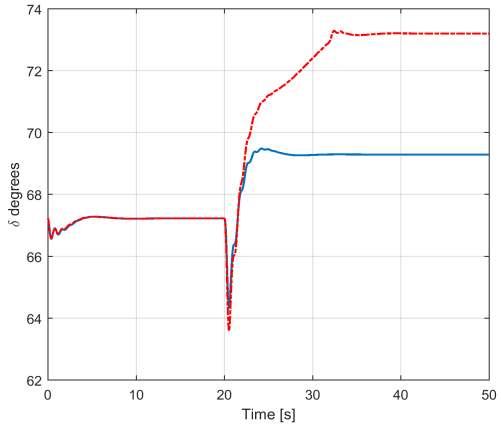


Fig. 13. Response of power angle δ

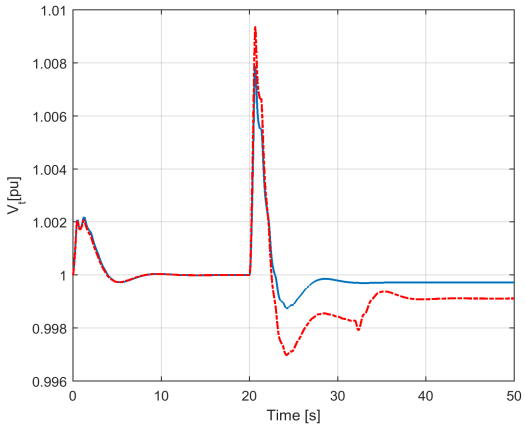


Fig. 14. Variation of terminal voltage

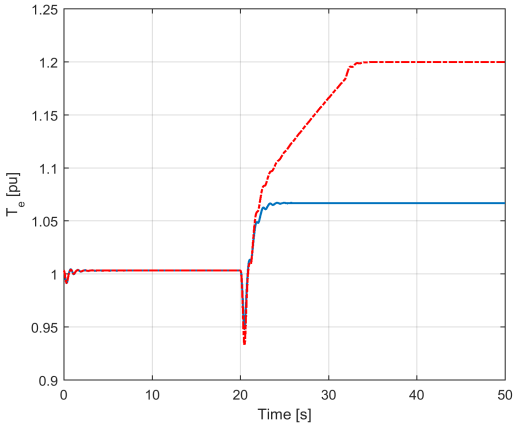


Fig. 15. Variation of electrical torque

were presented also. Finally, the initial nonlinear simulations results are presented; emphasizing that the initial results are satisfactory. However, these initial results are promising for continuing to research presented in this work.

ACKNOWLEDGEMENTS

The authors would like to thanks to Araraquara University and CNDC for support.

REFERENCES

Alavi, S. and Saif, M. (2013). On stability analysis by using Nyquist and Nichols Charts. University of Windsor, first edition.

Farmer, R.G. (2001). Power System Dynamics and Stability. Arizona State University, first edition.

IEEE, C.R. (1981). Excitation system models for power system stability studies. 494–509. IEEE Transactions on Power Apparatus and Systems.

Kundur, P. (1993). Power System Stability and Control. McGraw-Hill, first edition.

Machowski, J., Bialek, J., and Bumby, J. (2008). Power System Dynamics: Stability and Control. Wiley and Sons, first edition.

Naghizadeh, R.A., Jazebi, S., and Vahidi, B. (2012). Modeling hydro power plants and tuning hydro governors as an educational guideline. 1780–1790. International Review on Modelling and Simulations.

Padiyar, K. (2008). Power System Dynamics: Stability and Control. BS Publications, second edition.

Vargas, F.J.T., de Oliveira Moreira, F.J., and Paglione, P. (2015). Longitudinal stability and control augmentation with robustness and handling qualities requirements using the two degree of freedom controller. 1–11. Journal of the Brazilian Society of Mechanical Sciences and Engineering.

Vargas, F.J.T., de Pieri, E.R., and Castelan, E.B. (2004). Identification and friction compensation for an industrial robot using two degrees of freedom controllers. In ICARCV, 1146–1151. IEEE.

Vargas, F.J.T. and Paglione, P. (2015). Ferramentas de Algebra Computacional: Aplicações em Modelagem, Simulação e Controle para Engenharia. LTC, first edition.

Wolovich, W.A. (1994). Automatic Control Systems: Basic Analysis and Design. Oxford University Press, second edition.

Zhou, K. and Doyle, J.C. (1999). Essentials of Robust Control. Prentice Hall, first edition.

Appendix A. SYNCHRONOUS MACHINE VALUES

Table A.1. Synchronous machine values

$T_{do}^{'} = 6.66$	$T_{qo}^{'} = 0.44$	$x_d = 1.7572$	$x_d^{'} = 0.4245$
$x_q = 1.5845$	$x_q^{'} = 1.04$	$D = 0$	$R_a = 0$

Appendix B. IEEE 1. EXCITATION SYSTEM

In Fig. B.1 is illustrated the excitation model IEEE (1981).

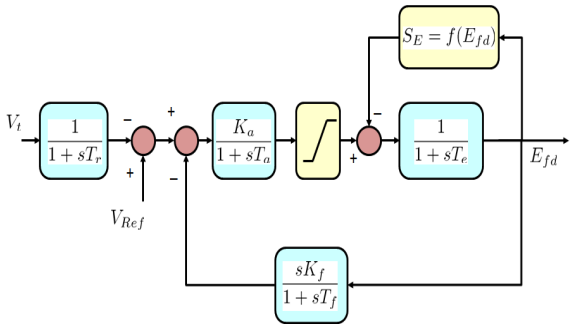


Fig. B.1. Excitation system Type IEEE 1

Table B.1. Excitation system values

$T_r = 0.01$	$T_a = 0.02$	$T_e = 1$
$T_f = 1$	$K_a = 400$	$K_f = 0.06$

SISO Pole Placement Algorithm: A Linear Transformation Approach

Nicolás E. Faedo * **Ernesto Aljinovic** ** **Virginia Mazzone** *

* *IACI - Departamento de Ciencia y Tecnología, Universidad Nacional de Quilmes, Roque Saenz Peña 352, Bernal, Buenos Aires, Argentina.*
** *Área de Matemática Superior - Departamento de Ciencia y Tecnología, Universidad Nacional de Quilmes, Roque Saenz Peña 352, Bernal, Buenos Aires, Argentina.*

Abstract: In this paper, an algorithm for SISO Pole Placement based on linear algebra concepts it's developed. This algorithm uses the knowledge of the degrees of certain polynomials associated to the *Internal Model Principle* and *Stable Zero-Pole cancellations* involved in the equation of the closed loop and it's coefficients, generating a linear system of equations for the desired closed loop poles in a systematic way.

Keywords: pole placement, SISO systems, linear algebra, diophantine equations

1. INTRODUCTION

The central problem in control is to find a way to act on a given process such that it behaves close to a desired behavior. Furthermore, this approximate behavior should be achieved in presence of uncertainty of the process and of uncontrollable external disturbances acting on the process. That means, given the closed loop of one degree of freedom shown in Figure 1, where the nominal model of the process to be controlled is $G_0(s)$, find a controller $K(s)$ that ensure that the nominal loop is stable and, if it's possible, to reach a desire behavior previously defined.

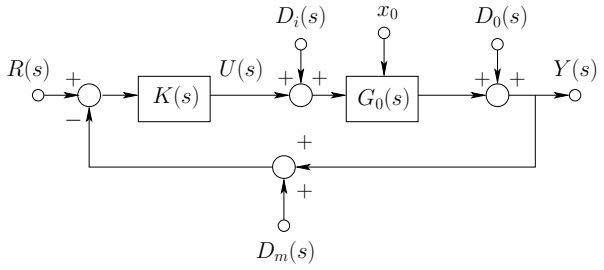


Fig. 1. Closed loop of one degree of freedom

In the loop shown in Figure 1 we use transfer functions and Laplace transforms to describe the relationships between signals in the loop, where $R(s)$ is the reference input, $U(s)$ is the control signal, $Y(s)$ is the output of the loop, $D_i(s)$ is the input disturbance, $D_0(s)$ is the output disturbance and $D_m(s)$ is the measurement noise. We also use x_0 to denote the initial conditions of the model. For linear time-invariant (LTI) systems, the nominal model and the controller can be written as

$$G_0(s) = \frac{B(s)}{A(s)} \qquad K(s) = \frac{P(s)}{L(s)}$$

The poles of the four sensitivity functions governing the closed loop belong to the same set, namely the roots of the characteristic equation $A(s)L(s) + B(s)P(s) = 0$. The poles have a deep impact on the dynamics of a transfer function; they define the stability of the loop. In this way, there exists a technique which deals with the choice of the roots of the characteristic equation, that is, given polynomials $A(s)$, $B(s)$ (defining the model) and given a polynomial $A_{cl}(s)$ (defining the desired location of closed loop poles), it is possible to find polynomials $P(s)$ and $L(s)$ such that

$$A(s)L(s) + B(s)P(s) = A_{cl}(s) \tag{1}$$

The Equation (1) is known as a Diophantine equation and the controller synthesis by solving it is known as pole placement. Polynomial Diophantine equations play a crucial role in the polynomial theory of control systems synthesis. Systems are described by input-output relations, similarly to the classical control techniques, however, the transfer functions are not regarded as functions of complex variable but as algebraic objects. Applications include closed loop pole placement (Kučera, 1993), minimum variance control (Hunt, 1993), LQ and LQG optimal compensators (Kučera, 1991) or adaptive and predictive control (Hunt, 1993). It is well known that, if the controller is biproper, the solution of the equation exists if

$$\deg\{P(s)\} = \deg\{L(s)\} \geq n - 1$$

with $n = \deg\{A(s)\}$. In this context, the minimum order controller is then of degree $n - 1$ and the condition on coprimeness between $A(s)$ y $B(s)$ is necessary to guarantee the existence and uniqueness of the solution (Sylvester theorem) (Goodwin et al., 2001). Solving this equation basically implies solving a linear system of equations, which involves a Sylvester matrix. A suitable and fast algorithm for invert this type of matrices was developed in (Li, 2011).

Many times control objective for the closed loop is to track a specific reference or reject a disturbance of a known frequency. In order to accomplish this we present a systematic way to solve the system equation obtained from using Internal Model Principle (IMP) defined for the first time in (Francis and Wonham, 1975), which establish

that the *reference or disturbance generating polynomial* (o simply *generating polynomial*) must be in the denominator of $K(s)$ (Goodwin et al., 2001). This can also be achieved by solving the Diophantine equation (1). Sometimes it is desirable to force the controller to cancel a subset of stable poles and/or zeros of the plant model, this is also taking into account by this systematization, which arises to an algorithm to solve this problem in an automatic way. This approach can be used for design adaptive controllers, or simply synthesize PID controllers.

2. LINEAR TRANSFORMATION APPROACH

2.1 Notation

Let $X(s) = x_n\, s^n + x_{n-1}\, s^{n-1} + \cdots + x_1\, s + x_0$ be a polynomial with real coefficients.

Notation 1. The set of all coefficients of $X(s)$ (in decreasing power order) is denoted by $C_X = \{x_n, x_{n-1}, \cdots, x_1, x_0\}$

Notation 2. The degree of $X(s)$ is denoted by $\deg\{X(s)\}$

Let $\mathbb{V}, \mathbb{W}$ be finite-dimensional vector spaces over a field K and choose bases $V = \{v_1, \ldots, v_m\}$ for $\mathbb{V}$ and $W = \{w_1, \ldots, w_n\}$ for $\mathbb{W}$.

Notation 3. The dimension of $\mathbb{V}$ is denoted by $\dim\{\mathbb{V}\}$.

Notation 4. Let $v^* \in \mathbb{V}$. The coordinates of v^* in the basis V are denoted by $(v^*)_V \in \mathbb{R}^m$.

Notation 5. Let $T : \mathbb{V} \longrightarrow \mathbb{W}$ be a linear transformation from $\mathbb{V}$ to $\mathbb{W}$. The matrix associated to T choosing bases V and W is denoted by T_{VW}.

Definition 1. The external direct sum of $\mathbb{V}$ and $\mathbb{W}$, denoted by $\mathbb{V}\tilde{\oplus}\mathbb{W}$ is defined as the set of all ordered pairs (v, w) with $v \in \mathbb{V}$ and $w \in \mathbb{W}$. Scalar multiplication is defined by $c(v, w) = (cv, cw)$ with $c \in K$, and addition is defined by $(v, w) + (v', w') = (v + v', w + w')$. One checks the other classical axioms for a vector space.

Note that the external direct sum of $\mathbb{V}$ and $\mathbb{W}$ can be expressed as the internal direct sum of $(\mathbb{V}, 0)$ and $(0, \mathbb{W})$. A basis for $\mathbb{V}\tilde{\oplus}\mathbb{W}$ is given by

$$\left\{\{(v_i, 0)\} \bigcup \{(0, w_j)\}\right\}$$

2.2 Pole Placement

Given the control loop of one degree of freedom as in Figure 1. Let $G_0(s)$ be the process nominal model and $K(s)$ the biproper controller defined as

$$G_0(s) = \frac{B(s)}{A(s)} \equiv \frac{B}{A} \qquad K(s) = \frac{P(s)}{L(s)} \equiv \frac{P}{L}$$

where

$$\begin{aligned} A &= a_n\, s^n + a_{n-1} s^{n-1} + \cdots + a_1\, s + a_0 \\ B &= b_m\, s^m + b_{m-1} s^{m-1} + \cdots + b_1\, s + b_0 \\ P &= p_{n-1}\, s^{n-1} + p_{n-2} s^{n-2} + \cdots + p_1\, s + p_0 \\ L &= l_{n-1}\, s^{n-1} + l_{n-2} s^{n-2} + \cdots + l_1\, s + l_0 \end{aligned}$$

The degrees of the polynomials are as it follows

$$\begin{aligned} \deg\{A\} &= n \\ \deg\{B\} &= m \qquad m \leq n \\ \deg\{P\} &= n - 1 \\ \deg\{L\} &= n - 1 \end{aligned}$$

The closed loop polynomial $A_{cl}(s) \equiv A_{cl}$ is given by the following Diophantine equation

$$A\, L + B\, P = A_{cl}$$

where

$$\deg\{A_{cl}\} = \deg\{A\} + \deg\{L\} = 2n - 1$$

and so

$$A_{cl} = c_{2n-1}\, s^{2n-1} + c_{2n-2}\, s^{2n-2} + \cdots + c_1\, s + c_0$$

Let $\mathbb{V}_l$, $\mathbb{V}_p$, $\mathbb{W}$ be finite-dimensional vector spaces over $\mathbb{R}$ such as

$$\begin{aligned} \mathbb{V}_l &= \text{span}\{s^{n-1}, s^{n-2}, \ldots, s, 1\} \\ \mathbb{V}_p &= \text{span}\{s^{n-1}, s^{n-2}, \ldots, s, 1\} \\ \mathbb{W} &= \text{span}\{s^{2n-1}, s^{2n-2}, \ldots, s, 1\} \end{aligned}$$

Notice that $L \in \mathbb{V}_l$, $P \in \mathbb{V}_p$ and $A_{cl} \in \mathbb{W}$. Although in this case $\mathbb{V}_l$ is exactly the same space as $\mathbb{V}_p$, we keep the subscripts for the sake of clarity. Let $\mathbb{V}_l\tilde{\oplus}\mathbb{V}_p$ be the external direct sum of $\mathbb{V}_l$ and $\mathbb{V}_p$. Let V and W be a basis for $\mathbb{V}_l\tilde{\oplus}\mathbb{V}_p$ and $\mathbb{W}$ respectively, such as

$$V = \left\{\{(s^{n-1}, 0), (s^{n-2}, 0), \ldots, (s, 0), (1, 0)\} \bigcup \{(0, s^{n-1}), (0, s^{n-2}), \ldots, (0, s), (0, 1)\}\right\} \tag{2}$$

$$W = \{s^{2n-1}, s^{2n-2}, \ldots, s, 1\} \tag{3}$$

Define the linear transformation Φ as it follows

$$\begin{aligned} \Phi &: \mathbb{V}_l\tilde{\oplus}\mathbb{V}_p \longrightarrow \mathbb{W} \\ \Phi\{(l, p)\} &\mapsto A\, l + B\, p \end{aligned}$$

The construction of the matrix associated to the linear transformation Φ in the bases V and W starts by computing the transformation of every vector of V

$$\begin{aligned} \Phi\{(s^{n-1}, 0)\} &\mapsto A\, s^{n-1} = a_n\, s^{2n-1} + \cdots + a_0\, s^{n-1} \\ \Phi\{(s^{n-2}, 0)\} &\mapsto A\, s^{n-2} = a_n\, s^{2n-2} + \cdots + a_0\, s^{n-2} \\ &\vdots \qquad\qquad \vdots \\ \Phi\{(s, 0)\} &\mapsto A\, s = a_n\, s^{n+1} + \cdots + a_0\, s \\ \Phi\{(1, 0)\} &\mapsto A = a_n\, s^n + \cdots + a_0 \\ \Phi\{(0, s^{n-1})\} &\mapsto B\, s^{n-1} = b_m\, s^{m+n-1} + \cdots + b_0\, s^{n-1} \\ \Phi\{(0, s^{n-2})\} &\mapsto B\, s^{n-2} = b_m\, s^{m+n-2} + \cdots + b_0\, s^{n-2} \\ &\vdots \qquad\qquad \vdots \\ \Phi\{(0, s)\} &\mapsto B\, s = b_m\, s^{m+1} + \cdots + b_0\, s \\ \Phi\{(0, 1)\} &\mapsto B = b_m\, s^m + \cdots + b_0 \end{aligned} \tag{4}$$

Getting coordinates in basis W yields

$$\begin{aligned} (A\, s^{n-1})_W &= (C_A, 0, 0, \ldots\ldots\ldots\ldots, 0, 0, 0) \\ (A\, s^{n-2})_W &= (0, C_A, 0, \ldots\ldots\ldots\ldots, 0, 0, 0) \\ &\vdots \\ (A\, s)_W &= (0, 0, 0, \ldots\ldots\ldots\ldots, 0, C_A, 0) \\ (A)_W &= (0, 0, 0, \ldots\ldots\ldots\ldots, 0, 0, C_A) \\ (B\, s^{n-1})_W &= (\overbrace{0, \ldots, 0}^{n-m}, C_B, 0, 0, \ldots, 0, 0, 0) \\ (B\, s^{n-2})_W &= (0, \ldots, 0, 0, C_B, 0, \ldots, 0, 0, 0) \\ &\vdots \\ (B\, s)_W &= (0, \ldots, 0, 0, 0, 0, \ldots, 0, C_B, 0) \\ (B)_W &= (0, \ldots, 0, 0, 0, 0, \ldots, 0, 0, C_B) \end{aligned} \tag{5}$$

Notice that every vector in $\mathbb{R}^{2n}$ defined above it's a shift of the coefficients of A and B polynomials respectively. Define

the following submatrices $\xi_A \in \mathbb{R}^{2n\times n}$ and $\xi_B \in \mathbb{R}^{2n\times n}$ (in columns)

$$\xi_A = \underbrace{\left[\, (A\, s^{n-1})_W^T \ \cdots \ (A)_W^T \,\right]}_{\dim\{\mathbb{V}_l\}=n}$$

$$\xi_B = \underbrace{\left[\, (B\, s^{n-1})_W^T \ \cdots \ (B)_W^T \,\right]}_{\dim\{\mathbb{V}_p\}=n}$$

Then, the matrix associated to the transformation in the bases V and W (in columns)

$$\Phi_{VW} = [\, \xi_A | \xi_B \,]$$

Where $\Phi_{VW} \in \mathbb{R}^{2n\times 2n}$ is a *Sylvester Matrix* associated to the polynomials A and B, and $|\Phi_{VW}| \neq 0$ because A and B are coprime. Moreover, because of the shifting property of the columns of ξ_A and ξ_B (and knowing the dimensions of $\mathbb{V}_l$, $\mathbb{V}_p$ and $\mathbb{W}$) constructing the matrix it's straightforward. With Φ_{VW} computed (which is systematic) knowing the coefficients of L and P it's reduced to solve the following linear system of equations:

$$[\, \xi_A | \xi_B \,] \begin{bmatrix} C_L \\ C_P \end{bmatrix} = C_{A_{cl}} \longrightarrow \begin{bmatrix} C_L \\ C_P \end{bmatrix} = [\, \xi_A | \xi_B \,]^{-1}\, C_{A_{cl}}$$

2.3 Internal Model Principle

Adding the Internal Model Principle to the loop, given by the *generating polynomial* $\Gamma(s) \equiv \Gamma$ where

$$\deg\{\Gamma\} = q$$

the pole placement problem can be reformulated: the *generating polynomial* must appear as part of the denominator of the controller. To accomplish that goal, one chooses

$$L = \Gamma \bar{L}$$

and the closed loop equation can be rewritten as

$$\bar{A}\, \bar{L} + B\, P = A_{cl} \quad \text{where} \quad \bar{A} = \Gamma\, A$$

including Γ inside the term that represents the denominator of the plant, creating an equivalent model of degree $\bar{n} = n + q$. Now, using the same criterion of design a biproper controller with one degree less than the plant:

$$\begin{aligned} \deg\{P\} &= \bar{n} - 1 = n + q - 1 \\ \deg\{L\} &= \bar{n} - 1 = n + q - 1 \\ \deg\{A_{cl}\} &= 2n + q - 1 \end{aligned}$$

and

$$\deg\{\bar{L}\} = \deg\{L\} - \deg\{\Gamma\} = n - 1$$

Let ${}^{\mathrm{I}}\mathbb{V}_{\bar{l}}$, ${}^{\mathrm{I}}\mathbb{V}_p$, ${}^{\mathrm{I}}\mathbb{W}$ be vector spaces over $\mathbb{R}$ such as

$$\begin{aligned} {}^{\mathrm{I}}\mathbb{V}_{\bar{l}} &= \mathrm{span}\{s^{n-1}, s^{n-2}, \ldots, s, 1\} \\ {}^{\mathrm{I}}\mathbb{V}_p &= \mathrm{span}\{s^{n+q-1}, s^{n+q-2}, \ldots, s, 1\} \\ {}^{\mathrm{I}}\mathbb{W} &= \mathrm{span}\{s^{2n+q-1}, s^{2n+q-2}, \ldots, s, 1\} \end{aligned}$$

So that $\bar{L} \in {}^{\mathrm{I}}\mathbb{V}_{\bar{l}}$, $P \in {}^{\mathrm{I}}\mathbb{V}_p$ and $A_{cl} \in {}^{\mathrm{I}}\mathbb{W}$. Let ${}^{\mathrm{I}}\mathbb{V}_{\bar{l}} \tilde{\oplus} {}^{\mathrm{I}}\mathbb{V}_p$ be the external direct sum of ${}^{\mathrm{I}}\mathbb{V}_{\bar{l}}$ and ${}^{\mathrm{I}}\mathbb{V}_p$. Construct bases ${}^{\mathrm{I}}V$ and ${}^{\mathrm{I}}W$ in the same way as in (2) and (3). Define the linear transformation ${}^{\mathrm{I}}\Phi$ as it follows

$$\begin{aligned} {}^{\mathrm{I}}\Phi &: {}^{\mathrm{I}}\mathbb{V}_{\bar{l}} \tilde{\oplus} {}^{\mathrm{I}}\mathbb{V}_p \longrightarrow \mathbb{W} \\ {}^{\mathrm{I}}\Phi\{(\bar{l}, p)\} &\mapsto \bar{A}\, \bar{l} + B\, p \end{aligned} \tag{6}$$

Computing the corresponding maps to every vector in ${}^{\mathrm{I}}V$ in the same way as in (4) and getting it's coordinates in the basis ${}^{\mathrm{I}}W$ as in (5), construct the submatrices ${}^{\mathrm{I}}\xi_{\bar{A}} \in \mathbb{R}^{2n+q\,\times\, n}$ and ${}^{\mathrm{I}}\xi_B \in \mathbb{R}^{2n+q\,\times\, n+q}$ as it follows (in columns)

$${}^{\mathrm{I}}\xi_{\bar{A}} = \underbrace{\left[\, (\bar{A}\, s^{n-1})_{{}^{\mathrm{I}}W}^T \ \cdots \ (\bar{A})_{{}^{\mathrm{I}}W}^T \,\right]}_{\dim\{{}^{\mathrm{I}}\mathbb{V}_{\bar{l}}\}=n}$$

$${}^{\mathrm{I}}\xi_B = \underbrace{\left[\, (B\, s^{n+q-1})_{{}^{\mathrm{I}}W}^T \ \cdots \ (B)_{{}^{\mathrm{I}}W}^T \,\right]}_{\dim\{{}^{\mathrm{I}}\mathbb{V}_p\}=n+q}$$

Then, the matrix associated to the transformation in the bases ${}^{\mathrm{I}}V$ and ${}^{\mathrm{I}}W$ (in columns)

$${}^{\mathrm{I}}\Phi_{{}^{\mathrm{I}}V\,{}^{\mathrm{I}}W} = \left[\, {}^{\mathrm{I}}\xi_{\bar{A}} \middle| {}^{\mathrm{I}}\xi_B \,\right]$$

Where ${}^{\mathrm{I}}\Phi_{{}^{\mathrm{I}}V\,{}^{\mathrm{I}}W} \in \mathbb{R}^{2n+q\,\times\, 2n+q}$ is a *Sylvester Matrix* associated to the polynomials $\bar{A}$ and B; and $|{}^{\mathrm{I}}\Phi_{{}^{\mathrm{I}}V\,{}^{\mathrm{I}}W}| \neq 0$ because $\bar{A}$ and B are coprime.

Example 1. Given

$$G_0(s) = \frac{s+1}{s^2 + 4\,s + 4}$$

we aim to design a biproper controller applying the Internal Model Principle with the *generating polynomial* $\Gamma = s(s^2+1)$.

The degrees of the polynomials

$$\begin{aligned} \deg\{A\} &= n = 2 \\ \deg\{B\} &= m = 1 \\ \deg\{\Gamma\} &= q = 3 \\ \deg\{L\} = \deg\{P\} &= n + q - 1 = 4 \\ \deg\{\bar{L}\} &= n - 1 = 1 \\ \deg\{A_{cl}\} &= 2n + q - 1 = 6 \end{aligned}$$

The corresponding dimensions

$$\dim\{{}^{\mathrm{I}}\mathbb{V}_{\bar{l}}\} = 2, \quad \dim\{{}^{\mathrm{I}}\mathbb{V}_p\} = 5, \quad \dim\{{}^{\mathrm{I}}\mathbb{W}\} = 7 \tag{7}$$

Computing $\bar{A}$ yields

$$\begin{aligned} \bar{A} &= \Gamma\, A = s^5 + 4\,s^4 + 5\,s^3 + 4\,s^2 + 4\,s \\ C_{\bar{A}} &= \{1, 4, 5, 4, 4, 0\} \end{aligned}$$

After defining the transformation ${}^{\mathrm{I}}\Phi$ as in (6) the corresponding ${}^{\mathrm{I}}\xi_{\bar{A}} \in \mathbb{R}^{7\times 2}$ and ${}^{\mathrm{I}}\xi_B \in \mathbb{R}^{7\times 5}$ (which are shifts of the coefficients of the polynomials $\bar{A}$ and B according to the dimensions stated in (7))

$${}^{\mathrm{I}}\xi_{\bar{A}} = \underbrace{\begin{bmatrix} 1 & 0 \\ 4 & 1 \\ 5 & 4 \\ 4 & 5 \\ 4 & 4 \\ 0 & 4 \\ 0 & 0 \end{bmatrix}}_{\dim\{{}^{\mathrm{I}}\mathbb{V}_{\bar{l}}\}} \qquad {}^{\mathrm{I}}\xi_B = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\dim\{{}^{\mathrm{I}}\mathbb{V}_p\}}$$

Computing the coefficients of $\bar{L}$ and P involves the following linear system

$$\begin{bmatrix} \bar{l}_1 \\ \bar{l}_0 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 1 & 1 & 0 & 0 & 0 & 0 \\ 5 & 4 & 1 & 1 & 0 & 0 & 0 \\ 4 & 5 & 0 & 1 & 1 & 0 & 0 \\ 4 & 4 & 0 & 0 & 1 & 1 & 0 \\ 0 & 4 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{-1}}_{{}^{\mathrm{I}}\Phi^{-1}_{{}^{\mathrm{I}}V\,{}^{\mathrm{I}}W}} \begin{bmatrix} c_6 \\ c_5 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix}$$

Where $C_{A_{cl}} = \{c_6, c_5, c_4, c_3, c_2, c_1, c_0\}$ are the coefficients of the desired closed loop polynomial A_{cl}. If we choose $A_{cl} = (s+3)^6$, then the controller transfer function is given by

$$K(s) = \frac{45s^4 + 209s^3 + 482s^2 + 853s + 729}{s(s^2+1)(s-31)} \tag{8}$$

The steady state response of the closed loop of Figure 1 is indeed the one chosen, but the transient response is affected by the dynamics of the zeros of the controller and the nominal model. The behavior of this zeros is undesirable if they are located at the right side of the poles in the left semi-plane of the complex plane (Seron et al., 1997). This can be sometimes avoided if we propose *Stable Zero-Pole cancellations* between the controller and the plant, as we explain in the following section.

2.4 Stable Zero-Pole cancellations

In addition to the implementation of the Internal Model Principle, it's from interest to obtain a systematic way to perform *Stable Zero-Pole cancellations*. To achieve that goal, the controller denominator(numerator) must include the pole(zero) dynamics to cancel. Suppose that the stable dynamics to cancel are represented by two polynomials $\alpha(s) \equiv \alpha$ (poles) and $\beta(s) \equiv \beta$ (zeros) such that

$$A = \alpha\,\tilde{A}$$
$$B = \beta\,\tilde{B}$$

where

$$\deg\{\alpha\} = w$$
$$\deg\{\beta\} = z$$

The Diophantine equation associated to the closed loop

$$A\,L + B\,P = A_{cl} \longrightarrow \alpha\,\tilde{A}\,L + \beta\,\tilde{B}\,P = A_{cl} \tag{9}$$

Choosing $L = \beta\,\tilde{L}$ and $P = \alpha\,\tilde{P}$ the equation (9) can be expressed as

$$\tilde{A}\,\tilde{L} + \tilde{B}\tilde{P} = \tilde{A}_{cl}$$

with $A_{cl} = \alpha\,\beta\,\tilde{A}_{cl}$ so that the remaining closed loop poles after the cancellations ($\tilde{A}_{cl}$) can be chose arbitrarily. The corresponding degrees using the same design criterion (biproper controller of one degree less than the plant) remains as follows

$$\deg\{\tilde{L}\} = n - z - 1$$
$$\deg\{\tilde{P}\} = n - w - 1$$
$$\deg\{\tilde{A}_{cl}\} = 2n - z - w - 1$$

Let ${}^{\mathrm{Z}}\mathbb{V}_{\tilde{l}}$, ${}^{\mathrm{Z}}\mathbb{V}_{\tilde{p}}$, ${}^{\mathrm{Z}}\mathbb{W}$ be vector spaces over $\mathbb{R}$ such as

$${}^{\mathrm{Z}}\mathbb{V}_{\tilde{l}} = \mathrm{span}\{s^{n-z-1}, s^{n-z-2}, \ldots, s, 1\}$$
$${}^{\mathrm{Z}}\mathbb{V}_{\tilde{p}} = \mathrm{span}\{s^{n-w-1}, s^{n-w-2}, \ldots, s, 1\}$$
$${}^{\mathrm{Z}}\mathbb{W} = \mathrm{span}\{s^{2n-z-w-1}, s^{2n-z-w-2}, \ldots, s, 1\}$$

So that $\tilde{L} \in {}^{\mathrm{Z}}\mathbb{V}_{\tilde{l}}$, $\tilde{P} \in {}^{\mathrm{Z}}\mathbb{V}_{\tilde{p}}$ and $\tilde{A}_{cl} \in {}^{\mathrm{Z}}\mathbb{W}$. Let ${}^{\mathrm{Z}}\mathbb{V}_{\tilde{l}} \tilde{\oplus} {}^{\mathrm{Z}}\mathbb{V}_{\tilde{p}}$ be the external direct sum of ${}^{\mathrm{Z}}\mathbb{V}_{\tilde{l}}$ and ${}^{\mathrm{Z}}\mathbb{V}_{\tilde{p}}$. Construct bases ${}^{\mathrm{Z}}V$ and ${}^{\mathrm{Z}}W$ in the same way as in (2) and (3). Define the linear transformation ${}^{\mathrm{Z}}\Phi$ as it follows

$${}^{\mathrm{Z}}\Phi : {}^{\mathrm{Z}}\mathbb{V}_{\tilde{l}} \tilde{\oplus} {}^{\mathrm{Z}}\mathbb{V}_{\tilde{p}} \longrightarrow {}^{\mathrm{Z}}\mathbb{W}$$
$${}^{\mathrm{Z}}\Phi\{(\tilde{l}, \tilde{p})\} \mapsto \tilde{A}\,\tilde{l} + \tilde{B}\,\tilde{p}$$

Using the same criterion as in (4) and (5) with bases ${}^{\mathrm{Z}}V$ and ${}^{\mathrm{Z}}W$ respectively, the submatrices ${}^{\mathrm{Z}}\xi_{\tilde{A}} \in \mathbb{R}^{2n-z-w \times n-z}$ and ${}^{\mathrm{Z}}\xi_{\tilde{B}} \in \mathbb{R}^{2n-z-w \times n-w}$ (in columns)

$${}^{\mathrm{Z}}\xi_{\tilde{A}} = \underbrace{\left[\, (\tilde{A}\,s^{n-z-1})^T_{{}^{\mathrm{Z}}W} \;\cdots\; (\tilde{A})^T_{{}^{\mathrm{Z}}W} \,\right]}_{\dim\{{}^{\mathrm{Z}}\mathbb{V}_{\tilde{l}}\}=n-z}$$

$${}^{\mathrm{Z}}\xi_{\tilde{B}} = \underbrace{\left[\, (\tilde{B}\,s^{n-w-1})^T_{{}^{\mathrm{Z}}W} \;\cdots\; (\tilde{B})^T_{{}^{\mathrm{Z}}W} \,\right]}_{\dim\{{}^{\mathrm{Z}}\mathbb{V}_{\tilde{p}}\}=n-w}$$

Then, the matrix associated to the transformation in the bases ${}^{\mathrm{Z}}V$ and ${}^{\mathrm{Z}}W$

$${}^{\mathrm{Z}}\Phi_{{}^{\mathrm{Z}}V\,{}^{\mathrm{Z}}W} = \left[\, {}^{\mathrm{Z}}\xi_{\tilde{A}} \,|\, {}^{\mathrm{Z}}\xi_{\tilde{B}} \,\right]$$

Where ${}^{\mathrm{Z}}\Phi_{{}^{\mathrm{Z}}V\,{}^{\mathrm{Z}}W} \in \mathbb{R}^{2n-z-w \times 2n-z-w}$ is a *Sylvester Matrix* associated to the polynomials $\tilde{A}$ and $\tilde{B}$; and $|{}^{\mathrm{Z}}\Phi_{{}^{\mathrm{Z}}V\,{}^{\mathrm{Z}}W}| \neq 0$ because $\tilde{A}$ and $\tilde{B}$ are coprime.

3. DEVELOPMENT OF THE ALGORITHM

Combining the criterion developed in Section 2.3 and Section 2.4 one can construct a linear transformation that takes into account the Internal Model Principle and Stable Zero-Pole cancellations at the same time, providing a systematic way to obtain the matrix involved in the determination of the coefficients of the desired closed loop polynomial. Choosing

$$\begin{aligned} L &= \Gamma\,\beta L^* \\ P &= \alpha\,P^* \\ A_{cl} &= \alpha\,\beta A^*_{cl} \end{aligned} \tag{10}$$

The corresponding Diophantine equation remains as follows

$$A^*L^* + B^*P^* = A^*_{cl}$$

where

$$A^* = \frac{\Gamma}{\alpha}\,A \quad \text{and} \quad B^* = \frac{1}{\beta}\,B \tag{11}$$

The corresponding degrees are

$$\deg\{L^*\} = n - z - 1$$
$$\deg\{P^*\} = n + q - w - 1$$
$$\deg\{A^*_{cl}\} = 2n + q - z - w - 1$$

Let ${}^*\mathbb{V}_{l^*}$, ${}^*\mathbb{V}_{p*}$, ${}^*\mathbb{W}$ be vector spaces over $\mathbb{R}$ such as

$${}^*\mathbb{V}_{l^*} = \mathrm{span}\{s^{n-z-1}, s^{n-z-2}, \ldots, s, 1\}$$
$${}^*\mathbb{V}_{p^*} = \mathrm{span}\{s^{n+q-w-1}, s^{n+q-w-2}, \ldots, s, 1\}$$
$${}^*\mathbb{W} = \mathrm{span}\{s^{2n+q-z-w-1}, s^{2n+q-z-w-2}, \ldots, s, 1\}$$

So that $L^* \in {}^*\mathbb{V}_{l^*}$, $P^* \in {}^*\mathbb{V}_{p^*}$ and $A^*_{cl} \in {}^*\mathbb{W}$. Let ${}^*\mathbb{V}_{l^*} \tilde{\oplus} {}^*\mathbb{V}_{p^*}$ be the external direct sum of ${}^*\mathbb{V}_{l^*}$ and ${}^*\mathbb{V}_{p^*}$. Construct bases ${}^*V$ and ${}^*W$ in the same way as in (2) and (3). Define the linear transformation ${}^*\Phi$ as it follows

$${}^*\Phi : {}^*\mathbb{V}_{l^*} \tilde{\oplus} {}^*\mathbb{V}_{p^*} \longrightarrow {}^*\mathbb{W}$$
$${}^*\Phi\{(l^*, p^*)\} \mapsto A^*\,l^* + B^*\,p^*$$

Using the same criterion as in (4) and (5) with bases ${}^*V$ and ${}^*W$ respectively, the construction of the submatrices ${}^*\xi_{A^*} \in \mathbb{R}^{2n+q-z-w \times n-z}$ and ${}^*\xi_{B^*} \in \mathbb{R}^{2n+q-z-w \times n+q-w}$ (in columns)

$$\begin{aligned} {}^*\xi_{A^*} &= \underbrace{\left[\, (A^*\,s^{n-z-1})^T_{{}^*W} \;\cdots\; (A^*)^T_{{}^*W} \,\right]}_{\dim\{{}^*\mathbb{V}_{l^*}\}=n-z} \\ {}^*\xi_{B^*} &= \underbrace{\left[\, (B^*\,s^{n+q-w-1})^T_{{}^*W} \;\cdots\; (B^*)^T_{{}^*W} \,\right]}_{\dim\{{}^*\mathbb{V}_{p^*}\}=n+q-w} \end{aligned} \tag{12}$$

Then, the matrix associated to the transformation in the bases ${}^*V$ and ${}^*W$

$$ {}^*\Phi_{*V^*W} = [\, {}^*\xi_{A^*} \,|\, {}^*\xi_{B^*} \,] $$

Where $\Phi^*_{VW} \in \mathbb{R}^{2n+q-z-w \times 2n+q-z-w}$ is a *Sylvester Matrix* associated to the polynomials A^* and B^*.

In synthesis, the algorithm can be summarized in the following simple steps

(i) Choose the *generating polynomial* Γ and the stable dynamics to cancel α (poles) and β (zeros).
(ii) Compute A^* and B^* as in (11) and extract their corresponding coefficients.
(iii) Using the information of the degrees of the denominator of the plant A (n), the generating polynomial Γ (q), the desired pole cancellations α (w) and the desired zero cancellations β (z) construct the submatrices ${}^*\xi_{A^*}$ and ${}^*\xi_{B^*}$ performing the corresponding shifts to the coefficients of the polynomials A^* and B^* as in (12).
(iv) Choose the desired dynamics for the closed loop polynomial A^*_{cl} of degree $(2n+q-z-1)$.
(v) Solve the corresponding linear equation system involving the matrix associated to the linear transformation $\Phi^* \in \mathbb{R}^{2n+q-z-w \times 2n+q-z-w}$ to find the coefficients of L^* and P^*.
(vi) Compute L, P and A_{cl} as in (10).

Example 2. (Example 1 revisited). We recall the Example 1, but this time we will force the Stable Zero-pole cancellations in addition of the Internal Model Principle using the algorithm stated before. In this case, we cancel all stable factors, that is $z = 1$ and $w = 2$, and

$$ A^* = s(s^2+1) \quad \text{and} \quad B^* = 1. $$

$$ \begin{aligned} \deg\{L^*\} &= n - z - 1 = 0 \\ \deg\{P^*\} &= n + q - w - 1 = 2 \\ \deg\{A^*_{cl}\} &= 2n + q - z - w - 1 = 3 \end{aligned} $$

The corresponding dimensions

$$ \dim\{{}^*\mathbb{V}_{l^*}\} = 1, \quad \dim\{{}^*\mathbb{V}_{p^*}\} = 3, \quad \dim\{{}^*\mathbb{W}\} = 4 $$

Constructing the submatrices ${}^*\xi_{A^*} \in \mathbb{R}^{4\times 1}$ and ${}^*\xi_{B^*} \in \mathbb{R}^{4\times 3}$ yields

$$ {}^*\xi_{A^*} = \underbrace{\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}}_{\dim\{{}^*\mathbb{V}_{l^*}\}} \qquad {}^*\xi_{B^*} = \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\dim\{{}^*\mathbb{V}_{p^*}\}} $$

Computing the coefficients of L^* and P^* involves the following linear system

$$ \begin{bmatrix} l_0^* \\ p_2^* \\ p_1^* \\ p_0^* \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1}}_{{}^*\Phi^{-1}_{*V^*W}} \begin{bmatrix} 1 \\ 9 \\ 27 \\ 27 \end{bmatrix} $$

Solving the system of equations using the matrix of the linear transformation ${}^*\Phi_{*V^*W}$ computed before, the final controller is given by the following transfer function

$$ K(s) = \frac{\alpha P^*}{\Gamma \beta L^*} = \frac{9(s+2)^2(s^2+2.889s+3)}{s(s^2+1)(s+1)} \qquad (13) $$

In Figure 2 we show the step response from reference input to output using the controller developed in (8) and using the controller with zero-pole cancellations (13). An output disturbance $d_0(t) = sin(t)$ was injected at $t = 5[sec.]$.

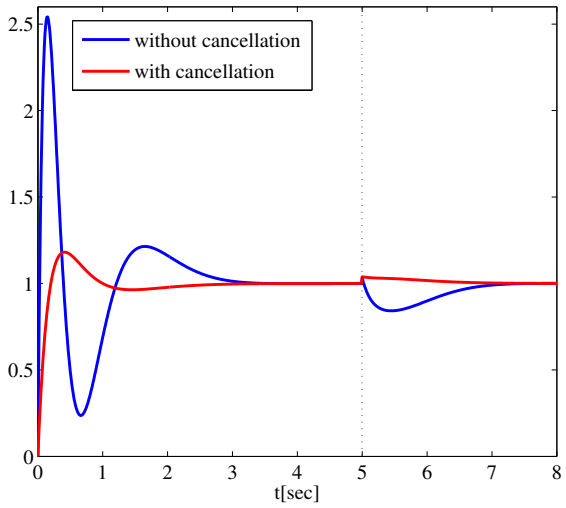


Fig. 2. Step response from reference input to output

4. CONCLUSIONS

In this work, we developed a simple and systematic algorithm to design SISO controllers based on a input-output mathematical model using linear algebra concepts. It considers the *Internal Model Principle* and allows to perform *Stable Zero-Pole cancellations* in the same linear transformation. The extrapolation of this algorithm to the discrete domain it's straightforward, which implies that it can be easily implemented in a microcontroller. In this way, it can be coupled to a model identification system turning the controller into an adaptive one, showing the versatility of the algorithm developed.

REFERENCES

Francis, B.A. and Wonham, W.M. (1975). The internal model principle for linear multivariable regulators. *Applied mathematics and optimization*, 2(2), 170–194.

Goodwin, G.C., Graebe, S.F., and Salgado, M.E. (2001). *Control system design*, volume 240. Prentice Hall New Jersey.

Hunt, K.J. (1993). *Polynomial methods in optimal control and filtering*. 49. Iet.

Kučera, V. (1991). *Analysis and design of discrete linear control systems*. Prentice-Hall, Inc.

Kučera, V. (1993). Diophantine equations in control—a survey. *Automatica*, 29(6), 1361–1375.

Li, H. (2011). A note on the inversion of sylvester matrices in control systems. *Mathematical Problems in Engineering*.

Seron, M.M., Braslavsky, J.H., and Goodwin, G.C. (1997). *Fundamental Limitations in Filtering and Control*. Springer.

Stencil computation for the approach to the numerical solution heat transfer problems on SoC FPGA

Luis F. Castaño * **Gustavo A. Osorio** **

* *Instituto Tecnológico Metropolitano, Medellín, Colombia (e-mail: luiscastano@itm.edu.co).*
Universidad Nacional de Colombia, Manizales, Colombia (e-mail: lfcastanol@unal.edu.co)
** *Universidad Nacional de Colombia, Manizales, Colombia (e-mail: gaosoriol@unal.edu.co).*

Abstract: A common kernel used in scientific computing is the stencil computation. FPGA based heterogeneous systems has been used to overcome stencil algorithm performance limitations due to the memory bandwidth on CPU and GPU based systems. Performance improvement is achieved through the combination of several data flow optimization techniques, taking advantage of the FPGA inherent parallelism. However 2D array architectures used in most cases, involves the need of considerable amount of FPGAs to simulate problems with sizes that can be treated by a CPU or GPU based system at a lower cost. In this document is presented a FPGA based system that use a one-dimensional array of processing elements for a stencil computation. The data-path is designed to reuse the processing elements and reduce the number of memory transfers using registers arrays to store data temporarily. The proposed architectures are implemented in a *ZedBoard Zynq Evaluation and Development Kit* using *Vivado® Design Suite* and *Xilinx SDK*. System performance is evaluated for the approach to numerical solution of heat transfer problems modelled with the heat equation for the one-dimensional case using stencil algorithm. An architecture for the Laplace equation for the two-dimensional case derived from the heat equation approach is proposed.

Keywords: FPGA, stencil computation, heat equation, Laplace equation

1. INTRODUCTION

A common kernel used in scientific computing is the stencil computation, particularly for linear algebra algorithms, partial differential equations (PDE) and image processing. It is quite efficient for the approach to the numerical solution of PDE using the explicit finite difference scheme (10). However, performance of algorithms based on stencil is limited by the remarkable difference between the maximum throughput and maximum bandwidth memory on multi-core CPU and GPU based systems (3). For this reason the study of optimization methods for stencil algorithms implementation has been of interest.

Cache blocking techniques have been developed to overcome the bandwidth limitations by exploiting the temporal and spatial locality. Studies on optimization of stencil computation over CPU or GPU can be found in (5; 6; 9; 11).

In (3) Sano *et al.* assert that with optimization methods for CPU or GPU based systems performance limitations could remain even though optimization methods due to memory bandwidth. FPGA-based accelerators are used as an alternative given that this devices has shown better performance with lower power consumption (1; 2). The performance improvement of the stencil computing scheme using FPGAs is study in (1; 3; 4; 7; 8).

FPGA based systems take advantage of the inherent parallelism for performance improvement through the combination of several data flow optimization techniques. For instance, grid array architectures as proposed in (3; 7) use streaming and pipelining to accelerate stencil computation. However, the use of such architectures involves the need of considerable amount of FPGAs to simulate problems for mesh sizes that can be treated by a CPU or GPU with suitable performance at a lower cost.

The use of FPGA based system has always represented multiple challenge from the number representations to the design flow complexity. The recent development of design tools has allowed overcoming many of these challenges. In (1) Schmitt *et al.* demonstrate the feasibility to deal with a grid of 4096×4096 on a single FPGA using a High-Level Synthesis (HLS) tool for system design.

With the aim to explore the optimization of stencil computation on an single FPGA in this work we present the algorithm implementation over a SoC FPGA, with a methodology that allows high level and traditional Register Transfer Level (RTL) methods for system description. A custom IP created using VHDL for the programmable logic (PL) section interacts with an ARM core that acts as

⋆ This work was supported by the AE&CC research group of the Instituto Tecnológico Metropolitano through the project P14208.

the host processor. A baseline architecture use a specific kernel as the processing element (PE) and a control unit for a sequential implementation of the stencil algorithm. For system parallelization is developed a data-path with a one-dimensional array of processing elements with feedback through a registers bank, with the aim to reduce the resources utilization and memory transfer operations.

System performance is evaluated for the approach to numerical solution of the one-dimensional heat equation with a maximum of 32 PE for a single FPGA. Performance analysis shows the improvement achieved in terms of elapsed time for the execution of stencil algorithm with two proposed parallel architectures in comparison with the execution time of the sequential algorithm written in C running on Linux over a Intel Xeon E5-2667 at 2.90GHz with 32 GB of RAM. The speed-up factor led to determine that implemented architectures offer different performance optimization due to the memory structure. Problem and system description are detailed in sections 2 and 3 respectively. Performance analysis and numerical results are presented in section 4. Finally conclusions and future work are drawn in section 5.

2. PROBLEM DESCRIPTION

2.1 Heat equation

Consider the PDE shown in (1).

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}, \qquad 0 < x < L \tag{1}$$

This expression represents a 1D parabolic PDE which is used to model the heat distribution over time in a bar with length L. Given an initial value and boundary conditions problem as shown in (2), equation solution shows the temperature variation in the space-time domain.

$$\begin{cases} u(x,0) = f(x) \\ u(0,t) = 0 \\ u(L,t) = 0 \end{cases} \tag{2}$$

An approach to the numerical solution of this equation is obtained using the explicit finite difference method. Defining J and N as the number of points for discretization in the space and time domain respectively, the approximate solution is obtained using (3).

$$u_j^{n+1} = u_j^n + \alpha(u_{j+1}^n - 2u_j^n + u_{j-1}^n) \tag{3}$$

From this expression a stencil kernel circuit is obtained as shown in Fig. 1.

This kernel is used to calculate each one of the mesh points as shown in Algorithm 1.

2.2 Laplace equation

Consider the elliptic PDE shown in (4) for $0 \leq x \leq l$ and $0 \leq y \leq h$.

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \ in \ (0,l) \times (0,h) \tag{4}$$

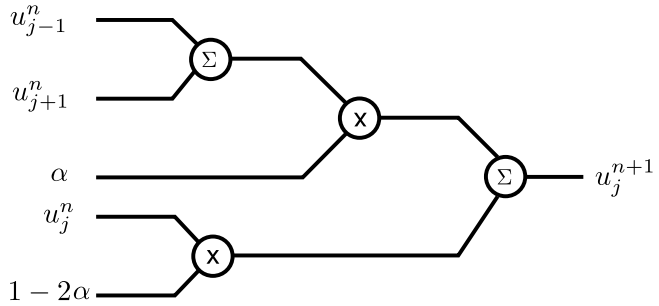


Fig. 1. Block diagram of the circuit for implementing the stencil operation for the heat equation with the explicit scheme.

Algorithm 1 Pseudocode for the stencil computation to obtain the approach to the numerical solution of heat equation with the explicit scheme

```
for n from 0 to N-1 do
    for j from 1 to J-2 do
        u_j^{n+1} ← (1 − 2α)u_j^n + α(u_{j+1}^n + u_{j−1}^n)
    end for
end for
```

Given the boundary conditions as shown in (5), equation solution shows the steady state temperature distribution in a two-dimensional rectangular plate.

$$\begin{cases} u(x,0) = u(x,h) = f(x) \\ u(0,y) = u(l,y) = g(x) \end{cases} \tag{5}$$

An approach to the numerical solution of this equation could be obtained using the explicit finite difference method as shown in (6).

$$u_{i,j}^{k+1} = \frac{u_{i-1,j}^k + u_{i+1,j}^k + \beta^2(u_{i,j-1}^k + u_{i,j+1}^k)}{2(1+\beta^2)} \tag{6}$$

where $\beta = \frac{\Delta x}{\Delta y}$

If there is a homogeneous distribution along x and y then the expression is as shown in (7).

$$u_{i,j}^{k+1} = \frac{u_{i-1,j}^k + u_{i+1,j}^k + u_{i,j-1}^k + u_{i,j+1}^k}{4} \tag{7}$$

The stencil kernel circuit is obtained from (7) as shown in Fig. 2.

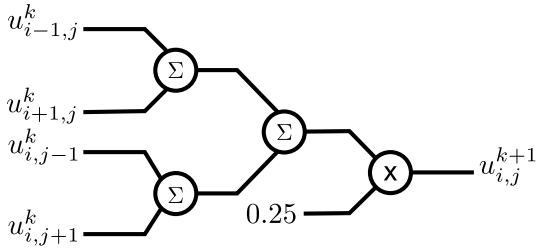


Fig. 2. Block diagram of the circuit for implementing the stencil operation for the Laplace equation with the explicit scheme.

This kernel is used to calculate each one of the mesh points as shown in Algorithm 2.

Algorithm 2 Pseudocode for the stencil computation to obtain the approach to the numerical solution of Laplace equation with the explicit scheme

```
for n from 0 to N − 1 do
    for j from 1 to h − 1 do
        for i from 1 to l − 1 do
            u^{n+1}_{i,j} ← 0.25(u^n_{i+1,j} + u^n_{i−1,j} + u^n_{i,j+1} + u^n_{i,j−1})
        end for
    end for
end for
```

3. SYSTEM DESCRIPTION

The system is implemented in a *ZedBoard Zynq Evaluation and Development Kit* using *Vivado® Design Suite*. The design takes advantage of the XC7Z020CLG484-1 *Xilinx* SoC FPGA architecture. The *ZYNQ-7* processing system (PS) for the *ARM® Cortex®-A9 MPCore™* interacts with a custom IP created for the programmable logic (PL) section. The ARM core acts as the host processor where the main application runs. The custom IP is used to calculate the approach to the numerical solution of the PDE using the stencil scheme. The system block diagram is shown in Figure 3.

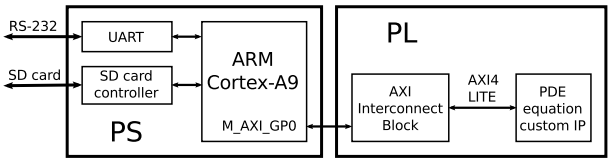


Fig. 3. Block diagram of the system implemented in *Vivado®*. Communication between PS and PL is made through AXI4-Lite interface with a fixed 32 data bits.

The source code for the PS is written in C over *Xilinx Software Development Kit (SDK)*. The application works as a host program to serve as user interface through serial terminal console. From this application, data such as mesh size, initial values, boundary conditions and control commands are send to the custom IP. Likewise, the status signals, performance counters and numerical results are reading from the PL. The last are written to a SD memory card.

The custom IP is fully described in VHDL. It is connected to the *ZYNQ-7* PS in a block design over *Vivado® IP integrator* tool. Communication between PS and PL sections is made through *AXI4-Lite* interface with fixed 32 data bits.

For number representation is used a customized 32-bit floating-point format with rounding to the nearest. The floating-point adders and multipliers used in the stencil kernel are described as combinational circuits, therefore there is no output latency in terms of the system clock cycles.

3.1 Architecture for the approach to the numerical solution of the one dimensional heat equation

A baseline architecture is designed for a sequential implementation of the stencil algorithm. The block diagram is shown in Figure 4.

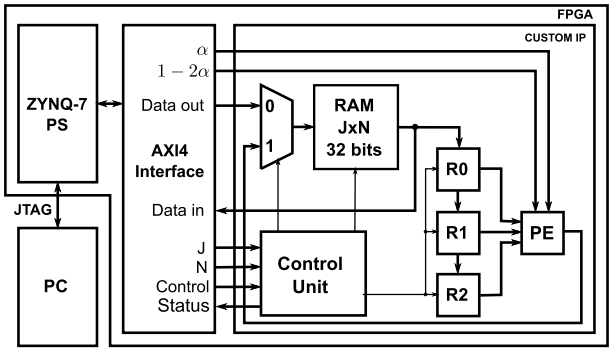


Fig. 4. Block diagram of the sequential baseline architecture. The stencil kernel is used as the processing element (PE) of the datapath.

Through serial console the mesh size ($J \times N$) is defined. The initial values and boundary conditions are setting and written to the RAM from host application. A control signal is sent to the control unit of the custom IP to start processing the memory data. The control unit is a finite state machine that coordinates the sequence of the stencil algorithm.

The registers R_0, R_1 and R_2 are connected in cascade to allow data streaming from memory. The term u_j^{n+1} calculated by the stencil kernel is saved to the RAM via multiplexer. A status signal from control unit tells the host application that has finished processing. Finally, memory data is read from the PL and stored in a file on SD card through PS. The flow chart for the stencil algorithm implemented in the heterogeneous system is shown in Fig. 5. Operations outside the dashed line are executed in PS and those found within the dashed line are executed in PL.

A performance counter is used to determine the number of clock cycles used to calculate all mesh points. This amount also can be calculated from the state machine sequence as shown in (8).

$$n_{CLK} = (N-1)(8J-8)+2 \tag{8}$$

To optimize the system performance for the implemented stencil algorithm and exploit the FPGA features, a variation of the baseline architecture described in section **??** is proposed. In order to increase the amount of space domain points that can be processed in one clock cycle more registers and PE are used. In Fig. 6 is shown the implementation for a $8 \times N$ mesh, with six PE and a register bank for eight data.

The control unit of this architecture has less states due to the J terms for the time step $n+1$ are obtained concurrently. To keep results available to calculate the values for the next time step without RAM access, the PE outputs are also stored concurrently in the register bank

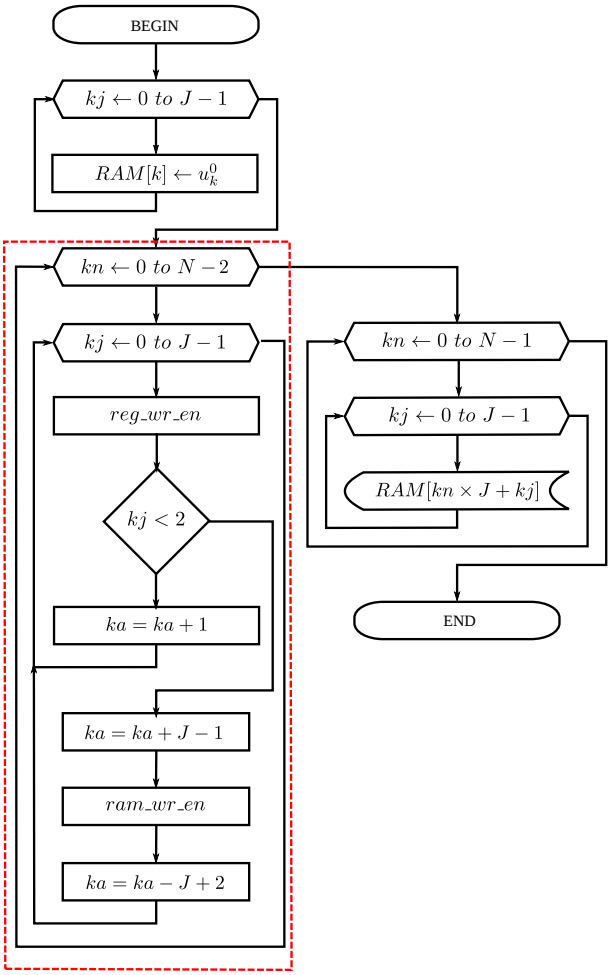


Fig. 5. Flow chart for the stencil algorithm implemented. Operations outside the dashed line are executed in PS and those found within the dashed line are executed in PL. The sequence for the execution of the stencil algorithm is coordinated by the control unit which is described in VHDL as a finite state machine. The counters k_j and k_n are used to control the loops. The counter k_a is used to address the RAM.

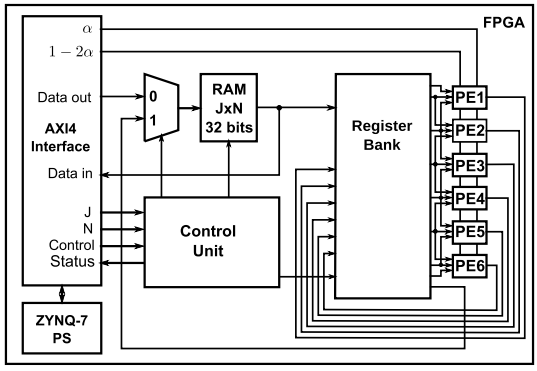


Fig. 6. Block diagram of the implementation for a $8 \times N$ mesh with concurrent processing.

trough internal multiplexers. With this configuration the algorithm inner loop is simplified.

The concurrent processing improves the algorithm performance, but data storage is still sequential given that the system has only one RAM. Therefore a memory structure that allows concurrent storage is proposed as shown in Fig. 7. In this architecture the inner loop is suppressed from the control unit sequence.

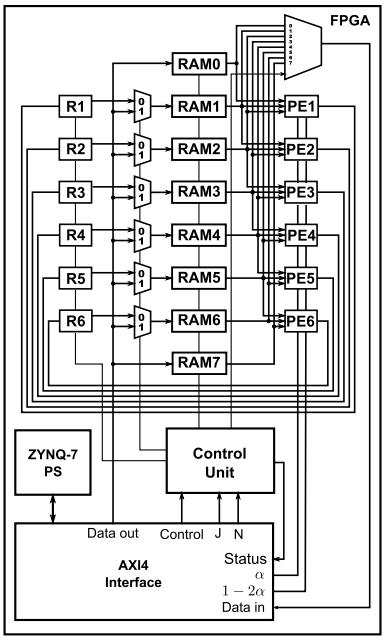


Fig. 7. Block diagram of the implementation for a $8 \times N$ mesh with concurrent processing and storage.

3.2 Architecture for the approach to the numerical solution of the two dimensional Laplace equation

From the architecture with concurrent processing and storage is developed a data-path with a two-dimensional register array to obtain the approach to the numerical solution of the 2D Laplace equation. The block diagram of the implemented circuit for a $8 \times N$ mesh is shown in Fig. 8.

4. EVALUATION

4.1 System performance

The system performance in terms of the time required for the algorithm execution is measured using an internal counter enabled from control unit. In Table 1 is shown the elapsed time in microseconds for 8, 16 and 32 points in the space domain and 512 iterations. The speed-up achieved with the parallel architectures (A_2 and A_3) is calculated in relation to the baseline architecture (A_1). The processing time for A_3 does not vary with respect to number of PE given that it just depends on the number of iterations.

Table 1. Algorithm execution time in microseconds obtained with the performance counter for J space points and N iterations.

$(J \times N)$	t_{A1} $[\mu s]$	t_{A2} $[\mu s]$	t_{A3} $[\mu s]$	Speedup t_{A1}/t_{A2}	Speedup t_{A1}/t_{A3}
8x512	286.18	138.52	30.68	1.215	2.066
16x512	613.22	261.64	30.68	2.344	19.987
32x512	1267.30	671.72	30.68	1.887	41.307

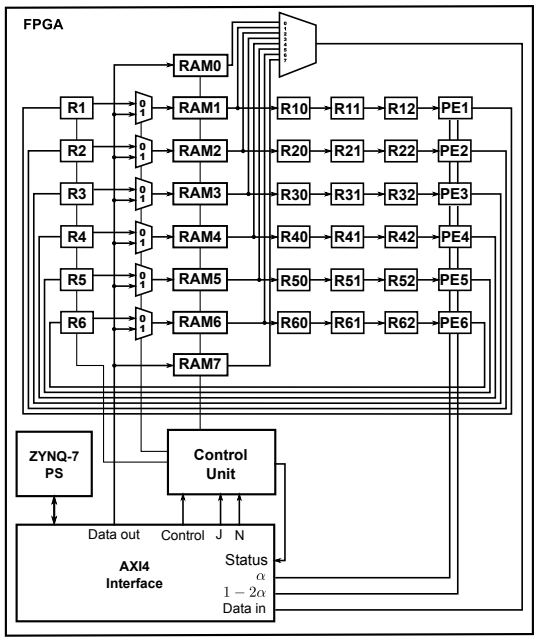


Fig. 8. Block diagram of the implementation for a $8 \times N$ mesh with concurrent processing and storage.

The speed-up achieved in comparison with the execution time of the algorithm written in C running on linux over a Intel Xeon E5-2667 at 2.90GHz with 32 GB of RAM is shown in Table 2. The values of t_{CPU} are the elapsed times used by the CPU processor to performs the nested loop.

Table 2. Speedup achieved in comparison with the execution time of the algorithm written in C running on linux over a Intel Xeon E5-2667 at 2.90GHz with 32 GB of RAM

$(J \times N)$	t_{CPU} $[\mu s]$	Speed-up t_{CPU}/t_{A1}	Speed-up t_{CPU}/t_{A2}	Speed-up t_{CPU}/t_{A3}
8x512	17.0	0.059	0.122	0.554
16x512	37.5	0.061	0.143	1.222
32x512	78.5	0.062	0.117	2.558

Performance can be improved for A2 if are stored only the results of the iteration N. Algorithm execution time in microseconds and speed-up obtained without storing all mesh for J space points and N iterations are shown in Table 3.

Table 3. Algorithm execution time in microseconds and speed-up obtained without storing all mesh points for J space points and N iterations.

$(J \times N)$	t_{A2} $[\mu s]$	Speed-up t_{A1}/t_{A2}	Speed-up t_{CPU}/t_{A2}
8x512	11.02	27.82	1.54
16x512	11.74	57.46	3.19
32x512	13.82	91.70	5.68

The stencil scheme used has four floating-point operations. Therefore with a 100 MHz clock the system has a peak performance of 400 MFLOPS in A_1 and 12 GFLOPS in A_2 and A_3. However, considering the number of mesh points, the stencil floating-point operations and the algorithm execution time, the system performance in GFLOPS corresponds to the values shown in Table 4.

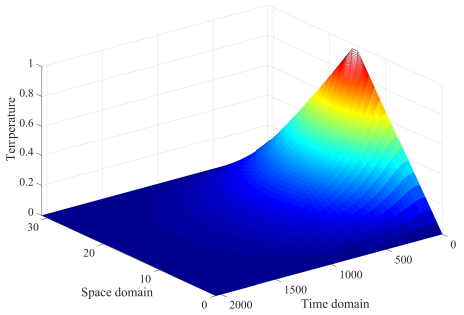
Table 4. System performance in GFLOPS considering the number of mesh points, the stencil operations and the execution time

	$GFLOPS_{A1}$	$GFLOPS_{A2}$	$GFLOPS_{A3}$
8x512	0.053	0.118	0.534
16x512	0.048	0.125	1.068
32x512	0.046	0.097	2.136

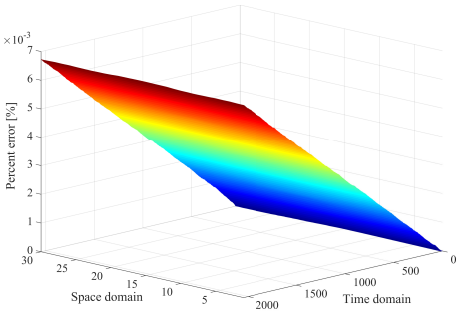
The FPGA resources utilization respect to the PE is summarized in Table 5 for A_1, A_2 and A_3. This reports corresponds to implementation using 65536x32 RAM for A_1 and A_2 and 512x32 RAM for A_3. The parallel implementations for 62 PE could not be performed because the circuit exceed the number of available FPGA slices.

4.2 Numerical results

For initial values a triangular function is generated and send from PS section. Results of the approach to the numerical solution of the 1D heat equation are stored in the SD card. Data values are printed in a text file using 15 decimal digits precision format. In Fig. 9a is shown a mesh for the approximated solution for 32 points and 2048 iterations plotted with MATLAB®. The percent error with respect to Matlab® results for the same mesh size, initial values and boundary conditions is shown in Fig. 9b. Although the error obtained until iteration 2048 does not exceed 0.007%, it is observed that is accumulative with the increase of time steps. This happen mainly because of rounding of the floating-point operations.



(a)



(b)

Fig. 9. (a) Approach to the numerical solution of the 1D heat equation obtained with the baseline architecture. (b) Percent error in comparison with Matlab® simulation. The plots are made using the Matlab® mesh function.

Table 5. FPGA resources utilization for the approximation to the numerical solution of heat equation.

	A1	A2			A3		
		6 PE	14 PE	30 PE	6 PE	14 PE	30 PE
Slice LUTs/ 53200	2461 (4.63%)	9381 (17.63%)	20131 (37.84%)	41953 (78.86%)	9212 (17.32%)	23244 (43.69%)	42600 (80.08%)
Slice Registers/ 106400	1692 (1.59%)	1404 (1.32%)	1792 (1.68%)	2581 (2.43%)	1641 (1.54%)	2068 (1.94%)	3136 (2.95%)
F7 Muxes/ 26600	182 (0.68%)	142 (0.53%)	246 (0.92%)	374 (1.41%)	160 (0.6%)	2240 (8.42%)	256 (0.96%)
F8 Muxes/ 13300	27 (0.2%)	59 (0.44%)	59 (0.44%)	187 (1.41%)	0 (0%)	1056 (7.94%)	128 (0.96%)
Block RAM Tile/ 140	58 (41.43%)	58 (41.43%)	58 (41.43%)	58 (41.43%)	4 (2.86%)	0 (0%)	16 (11.43%)
DSPs/ 220	4 (1.82%)	24 (10.91%)	56 (25.45%)	120 (54.55%)	24 (10.91%)	56 (25.45%)	120 (54.55%)

5. CONCLUSIONS AND FUTURE WORK

In this paper is presented a FPGA based heterogeneous system for stencil computation. The system is designed for the approach to the numerical solution of parabolic PDE for a one dimensional heat transfer problem with initial value and boundary conditions. The implementation is made using the SoC architecture of a XC7Z020CLG484-1 *Xilinx* FPGA of the ZedBoard.

Three different architectures based in the explicit finite difference method are described. Performance analysis shows the improvement achieved in terms of execution time for the stencil algorithm with two proposed parallel architectures. The speedup factor led to determine that implemented architectures offer different performance optimization due to the memory structure. However in both cases the use of the registers array allows to take advantage of spatial and temporal locality reducing the need of memory transfer operations.

For future work more deep performance analysis in terms of accuracy, precision, data transfer, scalability and power consumption could be performed. This evaluation should allow performance comparison with CPU and GPU based systems. Otherwise the study of variation for the implemented architectures to address bigger heat transfer problems in 2D using parabolic and elliptic PDEs could be developed.

ACKNOWLEDGEMENTS

This work was supported by the *AE&CC* research group of the Instituto Tecnológico Metropolitano through the project P14208. Luis Castaño acknowledges the financial support by the scholarship *Estudiantes Sobresalientes de Posgrado* Universidad Nacional de Colombia.

REFERENCES

[1] Schmitt, C., Schmid, M., Hannig, F., Teich, J., Kuckuk, S. and Harald Köstler, *A Challenge of Portable and High-Speed FPGA Accelerator*, in Proceedings of the Second International Workshop on High-Performance Stencil Computations, HiStencils 2015, pp. , 2015

[2] T. Usui, R. Kobayashi and K. Kise, *A Challenge of Portable and High-Speed FPGA Accelerator*, in Proceedings of the 11th International Symposium on Applied Reconfigurable Computing, ARC 2015, Springer International Publishing pp. 383-392, 2015

[3] K. Sano, Y. Hatsuda and S. Yamamoto, *Multi-FPGA Accelerator for Scalable Stencil Computation with Constant Memory-Bandwidth*, IEEE Transactions on Parallel and Distributed Systems, vol. PP, 2014

[4] R. Kobayashi and K. Kise, *Scalable stencil-computation accelerator by employing multiple FPGA*, IPSJ Transactions on Advanced Computing Systems, pp. 1-13, 2013

[5] V. Bandishti, I. Pananilath and U. Bondhugula, *Tiling Stencil Computations to Maximize Parallelism*, en Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE, pp. 1-11, 2012

[6] J. M. Cecilia, J. L. Abellán, J. Fernández, M. E. Acacio, J. M. García, M. Ujaldón, *Stencil computations on heterogeneous platforms for the Jacobi method: GPUs versus Cell BE*, The Journal of Supercomputing, Springer Science+Business Media, vol. 62, No. 2, pp. 787-803, 2012

[7] R. Kobayashi, S. Takamaeda-Yamazaki and K. Kise, *Towards a Low-Power Accelerator of Many FPGAs for Stencil Computations*, en Proceedings of the Third International Conference on Networking and Computing, IEEE, pp. 343-349, 2012

[8] K. Sano, Y. Hatsuda and S. Yamamoto, *Scalable Streaming-Array of Simple Soft-Processors for Stencil Computations with onstant Memory-Bandwidth*, en Proceeding of the 19th Annual International Symposium on Field-Programmable Custom Computing Machines, IEEE, pp. 234-241, 2011

[9] R. Strzodka, M. Shaheen, D. Pajak and H. Seidel, *Cache Accurate Time Skewing in Iterative Stencil Computations*, en Proceedings of the International Conference on Parallel Processing, IEEE, pp. 571-581, 2011

[10] A. R. Brodtkorb, C. Dyken, T. R. Hagen, J. M. Hjelmervik and O. O. Storaasli, *State-of-the-art in heterogeneous computing*, Scientific Programming, IOS Press Amsterdam, vol 18, pp. 1-33, 2010

[11] K. Datta, S. Kamil, S. Williams, L. Oliker, J. Shalf and K. Yelick, *Optimization and Performance Modeling of Stencil Computations on Modern Microprocessors*, SIAM Review, vol. 51, no. 1, pp. 129-159, 2009

Synthesis of Four-bar Mechanisms for Trajectory Control Using the Modified Brainstorm Optimization Algorithm and Linkage Normalization

Pedro Bautista-Camino * **Eduardo Vega-Alvarado** *
Edgar Alfredo Portilla-Flores * **Paola Andrea Niño-Suárez** **
Maria Bárbara Calva-Yañez *

* *Instituto Politécnico Nacional, México - CIDETEC - Laboratorio de Mecatrónica (e-mail: peterbc1@gmail.com)*
** *Instituto Politécnico Nacional, México - ESIME Azcapotzalco - SEPI*

Abstract: The Modified Brainstorm Optimization (MBSO) algorithm, explored in recent literature, is an efficient metaheuristic optimization technique inspired on the process of solving a problem by humans when they get together to interchange ideas and opinions, complementing each other in order to reach a general solution. This paper presents a novel application of MBSO for the dimensional synthesis of four-bar planar mechanisms for trajectory control, with three different case studies. The four-bar mechanisms are used in a wide variety of industrial applications because of their simplicity, but the associated design process is a very complex task, so they are a good example of hard optimization problems. MBSO was modified to handle design constraints by implementing the feasibility rules of Deb. Additionally, a normalization function was incorporated considering the aesthetics of the mechanisms, to avoid unfeasible combinations because of the lenght of their components. Simulation results show a high-precision control of the proposed trajectories for the designed mechanisms, thus suggesting that MBSO can be used successfully as a tool for solving design engineering cases.

Keywords: MBSO, optimization, dimensional synthesis, bar normalization, four-bar mechanism.

1. INTRODUCTION

In recent years, the use of metaheuristics has been a very effective tool for solving real world problems. In engineering, *synthesis* is the process for designing mechanical systems; the dimensional synthesis is the length calculus of the components of a mechanism for producing a specific movement or behavior. The application of metaheuristics in this kind of numerical problems has been treated previously in related literature as in Santiago-Valentin (2016); Jiménez et al. (2014); Portilla-Flores et al. (2014), among others.

The Brainstorm Optimization Algorithm is a relatively new algorithm proposed by Shi (2011), that has been used successfully in engineering optimization problems as in Duan et al. (2013), Jordehi (2015) and Fernández-Jiménez (2014), among others. Subsequently a modification of the original algorithm was proposed by (Zhan et al., 2012) called Modified Brainstorm Optimization Algorithm (MBSO) which computationally improves the performance of the original algorithm; this algorithm is used in this work. In this paper, a new proposal based on MBSO with the addition of the rules of Deb (2000) for handling constrains is presented, as a tool for solving real world optimization problems.

The four-bar mechanism is used in many industrial applications because of its simplicity; however, its synthesis for trajectory tracking is a complex task that can not be solved efficiently by using deterministic methods. Three case studies are presented here, in order to test the efficiency of the MBSO algorithm and the quality of its results; this cases are variations of a four-bar mechanism that are designed for controlling specific trajectories. In the modeling of the problems was included a normalization function for the links (Capistran-Gumersindo, 2015), in order to improve the aesthetic of the mechanisms; this means that the ratio between the bars should be balanced. A mechanism is more aesthetic as it is better suited for its manufacturing; it implies an enhance on stress distribution, the use of cheaper and/or less specialized materials, etc.

This paper is distributed as follows: In Section 2 the methodology is introduced, including the analysis of the four-bar mechanism, the optimization strategy and the computational implementation. In Section 3 the results of the case studies are analyzed, while the final discussion is

⋆ The authors would like to thank Instituto Politécnico Nacional (IPN) of México, for its support via Secretaría de Investigación y Posgrado (SIP) with the project SIP-20161615.

presented in Section 4. Finally, there is an appendix with the data tables resulting from the simulation carried out.

2. METHODOLOGY

This research addresses the dimensional synthesis of a four-bar mechanism with three different case studies for trajectory control; so, as a first point the general model of this mechanism is developed.

2.1 Kinematics of the mechanism

The four-bar mechanism is one of the most used devices in machinery design due to its simplicity for controlling movements with one freedom degree. These mechanisms have been well studied in the corresponding literature; a detailed description is in Sanchez-Marquez et al. (2014). Fig. 1 shows a planar mechanism whose elements are: a reference bar (r_1), a crank bar (r_2), a couple bar (r_3) and a rocker bar (r_4).The equation of the closed loop is in (1):

$$\boldsymbol{r}_1 + \boldsymbol{r}_4 = \boldsymbol{r}_2 + \boldsymbol{r}_3 \tag{1}$$

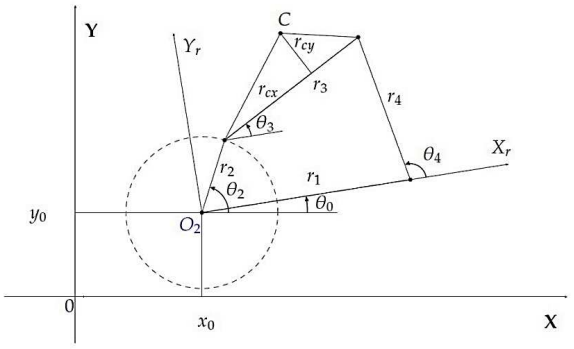


Fig. 1. Four-bar mechanism

The spinning of r_2 is determined by the angle θ_2, and this causes the oscillation of r_4, which in turn produces the movement of r_3 in a closed trajectory. Each point in this trajectory is determined by the position of the coupler C, whose spacial coordinate is established in the system OX_rY_r as indicated in (2),

$$\begin{aligned} C_{xr} &= r_2 cos\theta_2 + r_{cx} cos\theta_3 - r_{cy} sin\theta_3 \\ C_{yr} &= r_2 sin\theta_2 + r_{cx} sin\theta_3 + r_{cy} cos\theta_3 \end{aligned} \tag{2}$$

In the global coordinate system, this point can be expressed as in (3)

$$\begin{bmatrix} C_x \\ C_y \end{bmatrix} = \begin{bmatrix} cos\theta_0 & -sin\theta_0 \\ sin\theta_0 & cos\theta_0 \end{bmatrix} \begin{bmatrix} C_{xr} \\ C_{yr} \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{3}$$

It is important to note that equations (2), (3), and the expressions of the mechanism kinetics are enough to calculate any position of point C along the trajectory of the mechanism.

2.2 Design constraints

Since the case studies presented are taken from real-world problems, in order to solve the synthesis of the required four-bar mechanisms it is necessary to consider the following constraints:

- Law of Grashof: This law establishes that for a planar four-bar linkage, *the sum of the shortest bar and the largest bar cannot be larger than the sum of the remaining bars, if a continual relative rotation between two elements is desired* (Norton, 1995). It is expressed in (4),

$$\boldsymbol{r}_1 + \boldsymbol{r}_2 \leq \boldsymbol{r}_3 + \boldsymbol{r}_4 \tag{4}$$

So, the relationships in (5) must be satisfied,

$$\boldsymbol{r}_2 < \boldsymbol{r}_3, \quad \boldsymbol{r}_3 < \boldsymbol{r}_4, \quad \boldsymbol{r}_4 < \boldsymbol{r}_1 \tag{5}$$

- Sequence of angles: The coupler of the mechanism must pass in a logical order along the points that delimited the trajectory (precision points). It can be achieved if the angular positions corresponding to each of the precision points are ordered in magnitude as shown in (6), where n is the number of precision points in the desired path.

$$\sum_{i=1}^{n-1} \theta_2^i - \theta_2^{i+1} \leq 0 \tag{6}$$

2.3 Objective function

Since an objective function is a mathematical relationship for measuring the performance of a system, for these case studies the mean square error between the calculated points and the desired points is used, so the ideal value is zero. This objective function is expressed in (7), where n is the number of points precision, C_{xd}^i and C_{yd}^i represent the components x and y of the desired points, and C_x^i and C_y^i correspond to the similar components for the calculated points of the system, that are a function of θ_2.

$$f(\theta_2^i) = \sum_{i=1}^{n} [(C_{xd}^i - C_x^i)^2 + (C_{yd}^i - C_y^i)^2] \tag{7}$$

Additionally, a second objective function is considered in order to avoid a disproportionated mechanism, in spite of this unbalanced mechanism could deliver a zero error, taking into account that the expression in (7) does not consider a balance on the final length of the bars. The extra function for normalizing the length of the bars is described in (Capistran-Gumersindo, 2015), and corresponds to the equation (8),

$$f(\boldsymbol{x}) = \frac{\sqrt{\sum_{i=1}^{6} \sum_{j=1}^{6} (x_i - x_j)^2}}{\sqrt{\sum_{i=1}^{6} x_i^2}} \tag{8}$$

where $\boldsymbol{x}$ is a vector formed by the variables that represents the magnitude of the bars, as expressed in (9),

$$\boldsymbol{x} = [r_1, r_2, r_3, r_4, r_{cx}, r_{cy}]^T = [x_1, x_2, x_3, x_4, x_5, x_6,]^T \tag{9}$$

The merge of functions (7) and (8) forms the weighted sum indicated in (10), where w_1 and w_2 are the coefficients for controlling the aesthetic of the mechanism. With a value

of w_2 bigger than w_1 the bars tend to resemble each other, but the precision of the generated trajectory decreases; by the opposite, a bigger value on w_1 will produce more precision with less aesthetic.

$$f_r = w_1 f(\theta_2^i) + w_2 f(\boldsymbol{x}) \tag{10}$$

2.4 Case studies

Three case studies for the synthesis of four-bar mechanisms are developed, with their respective numerical optimization problems:

(1) Control of a linear trajectory (*CS1*): The mechanism must follow a trajectory which describes a vertical line; the set of precision points describing this line is listed in (11), while the vector of design variables is given by (12).

$$\omega_1 = \begin{bmatrix} (20,20),\ (20,25),\ (20,30) \\ (20,35),\ (20,40),\ (20,45) \end{bmatrix} \tag{11}$$

$$\begin{aligned} \boldsymbol{p} &= [r_1, ..., r_4, r_{cx}, r_{cy}, \theta_0, x_0, y_0, \theta_2^1, ..., \theta_2^6]^T \\ &= [p_1, ..., p_{15}]^T \end{aligned} \tag{12}$$

where the variables r_1, ... , r_4 are the length of the bars, r_{cx} and r_{cy} are the components of the coupler, θ_0 is the angle of inclination of the mechanism, x_0 and y_0 are the components of the relative coordinate system, and $\theta_2^1 ... \theta_2^6$ represent the position angular for each precision point in θ_2. The upper and lower range for each variable are listed in (13):

$$\begin{aligned} r_1, r_2, r_3, r_4 &\in [0, 60] \\ r_{cx}, r_{cy}, x_0, y_0 &\in [-60, 60] \\ \theta_0, \theta_2^1, \theta_2^2, \theta_2^3, , \theta_2^4, \theta_2^5, \theta_2^6 &\in [0, 2\pi] \end{aligned} \tag{13}$$

So, the problem is defined by the expression in (14),

$$\begin{aligned} \min\ f(\boldsymbol{p}) &= w_1 f(\theta_2^i) + w_2 f(\boldsymbol{x}) \\ \boldsymbol{p} &\in \Re^{15} \end{aligned} \tag{14}$$

Subject to the conditions in (15):

$$\begin{aligned} g_1(\boldsymbol{p}) &= p_1 + p_2 - p_3 - p_4 \leq 0 \\ g_2(\boldsymbol{p}) &= p_2 - p_3 \leq 0 \\ g_3(\boldsymbol{p}) &= p_3 - p_4 \leq 0 \\ g_4(\boldsymbol{p}) &= p_4 - p_1 \leq 0 \\ g_5(\boldsymbol{p}) &= p_{10} - p_{11} \leq 0 \\ g_6(\boldsymbol{p}) &= p_{11} - p_{12} \leq 0 \\ g_7(\boldsymbol{p}) &= p_{12} - p_{13} \leq 0 \\ g_8(\boldsymbol{p}) &= p_{13} - p_{14} \leq 0 \\ g_9(\boldsymbol{p}) &= p_{14} - p_{15} \leq 0 \end{aligned} \tag{15}$$

(2) Trajectory tracking with unaligned points (*CS2*): For this case, the mechanism must follow a trajectory that passes along the points indicated by the set in (16), with a previously established sequence of angles for each precision point, as shown in (17),

$$\omega_2 = \begin{bmatrix} (3,3), (2.759, 3.363), & (2.372, 3.663) \\ (1.890, 3.862), & (1.355, 3.943) \end{bmatrix} \tag{16}$$

$$\theta_2^i = \{\frac{\pi}{6}, \frac{\pi}{4}, \frac{\pi}{3}, \frac{10\pi}{24}, \frac{\pi}{2}\} \tag{17}$$

Additionally θ_0, x_0 and $y_0 = 0$; so, the solution is reduced to the design vector in (18):

$$\begin{aligned} \boldsymbol{p} &= [r_1, r_2, r_3, r_4, r_{cx}, r_{cy}]^T \\ &= [p_1, p_2, p_3, p_4, p_5, p_6]^T \end{aligned} \tag{18}$$

where the upper and lower limits for each variable are given by (19):

$$\begin{aligned} r_1, r_2, r_3, r_4 &\in [0, 50] \\ r_{cx}, r_{cy} &\in [-50, 50] \end{aligned} \tag{19}$$

The optimization problem is defined in (20):

$$\begin{aligned} \min\ f(\boldsymbol{p}) &= w_1 f(\theta_2^i) + w_2 f(\boldsymbol{x}) \\ \boldsymbol{p} &\in \Re^6 \end{aligned} \tag{20}$$

Subject to the constraints in (21):

$$\begin{aligned} g_1(\boldsymbol{p}) &= p_1 + p_2 - p_3 - p_4 \leq 0 \\ g_2(\boldsymbol{p}) &= p_2 - p_3 \leq 0 \\ g_3(\boldsymbol{p}) &= p_3 - p_4 \leq 0 \\ g_4(\boldsymbol{p}) &= p_4 - p_1 \leq 0 \end{aligned} \tag{21}$$

(3) Motion generation delimited by a set with ten pairs of precision points (*CS3*): For this case, the mechanism must generate a path such that the coupler passes between every pair of the precision points included in the Table 1:

Table 1. Set of points for *CS3*

P1	(1.768 , 2.3311)	(1.9592 , 2.44973)
P2	(1.947 , 2.6271)	(2.168 , 2.675)
P3	(1.595 , 2.7951)	(1.821 , 2.804)
P4	(1.019 , 2.7241)	(1.244 , 2.720)
P5	(0.479 , 2.4281)	(0.705 , 2.437)
P6	(0.126 , 2.0521)	(0.346 , 2.104)
P7	(-0.001 , 1.720)	(0.195 , 1.833)
P8	(0.103 , 1.514)	(0.356 , 1.680)
P9	(0.442 , 1.549)	(0.558 , 1.742)
P10	(1.055 , 1.905)	(1.186 , 2.088)

The vector of design variables is defined by (22):

$$\begin{aligned} \boldsymbol{p} &= [r_1, .., r_4, r_{cx}, r_{cy}, \theta_0, x_0, y_0, \theta_2^1, ..., \theta_2^{10}]^T \\ &= [p_1, ..., p_{19}]^T \end{aligned} \tag{22}$$

The limits for each variable are expressed by (23),

$$\begin{aligned} r_1, r_2, r_3, r_4 &\in [0, 60] \\ r_{cx}, r_{cy}, x_0, y_0 &\in [-60, 60] \\ \theta_0, \theta_2^1, ..., \theta_2^{10} &\in [0, 2\pi] \end{aligned} \tag{23}$$

For this case the trajectory is defined for a set of point pairs precision; so, the modification of the objective function in (24) is used:

$$\begin{aligned} \min\ f(\boldsymbol{p}) &= w_1(Error_1 + Error_2) + w_2 f(\boldsymbol{x}) \\ \boldsymbol{p} &\in \Re^{19} \end{aligned} \tag{24}$$

and the partial errors are defined in (25):

$$Error_1 = \sum_{i=1}^{n}[(C^i_{1xd} - C^i_x)^2 + (C^i_{1yd} - C^i_y)^2]$$

$$Error_2 = \sum_{i=1}^{n}[(C^i_{2xd} - C^i_x)^2 + (C^i_{2yd} - C^i_y)^2] \quad (25)$$

This problem is subject to the constraints in (26):

$$\begin{aligned}
g_1(\boldsymbol{p}) &= p_1 + p_2 - p_3 - p_4 \leq 0 \\
g_2(\boldsymbol{p}) &= p_2 - p_3 \leq 0 \\
g_3(\boldsymbol{p}) &= p_3 - p_4 \leq 0 \\
g_4(\boldsymbol{p}) &= p_4 - p_1 \leq 0 \\
g_5(\boldsymbol{p}) &= p_{10} - p_{11} \leq 0 \\
g_6(\boldsymbol{p}) &= p_{11} - p_{12} \leq 0 \\
g_7(\boldsymbol{p}) &= p_{12} - p_{13} \leq 0 \\
g_8(\boldsymbol{p}) &= p_{13} - p_{14} \leq 0 \\
g_9(\boldsymbol{p}) &= p_{14} - p_{15} \leq 0 \\
g_{10}(\boldsymbol{p}) &= p_{15} - p_{16} \leq 0 \\
g_{11}(\boldsymbol{p}) &= p_{16} - p_{17} \leq 0 \\
g_{12}(\boldsymbol{p}) &= p_{17} - p_{18} \leq 0 \\
g_{13}(\boldsymbol{p}) &= p_{18} - p_{19} \leq 0
\end{aligned} \quad (26)$$

2.5 Optimization Strategy

The Brainstorm Optimization algorithm was proposed by Shi (2011), based in four ideas developed by Osborn (1957). Zhan et al. (2012) presented an enhanced version, MBSO, which is used in this work. It is inspired on the process of solving a problem by humans when they get together to interchange ideas and opinions, complementing each other in order to reach a general solution. In the algorithm the ideas are represented with vectors of real numbers corresponding to the design variables. The ideas are grouped accordingly to their similarity, in a process called clustering; then, it is determined whether one of these central ideas will be replaced for a completely new one. New ideas can be formed from one or two clusters, and a process of competition is produced (Zhan et al., 2013); the complete MBSO is described in the Algorithm 1.

MBSO uses the parameters listed in the Table 2. The process of creating a new idea is governed by the relationship in (27). This equation indicates that the new idea Y_1 in the *nth* dimension (d) is a random number between the limits L_d and H_d if and only if a random number is lower than a preset probability P_r. Otherwise, the new idea will be generated by adding noise; this noise is produced from the difference between two elements of the same dimension from another two ideas, picked randomly and multiplied by a random number between 0 and 1. The function *rand* generates random numbers with a uniform distribution.

$$Y^d_{1new} = \begin{cases} rand(L_d, H_d) & if(rand(0,1) < P_r) \\ p^d_i + rand(0,1)(p^d_a - p^d_b) & otherwise \end{cases} \quad (27)$$

Experiment design A set of 300 simulations was executed for each case study, with 30 executions for each combination of weights; these combinations start with $w_1 = 1$ and $w_2 = 0$, and by subtracting 0.1 and adding 0.1 respectively they finish in $w_1 = 0.1$ and $w_2 = 0.9$. The number of evaluations of the objective function was taken as a stop condition, with the values of 350,000, 150,000 and 350,000 for *CS1*, *CS2*, and *CS3*, respectively; the values of the complementary parameters are listed in the Table 3.

The constraints are handled by implementing in MSBO the feasibility rules of Deb (2000), which dictate that:

Algorithm 1 Modified Brain Storm Optimization

```
Randomly generate N ideas and evaluate them
while (CurrentEval ⩽ EvalMax) do
    Cluster the N ideas into M clusters, keep the best idea in each cluster as the cluster center
    if (rand(0,1) < P_R) then
        Randomly select a cluster and replace its center with a randomly generated idea
    end if
    for (i = 1 to N) do
        if (rand(0,1) < P_G) then ▷ Create an idea based on one cluster
            Randomly select a cluster j with probability P_j
            if (rand(0,1) < P_C1) then
                Add random values to the selected cluster center, to generate a new idea Y_i
            else
                Add random values to a random idea from the selected cluster, to generate a new idea Y_i
            end if
        else ▷ Create an idea based on two clusters
            Randomly select two clusters, j_1 and j_2
            if (rand(0,1) < P_C2) then
                Combine two cluster center and add a random value to generate a new idea Y_i;
            else
                Combine two random ideas from the two clusters and add a value to generate a new idea Y_i;
            end if
        end if
        Evaluate the idea Y_i and replace X_i if Y_i is better than X_i;
    end for
end while
```

Table 2. Parameters of the MBSO algorithm

Number of ideas	N
Number of clusters	M
Probability of replacing a full idea	P_R
Probability of replacing one component of the idea	P_r
Probability of creating a new idea based in one cluster	P_G
First probability of center cluster idea	P_{C1}
Second probability of center cluster idea	P_{C2}

Table 3. Tuned parameters for all cases

Parameter	***CS1***	***CS2***	***CS3***
N	45	100	100
M	10	20	30
P_R	0.4	0.4	0.4
P_G	0.8	0.8	0.8
P_{C1}	0.8	0.7	0.8
P_{C2}	0.8	0.7	0.7
P_r	0.01	0.01	0.01

(1) Between two feasible individuals the one with the best value in the objective function is selected.
(2) Between a feasible individual and another unfeasible, the feasible one will be chosen.
(3) Between two infeasible individuals, the individual with the lower sum of constraint violations will be accepted.

The feasibility is measured by the sum of constraint violation determined by (28), where p is the number of constraints; only inequality constraints were used in the three case studies:

$$SVR = \sum_{i=1}^{p} max(0, g_i(\boldsymbol{p})) \tag{28}$$

3. RESULTS

In all the three case studies it was found that, without the normalization function, the mechanisms tend to be quite disproportionate, specially in *CS2* and *CS3*; e.g., in the second case the value of r_2 for a combination of weights $w_1 = 1$ and $w_2 = 0$ is about ten times smaller than the rest of the bars. These results can be consulted in the appendix of this paper. On the other hand, if it is given more weight to the normalization function, the obtained mechanisms tend to present unexpected behaviors since bars are more like each other and they appear to be more a double rocker device. The best aesthetic and functional mechanisms were generated when the weight assigned to the normalization coefficient was in the range $0 < w_2 < 0.5$.

Table 4. Statistical analysis - *CS1*

Weight 0.1+0.9					
Average	0.5107	σ^2	0.3311	Best	0.0706
Median	0.1998	σ	0.5754	Worse	2.0423
Weight 0.5+0.5					
Average	1.2645	σ^2	0.4008	Best	0.15381
Median	1.2354	σ	0.633	Worse	2.43293
Weight 0.9+0.1					
Average	0.3881	σ^2	0.0862	Best	0.13537
Median	0.2995	σ	0.2936	Worse	1.30035
Weight 1+0					
Average	0.3823	σ^2	0.2603	Best	0.0019
Median	0.0454	σ	0.5102	Worse	1.6361

MBSO generated good results from the point of view of engineering as they converged to feasible solutions in the three problems, and the values of the objective function shown quality results corresponding to acceptable trajectories. Tables 4, 5, and 6 show the statistic analysis for *CS1*, *CS2*, and *CS3*, respectively, taking into account the extreme and the medium cases regarding the weights. In the first two case studies, the value $\sigma^2 < 1$ indicates that the results tend to be accurate.

For *CS1*, the best result was obtained with the combination of weights of 0.9 to the mean square error function and 0.1 to the normalization function; a simulation of the corresponding mechanism is shown in Fig. 2.

For the second problem the mechanisms generated without the normalization function are very disproportionate; in this case the best combination of weights was $w_1 = 0.7$ and $w_2 = 0.3$, whose corresponding mechanisms is presented in

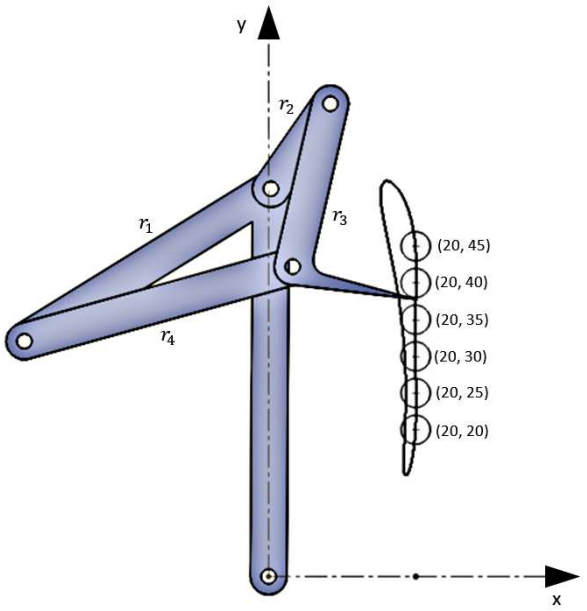


Fig. 2. Mechanism for the first case study

Fig. 3. It is noteworthy that in this case the trajectory marked by the mechanism has a error that could be questionable for the purpose of an optimal trajectory tracking, but otherwise complies with the ratio of the bars; without the normalization function the mechanism would be infeasible in the real world.

As in the second case, for *CS3* the mechanisms produced without a normalization tend to be physically irreproducible; the algorithm generated its best result with $w_1 = 0.7$ and $w_2 = 0.3$. This mechanism is shown in Fig. 4.

4. CONCLUSION

As noted in the results section, the use of an additional function for normalization allow to control the size of the bars by simply varying the proportion in a weighted sum. Depending on how much accuracy is required in the trajectory less weight should be assigned to the normalization function; on the other hand, if the weight assigned to the normalization function is greater than 0.5 the obtained mechanisms have bars that tend to be very similar among themselves, generating unexpected behaviour. This suggests that the adequate range for aesthetic mechanisms should be $w_2 \in [0, 0.5]$. This range allows to have a balance

Table 5. Statistical analysis - *CS2*

Weight 0.1+0.9					
Average	0.0924	σ^2	2.03E-17	Best	0.0924
Median	0.0924	σ	4.51E-09	Worse	0.0924
Weight 0.5+0.5					
Average	0.1247	σ^2	4.70E-13	Best	0.1247
Median	0.1247	σ	6.86E-07	Worse	0.1247
Weight 0.9+0.1					
Average	0.0379	σ^2	9.37E-10	Best	0.0379
Median	0.0379	σ	3.06E-05	Worse	0.03803
Weight 1+0					
Average	0.0031	σ^2	2.56E-08	Best	0.0028
Mediana	0.0031	σ	0.00016	Worse	0.0035

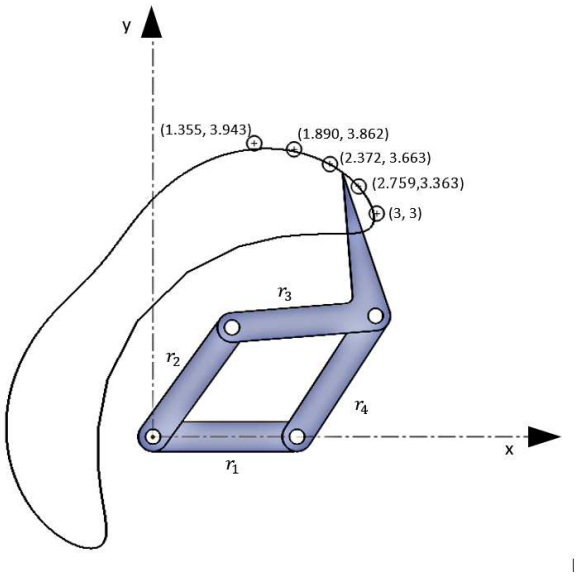


Fig. 3. Mechanism for the second case study

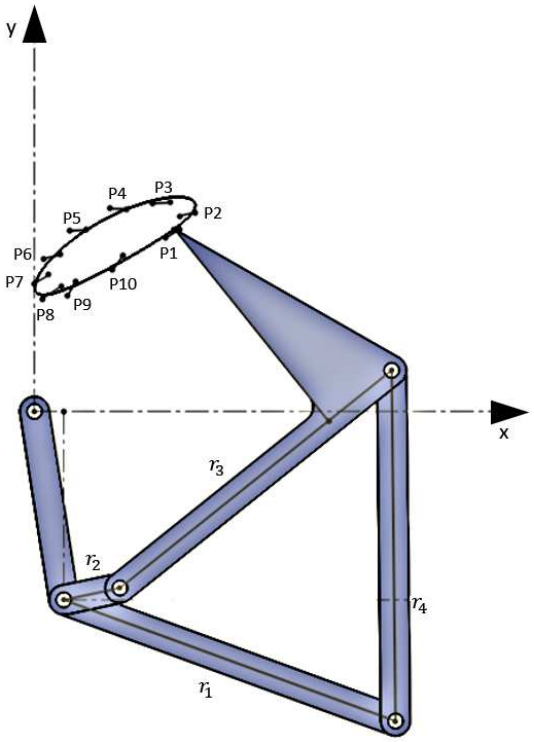


Fig. 4. Mechanism for the third case study

between the aforementioned aesthetics and the precise control of the trajectories, in order to produce solutions that are acceptable from the point of view of engineering.

Analyzing the MSBO from the point of view of computing, the algorithm meets the requirements for its use in optimization since it presents a steady behavior, obtaining competitive results that suggest that MSBO can be use as a tool for real world problems. However, the algorithm still needs new adjustments, so it is proposed as a future research work, focusing in the algorithm tuning and in the use of normalization functions for other case studies; this is in order to validate its operation as a tool in the synthesis to feasible solutions, accordingly with the constraints consider in the modeling stage.

REFERENCES

Capistran-Gumersindo, E. (2015). *Cómputo inspirado en la naturaleza para la optimizacion mono-objetivo de un efector final de tres dedos.* Master's thesis, Laboratorio Nacional de Informatica Avanzada, Mexico.

Deb, K. (2000). An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering.*

Duan, H., Li, S., and Shi, Y. (2013). Predator–prey brain storm optimization for dc brushless motor. *IEEE Transactions on Magnetics*, 49(10), 5336–5340.

Fernández-Jiménez, M. (2014). *Optimización de un mecanismo de cuatro barras usando el algoritmo de optimización basado en tormenta de ideas.* Master's thesis, Laboratorio Nacional de Informatica Avanzada, Mexico.

Jiménez, V.P., Sánchez, O.F.A., and Mauledoux, M. (2014). Diseño de un efector final de tres dedos para agarre óptimo. *DYNA: revista de la Facultad de Minas. Universidad Nacional de Colombia. Sede Medellín*, 81(184), 93–101.

Jordehi, A.R. (2015). Brainstorm optimisation algorithm (bsoa): An efficient algorithm for finding optimal location and setting of facts devices in electric power systems. *International Journal of Electrical Power & Energy Systems*, 69, 48–57.

Norton, R.L. (1995). *Diseño de Maquinaria, una Introduccion a la Sintesis y Analisis Mecanismos y Maquinas.* McGraw Hill.

Osborn, A. (1957). *Applied Imagination-Principles and Procedures of Creative Writing.* Charles Scribner's Sons.

Portilla-Flores, E., Santiago-Valentin, E., Solano-Palma, A., Vega-Alvarado, E., and Calva-Yáñez, B. (2014). Cálculo de fuerza para agarre óptimo de un efector final de tres dedos utilizando el algoritmo de forrajeo de bacterias. *Second International Conference on Advanced Mechatronics, Design and Manufacturing Technology.*

Sanchez-Marquez, A., Vega-Alvarado, E., Portilla-Flores, E.A., and Mezura-Montes, E. (2014). Synthesis of a planar four-bar mechanism for position control using the harmony search algorithm. In *Electrical Engineering, Computing Science and Automatic Control (CCE), 2014 11th International Conference on*, 1–6. IEEE.

Santiago-Valentin, E. (2016). *Optimización de sistemas mecatronicos utilizando herramientas de inteligencia ar-*

Table 6. Statistical analysis - *CS3*

Weight 0.1+0.9					
Average	1.1679	σ^2	0.6338	Best	0.13004
Median	1.265	σ	0.7961	Worse	2.64519
Weight 0.5+0.5					
Average	2.57	σ^2	6.1875	Best	0.191
Median	1.5815	σ	2.4874	Worse	8.2547
Weight 0.9+0.1					
Average	2.4647	σ^2	15.435	Best	0.4662
Median	0.8776	σ	3.9288	Worse	12.4385
Weight 1+0					
Average	3.7664	σ^2	29.218	Best	0.3041
Median	0.633	σ	5.4054	Worse	13.508

tificial. Master's thesis, Instituto Politécnico Nacional, México.

Shi, Y. (2011). Brain storm optimization algorithm. In *Advances in Swarm Intelligence*, 303–309. Springer.

Zhan, Z.h., Chen, W.n., Lin, Y., Gong, Y.j., Li, Y.L., and Zhang, J. (2013). Parameter investigation in brain storm optimization. In *Swarm Intelligence (SIS), 2013 IEEE Symposium on*, 103–110. IEEE.

Zhan, Z.h., Zhang, J., Shi, Y.h., and Liu, H.l. (2012). A modified brain storm optimization. In *Evolutionary Computation (CEC), 2012 IEEE Congress on*, 1–8. IEEE.

Appendix A. RESULTS OF THREE STUDY CASES OF THE MAIN WEIGHTS

Table A.1. Main weights of the CS1

Weight 0.1+0.9						
	r_1	r_2	r_3	r_4	r_{cx}	r_{cy}
Best	36.084	34.994	36.084	36.084	35.799	35.894
Worse	49.701	12.906	13.411	49.663	-1.164	12.869
	θ_0	x_0	y_0	θ_2^1	θ_2^2	θ_2^3
Best	4.078	-17.102	41.158	3.095	3.266	3.427
Worse	5.703	19.934	22.020	0.548	0.714	0.892
	θ_2^4	θ_2^5	θ_2^6	**F.O**		
Best	3.578	3.725	3.868	**0.0706**		
Worse	1.087	1.335	1.681	**2.0423**		
Weight 0.5+0.5						
	r_1	r_2	r_3	r_4	r_{cx}	r_{cy}
Best	30.476	27.127	30.044	30.46	25.245	25.209
Worse	41.972	23.304	41.962	41.96	-50.923	-44.808
	θ_0	x_0	y_0	θ_2^1	θ_2^2	θ_2^3
Best	3.802	-8.144	52.152	2.61	2.7974	2.991
Worse	1.9	-60	18.436	3.484	3.5913	3.696
	θ_2^4	θ_2^5	θ_2^6	**F.O**		
Best	3.191	3.399	3.606	**0.1538**		
Worse	3.797	3.894	3.987	**2.4323**		
Weight 0.9+0.1						
	r_1	r_2	r_3	r_4	r_{cx}	r_{cy}
Best	39.452	14.133	22.785	37.883	23.342	17.332
Worse	43.732	6.0066	10.835	43.717	-28.829	59.999
	θ_0	x_0	y_0	θ_2^1	θ_2^2	θ_2^3
Best	3.696	0.309	52.83	2.267	2.636	2.985
Worse	2.862	-49.718	36.512	3.882	4.021	4.155
	θ_2^4	θ_2^5	θ_2^6	**F.O**		
Best	3.343	3.717	4.115	**0.1353**		
Worse	4.284	4.411	4.534	**1.3**		
Weight 1+0						
	r_1	r_2	r_3	r_4	r_{cx}	r_{cy}
Best	50.44	6.191	8.794	47.872	-8.510	-10.107
Worse	59.527	8.582	8.595	59.518	8.484	-40.094
	θ_0	x_0	y_0	θ_2^1	θ_2^2	θ_2^3
Best	4.513	27.018	35.655	0.508	1.022	1.385
Worse	3.253	59.999	46.489	1.153	1.277	1.406
	θ_2^4	θ_2^5	θ_2^6	**F.O**		
Best	1.716	2.049	2.417	**0.0019**		
Worse	1.531	1.650	4.641	**1.6361**		

Table A.2. Main weights of the CS2

Weight 0.1+0.9				
	r_1	r_2	r_3	r_4
Best	1.861134	1.790211	1.811698	1.861134
Worse	1.861238	1.790278	1.811783	1.861238
	r_{cx}	r_{cy}	**F.O**	
Best	1.808446	1.846364	**0.092499**	
Worse	1.808516	1.84644	**0.092499**	
Weight 0.5+0.5				
	r_1	r_2	r_3	r_4
Best	1.927416	1.785747	1.927416	1.927416
Worse	1.925981	1.784889	1.925981	1.925981
	r_{cx}	r_{cy}	**F.O**	
Best	1.69976	1.946801	**0.12476**	
Worse	1.699511	1.947457	**0.124764**	
Weight 0.7+0.3				
	r_1	r_2	r_3	r_4
Best	1.946624	1.844081	1.946624	1.946624
Worse	1.957514	1.850633	1.957514	1.957514
	r_{cx}	r_{cy}	**F.O**	
Best	1.585674	1.963814	**0.093646**	
Worse	1.587307	1.956146	**0.093749**	
Weight 0.9+0.1				
	r_1	r_2	r_3	r_4
Best	1.990289	1.921494	1.990289	1.990289
Worse	2.012848	1.934196	2.012847	2.012848
	r_{cx}	r_{cy}	**F.O**	
Best	1.477738	1.947855	**0.037909**	
Worse	1.482921	1.929627	**0.038037**	
Weight 1+0				
	r_1	r_2	r_3	r_4
Best	20.32431	2.12592	20.31793	20.3206
Worse	33.67168	2.040306	33.64924	33.65048
	r_{cx}	r_{cy}	**F.O**	
Best	2.252166	-0.04672	**0.002893**	
Worse	2.328561	-0.10588	**0.003544**	

Table A.3. Main weights of the CS3

Weight 0.1+0.9					
	r_1	r_2	r_3	r_4	r_{cx}
Best	2.700134	2.650182	2.700054	2.700064	2.682684
Worse	36.04836	5.31E-06	32.4994	35.81544	29.10311
	r_{cy}	θ_0	x_0	y_0	θ_2^1
Best	2.68294	0.12403	1.440574	-0.78327	0.004591
Worse	36.04546	3.409883	-28.3069	38.10563	0.506397
	θ_2^2	θ_2^3	θ_2^4	θ_2^5	θ_2^6
Best	0.005812	0.005814	2.853636	3.057053	3.342652
Worse	0.747988	2.205879	2.329498	2.877531	4.023138
	θ_2^7	θ_2^8	θ_2^9	θ_2^{10}	**F.O**
Best	3.419362	3.419477	3.419516	6.283183	**0.130049**
Worse	4.055401	4.955032	5.078083	5.825513	**2.645194**
Weight 0.5+0.5					
	r_1	r_2	r_3	r_4	r_{cx}
Best	1.22209	1.206991	1.213745	1.21838	1.219299
Worse	44.80746	8.87E-07	42.24817	44.78088	-40.4217
	r_{cy}	θ_0	x_0	y_0	θ_2^1
Best	1.216139	5.273249	1.36589	1.61437	3.49E-05
Worse	15.79192	2.947508	-36.8667	-19.0761	0.771197
	θ_2^2	θ_2^3	θ_2^4	θ_2^5	θ_2^6
Best	3.56587	3.876966	4.367836	4.92191	5.350395
Worse	1.153899	1.687836	2.513014	3.569076	3.617679
	θ_2^7	θ_2^8	θ_2^9	θ_2^{10}	**F.O**
Best	5.614382	5.766777	6.26176	6.275974	**0.191004**
Worse	4.006044	4.504363	4.716411	6.177327	**8.2547**
Weight 0.7+0.3					
	r_1	r_2	r_3	r_4	r_{cx}
Best	4.74585	0.77120	4.67949	4.71595	3.58997
Worse	50.55098	0.98628	41.34927	41.43773	2.519878
	r_{cy}	θ_0	x_0	y_0	θ_2^1
Best	3.307737	5.932793	0.394543	-2.53523	0.514331
Worse	-18.3359	4.160592	17.42196	10.73462	2.258997
	θ_2^2	θ_2^3	θ_2^4	θ_2^5	θ_2^6
Best	0.880699	2.16406	2.750446	3.303229	3.807311
Worse	2.479596	2.852924	3.796082	4.388886	4.768158
	θ_2^7	θ_2^8	θ_2^9	θ_2^{10}	**F.O**
Best	4.261585	5.273711	5.594033	6.14547	**0.590077**
Worse	4.992503	4.997823	4.998617	4.998661	**2.226639**
Weight 0.9+0.1					
	r_1	r_2	r_3	r_4	r_{cx}
Best	54.92273	1.007059	35.39606	45.01554	23.03694
Worse	58.62993	3.87E-07	43.21731	43.27921	5.305051
	r_{cy}	θ_0	x_0	y_0	θ_2^1
Best	13.63005	5.746743	-14.44	-19.5711	0.535914
Worse	-24.2068	2.007463	-1.31272	-22.4677	0.981415
	θ_2^2	θ_2^3	θ_2^4	θ_2^5	θ_2^6
Best	1.023879	2.019396	2.633527	3.192861	3.719558
Worse	1.499258	1.893784	2.620466	3.169049	3.557299
	θ_2^7	θ_2^8	θ_2^9	θ_2^{10}	**F.O**
Best	4.345817	4.921324	5.297749	5.998191	**0.466286**
Worse	4.020403	5.104533	5.242441	5.635895	**12.43857**
Weight 1+0					
	r_1	r_2	r_3	r_4	r_{cx}
Best	43.8394	1.114893	5.066177	43.83902	3.456882
Worse	58.77674	2.33E-15	46.45654	48.70741	6.787923
	r_{cy}	θ_0	x_0	y_0	θ_2^1
Best	-0.74177	5.357885	-2.21474	0.95963	0.565488
Worse	20.25242	1.795314	15.26328	18.06667	0.060799
	θ_2^2	θ_2^3	θ_2^4	θ_2^5	θ_2^6
Best	1.01483	2.03882	2.604173	3.137856	3.670335
Worse	0.543971	1.060257	1.391867	2.104119	2.653932
	θ_2^7	θ_2^8	θ_2^9	θ_2^{10}	**F.O**
Best	4.314163	4.900529	5.312059	6.053438	**0.304117**
Worse	3.204741	5.267743	5.746124	6.227666	**13.50808**

Trajectory Following of Truck-Trailer Mobile Robots Integrating Linear and Fuzzy Control

Antonio Moran

Pontifical Catholic University of Peru, Lima, PERU
amoran@pucp.edu.pe

Abstract: The modeling and autonomous control of truck-trailer mobile robots for trajectory following are addressed. The robot kinematical model is analyzed and used for designing a positioning control system based on linear controllers integrated in a fuzzy-logic approach. The design takes into account both positioning performance and jack-knife avoidance. The results of robot positioning control are extended to trajectory following for which a novel strategy is proposed applicable to general shape desired trajectories. The effectiveness of the proposed methods are verified for linear, circular and sinusoidal trajectories where the mobile robot converges to the desired trajectories, avoiding jack-knife positions and with bounded values of input steering angle.

Keywords: Mobile robot, Fuzzy control, Linear-fuzzy integration, Trailer-type robot, Trajectory following

1. INTRODUCTION

Autonomous truck-trailer mobile robots are used in diverse fields in industry given their advantages in delivery and transportation applications. They can accomplish transportation tasks in a faster and cheaper way compared to multiple individual mobile robots. Their transportation capacity increases with the number of trailers pulled or pushed by a truck moving forward or backward.

However, truck-trailer mobile robots configure a complex, nonlinear, unstable, underactuated and nonholonomic system difficult to control, especially when moving backwards, which have led to an intensive research work for analyzing their motion characteristics and autonomous control. The most of work have been based on robot kinematical model valid when the robot moves at low speeds without wheels side-slipping. In this condition, the robot motion is determined only by geometrical considerations independent of masses, inertias and road friction forces.

Diverse control strategies have been proposed to make the truck-trailer robot autonomously moves describing desired trajectories in complex environments. Approximate linearization and feedback linearization of kinematic model equations were used in David et al. (2014), Altafini et al. (2001) and Laumond et al. (1998) for designing stabilizing controllers for robot positioning applicable to a limited range of operating conditions. Chained representation of robot kinematical equations have been used in Sordalen et al. (1993) and Fierro et al. (1995) for designing nonlinear controllers based on feedback linearization and backstepping techniques for positioning and path tracking control. The differentially flat structure of mobile robots has been used for designing controllers in Rouchon et al, (1993) and Michalek et al. (2012).

Fuzzy logic have been used in Cheng et al. (2009), Tanaka et al. (1994) and Kong et al. (1992) to propose diverse control strategies based on human driver experience expressed through linguistic rules. Neural networks have been applied in Nguyen et al. (1989) and Moran (2004) for training connectionist controllers based on static or dynamic learning algorithms. Other techniques based on genetic algorithms and their integration with neural networks and fuzzy systems, have been proposed in Kinjo et al. (2000). The control schemes have been applied to robot positioning, backing up, linear and nonlinear trajectory following, path planning, parallel parking, jack-knife avoidance, robots formation among other control objectives.

2. PROBLEM DEFINITION AND CONTROL STRATEGY

The problem to be solved is the designing of an autonomous control system for the positioning and trajectory following of a truck-trailer mobile robot. The positioning control problem is shown in Figure 1: the mobile robot, starting from arbitrary initial positions, should achieve the desired position without colliding with obstacles around the goal position.

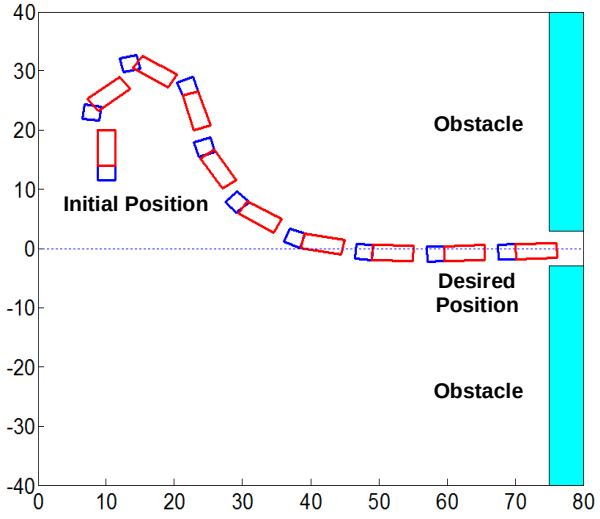


Fig. 1. Mobile robot positioning control problem.

Figure 2 shows a truck-trailer vehicle consisting of a truck with front steering wheels and traction wheels, and a passive trailer with support rear wheels. The trailer is articulated to the truck at the midpoint of the traction axis and it is pulled or pushed by the truck as it moves forward or backward.

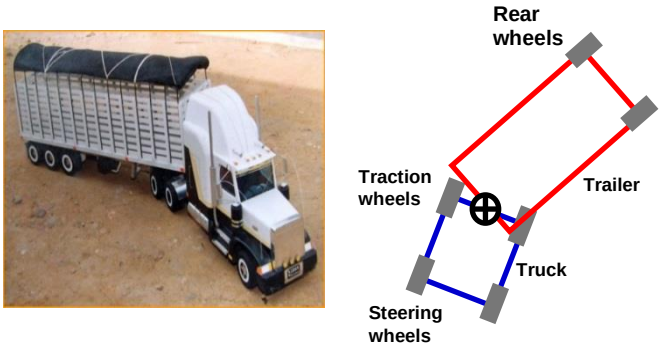


Fig. 2. Truck-trailer mobile robot.

Assuming that left and right wheels move in a similar pattern, the truck-trailer robot can be modeled as a two articulated bars as it is shown in Figure 3. Coordinates (x, y) represent the position of trailer rear wheel, θ_1 and θ_2 are the angles of truck and trailer respect to X axis, θ_{12} is the angle of truck respect to trailer, δ is the steering angle, and L_1 and L_2 are the lengths of truck and trailer, respectively. Counter clockwise angles are positive.

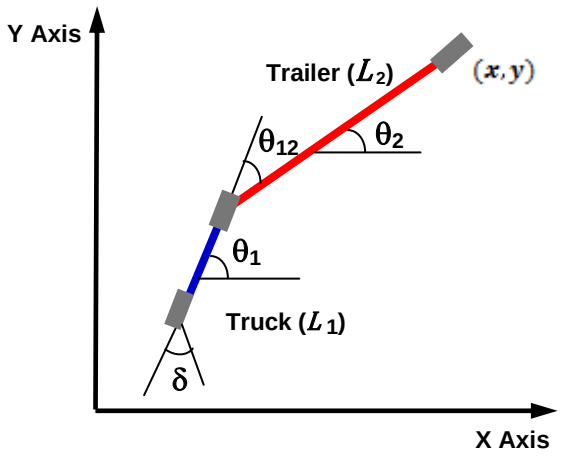


Fig. 3. Two-bars model of truck-trailer robot.

Considering the robot moves at low speeds, it can be assumed that wheels do not side-slip so that lineal velocities of traction and rear wheels are aligned to truck and trailer axis, respectively. Under this consideration, the truck-trailer mobile robot model is given by the following equations:

$$\dot{x} = v \cos\theta_{12} \cos\theta_2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (1)$$

$$\dot{y} = v \cos\theta_{12} \sin\theta_2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2)$$

$$\dot{\theta}_1 = -\frac{v}{L_1} \tan\delta \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3)$$

$$\dot{\theta}_2 = -\frac{v}{L_2} \sin\theta_{12} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4)$$

The truck-trailer angle θ_{12} is:

$$\theta_{12} = \theta_1 - \theta_2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (5)$$

and from equations (3), (4) and (5) the equation of $\dot{\theta}_{12}$ is obtained:

$$\dot{\theta}_{12} = \frac{v}{L_2} \sin\theta_{12} - \frac{v}{L_1} \tan\delta \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (6)$$

Considering that the traction wheels moves at constant backward speed (v=constant) and defining the state vector $\mathbf{x}$ and control vector $\mathbf{u}$ as:

$$\mathbf{x} = [\, y \;\; \theta_2 \;\; \theta_{12} \,]^T \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (7)$$

$$\mathbf{u} = \tan\delta \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (8)$$

equations (2), (4) and (6) can be represented by the following affine nonlinear state-space equation:

$$\dot{\mathbf{x}} = \boldsymbol{f}(\mathbf{x}) + \boldsymbol{g}(\mathbf{x})\mathrm{u} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (9)$$

The no inclusion of coordinate x in state vector $\mathbf{x}$ simplifies the controller design process without affecting the robot positioning and trajectory following capacity as far as it moves only forward or only backward.

To achieve the goal position without colliding with obstacles, it is proposed a control strategy in which the robot prioritizes the achievement of line y^*=0 with horizontal inclination θ_2^*=0°, θ_{12}^*=0° and, afterwards the robot moves straightforward to the goal position as it is shown in Figure 1. With this strategy, coordinate x is not required for control and it is applicable if there is enough space between the robot initial position and the goal position. Considering only three variables $(y, \theta_2, \theta_{12})$ the positioning control problem turns to be a stabilization problem.
Linearizing equations (2), (4) and (6) around the desired angles θ_2^*=0° and θ_{12}^*=0°, the following linear state-space equation is obtained:

$$\begin{bmatrix} \dot{y} \\ \dot{\theta}_2 \\ \dot{\theta}_{12} \end{bmatrix} = \begin{bmatrix} 0 & v & v \\ 0 & v & -v/L_2 \\ 0 & v & v/L_2 \end{bmatrix} \begin{bmatrix} y \\ \theta_2 \\ \theta_{12} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -v/L_2 \end{bmatrix} \tan\delta \qquad \ldots\ldots\ldots\ldots \quad (10)$$

A full-state stabilizing control law for the linear system is given by:

$$\tan\delta = -k_1 y - k_2 \theta_2 - k_3 \theta_{12} \qquad \ldots\ldots\ldots\ldots\ldots \quad (11)$$

where coefficients k_1, k_2 and k_3 are properly chosen so that the closed-loop linear system is stable. This control law is only valid around θ_{12}=0 and it is not guaranteed it will stabilize the mobile robot for other angles θ_{12} in the range -90° to +90°, where the extreme values correspond to jack-knife positions. To solve this problem, a rule-based fuzzy control will be applied: the range of variation of θ_{12} will be partitioned in three parts and a simple linear controller is designed for each part. Afterwards, the three controllers are integrated in a fuzzy-logic approach as a weighted sum of their outputs (weights given by the membership values of each partition).

Figure 4 shows the partitions and membership functions of angle θ_{12} in the range -90° to +90° (*Negative Big*, *Zero*, *Positive Big*). Membership functions (MF) are described by the following equations:
Negative Big:

IF $-90^o \leq \theta_{12} \leq -60^o$ MF = 1
IF $-60^o < \theta_{12} \leq 0^o$ MF = $-\theta_{12}/60$
IF $0^o < \theta_{12} \leq 90^o$ MF = 0

Zero:

IF $-90^o \leq \theta_{12} \leq -60^o$ MF = 0
IF $-60^o < \theta_{12} \leq 0^o$ MF = $\theta_{12}/60 + 1$
IF $0^o < \theta_{12} \leq 60^o$ MF = $-\theta_{12}/60 + 1$
IF $60^o \leq \theta_{12} \leq 90^o$ MF = 0

Positive Big:

IF $-90^o \leq \theta_{12} \leq 0^o$ MF = 0
IF $0^o < \theta_{12} \leq 60^o$ MF = $\theta_{12}/60$
IF $60^o < \theta_{12} \leq 90^o$ MF = 1

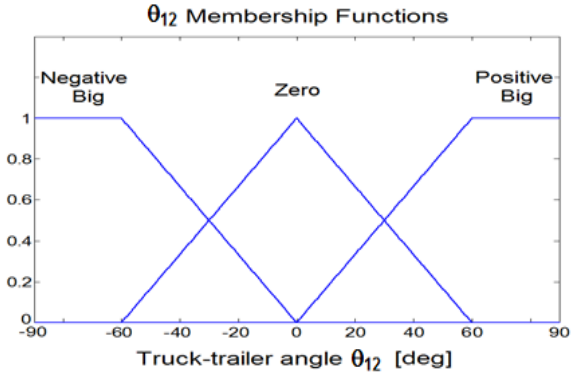


Fig. 4. Partitions and membership functions of truck-trailer angle θ_{12}.

As it was stated, control law (11) is valid when θ_{12}=*Zero*, but it does not apply for other linguistic values of θ_{12}. Then, different controllers will be designed for the other two partitions (*Negative Big* and *Positive Big*).

From equation (6), it is noted that there is an inverse relationship between $\dot{\theta}_{12}$ and $\tan(\delta)$: if $\tan(\delta)$ increases, $\dot{\theta}_{12}$ decreases and vice versa. Also, it is clear that one of the positioning control objectives is to keep truck-trailer angle θ_{12} at small values in order to avoid jack-knife positions. To do that, the following fuzzy reasoning is applied: if θ_{12} is *Positive Big*, $\dot{\theta}_{12}$ should be negative for bringing θ_{12} toward zero and, to achieve that, $\tan(\delta)$ should be positive. Similarly, when θ_{12} is *Negative Big*, $\dot{\theta}_{12}$ should be positive for bringing θ_{12} toward zero and, to achieve that, $\tan(\delta)$ should be negative. This reasoning is summarized in the following fuzzy rules:

IF θ_{12} = *Positive Big* THEN $\delta = \delta_{max}$ (positive)
IF θ_{12} = *Zero* THEN δ = Equation (11)
IF θ_{12} = *Negative Big* THEN $\delta = \delta_{min}$ (negative)
……(12)

It is important to point out that the proposed control strategy does not only focus on attainment of the desired position but also on the avoidance of jack-knife positions (angle θ_{12} close to +90° or -90°). This fuzzy control law was applied to the

mobile robot in order it attains different desired positions starting from arbitrary initial positions. Figure 1 and Figure 5 show the trajectories of the mobile robot from two different initial positions to the goal position x^*=80, y^*=0, θ_2^*=0°, θ_{12}^*=0°. In both cases, the robot is able to asymptotically achieve the goal position without colliding with obstacles around. The steering angle δ was bounded to the range from δ_{min} = -30° to δ_{max} = +30°.

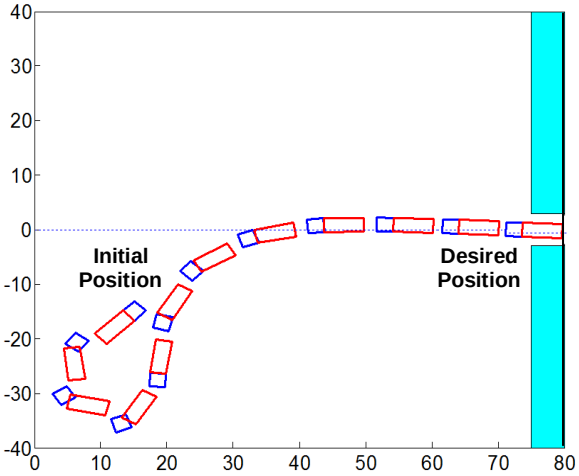


Fig. 5. Trajectory of mobile robot starting from initial position (x=10, y=-20, θ_2=-135°, θ_{12}=0°) and achieving the goal position at y^*=0.

The control law (11) can be easily modified to make the robot achieves other fixed goal positions as follows:

$$\tan\delta - \tan\delta^* = -k_1(y - y^*) - k_2(\theta_2 - \theta_2^*) - k_3(\theta_{12} - \theta_{12}^*) \quad \text{..... (13)}$$

where y^*, θ_2^* and θ_{12}^* represent the desired position of the mobile robot, and δ^* is the corresponding steering angle. It is important to coherently set the desired values of y^*, θ_2^*, θ_{12}^* and δ^* to physically realizable values to attain consistent robot responses. Figure 6 shows the trajectory of the mobile robot from an arbitrary initial position to goal position x^*=80, y^*=20, θ_2^*=0°, θ_{12}^*=0° with δ^*=0° which represents a consistent and physically attainable robot position at convergence.

It is important to note that, since velocity v is constant, when the robot converges to line y=y^* and moves along it toward the goal position, the value of $\dot{x}$ equals to v. It is equivalent to state that coordinate x is proportional to time and behaves as an independent variable.

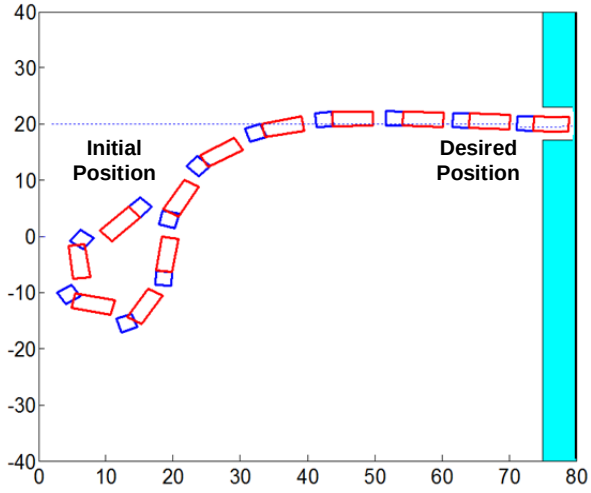


Fig. 6. Trajectory of mobile robot starting from initial position (x=10, y=0, θ_2=-135°, θ_{12}=0°) and achieving the goal position at y^*=20.

3. TRAJECTORY FOLLOWING

The controller given by equation (13) can be applied to the mobile robot for following any desired trajectory. To do that, instantaneous proper desired values of y^*, θ_2^*, θ_{12}^* and δ^* should be determined based on the robot kinematical equations and the geometry of the desired trajectory. The desired values will be found using the *perpendicular desired position methodology* explained afterwards.

3.1 Perpendicular desired position

In this approach, a perpendicular line is drawn from the present robot position coordinates (x, y) to the desired trajectory. The coordinate y of the intersection point represents the instantaneous desired coordinate y^*, and the angle of the tangent to the desired trajectory in the intersection point represents the desired trailer inclination angle θ_2^*. Using these values, the desired values θ_{12}^* and δ^* can be obtained from the robot kinematical equations and the desired trajectory equation. It is important to note that the perpendicular line between the robot present position and the desired trajectory represents the instantaneous minimum distance between them. This methodology is applicable when the computation of the intersection point is not cumbersome and it is unique. In the following, the methodology will be explained for linear and circular desired trajectories.

3.1.1 Linear desired trajectory

Figure 7 shows the linear trajectory control problem. Point P represents the robot instantaneous position given by coordinates (x, y) and line AB, with inclination angle α, represents the trajectory to be followed. The equation of line AB is:

$$y^* = ax + b \qquad \ldots\ldots\ldots\ldots\ldots\ldots \quad (14)$$

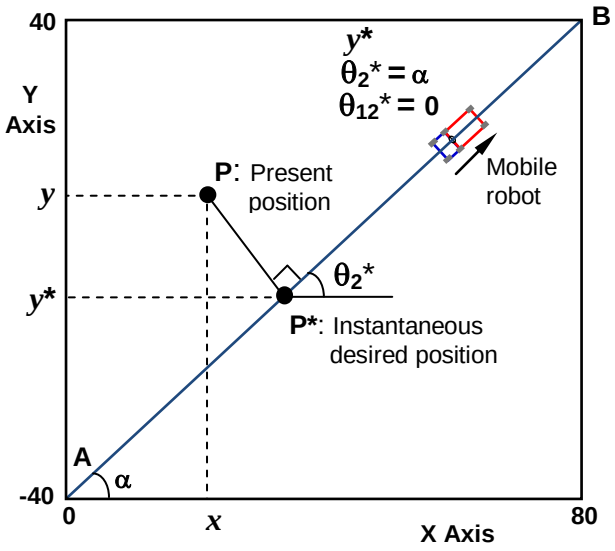


Fig. 7. Linear trajectory following. P* represents the instantaneous desired position for computing coordinate y^* and inclination angle θ_2^*.

Drawing a perpendicular line from point P to line AB, point P* is obtained whose coordinate y^* represents the instantaneous desired value of robot coordinate y. By geometrical relationships, the value of y^* is given by:

$$y^* = \frac{a\,x + a^2 y + b}{1 + a^2} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (15)$$

Considering that the tangent to line AB at point P* is the same line, the desired angle θ_2^* is equal to the line inclination angle α. Also, considering that truck and trailer should be aligned to line AB, it is clear that the desired value of truck-trailer angle $\theta_{12}^*=0^o$, and the desired value of steering angle $\delta^*=0^o$. Figure 8 (a) and (b) show the trajectory of the mobile robot following a linear trajectory with inclination angle $\alpha=45^o$ given by the equation:

$$y^* = x - 40 \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (16)$$

starting from two different initial positions. In both cases, the robot asymptotically converges to the desired linear trajectory without steady-state error which verifies the effectiveness of the proposed control strategy. Similarly as the positioning control problem presented in the previous section, since velocity v is constant, when the robot converges to the desired linear trajectory and moves along it, the value of $\dot{x}$ equals to $v\cos\alpha$ which is constant. It is equivalent to state that coordinate x is proportional to time and behaves as an independent variable.

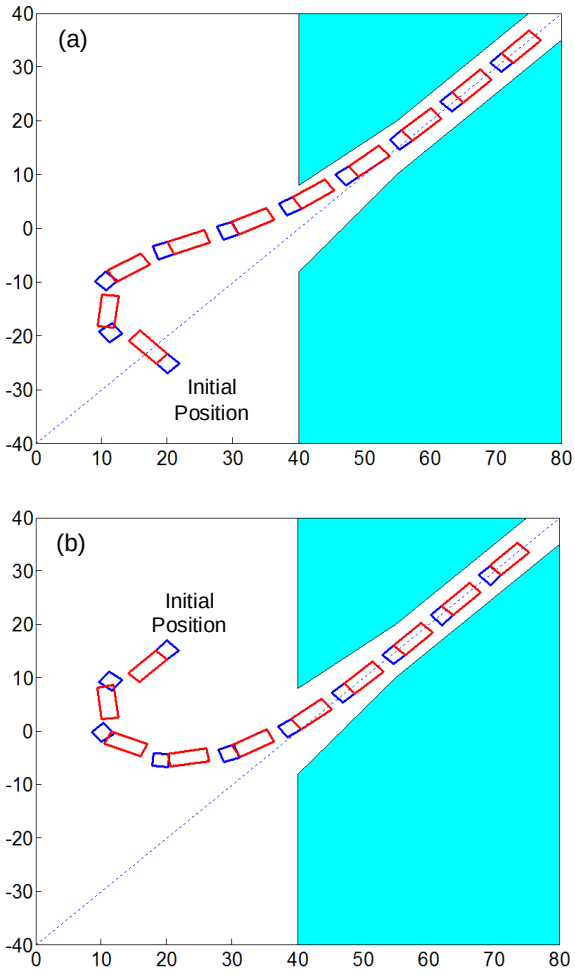


Fig. 8. Trajectories of mobile robot from two different initial positions converging into the desired linear trajectory.
Initial position (a) x=15, y=-20, θ_2=135°, θ_{12}=0°
(b) x=15, y=10, θ_2=135°, θ_{12}=0°

3.1.2. Circular Desired Trajectory

Figure 9 shows the circular trajectory control problem. Point P represents the robot instantaneous position given by coordinates (x, y) and the circular line represents the trajectory to be followed. The equation of the circular path with center C and coordinates (x_C, y_C), and radius R is:

$$y = \sqrt{R^2 - (x - x_c)^2} + y_c \qquad (x_c - R) \le x \le x_c$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (17)$$

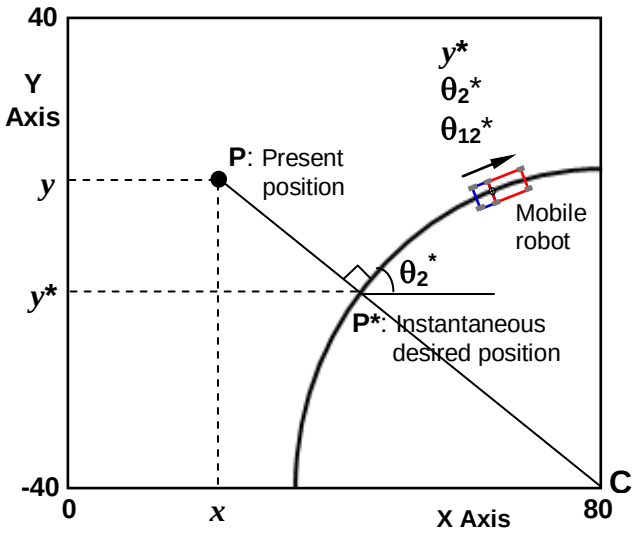


Fig. 9. Circular trajectory following. P* represents the instantaneous desired position for computing coordinate y^* and inclination angle θ_2^*.

Drawing a perpendicular line from point P to the circular path, point P* is obtained whose coordinate y^* represents the instantaneous desired value of robot coordinate y. The perpendicular line also passes by the center C of the circular path, and the distance PP* represents the minimum distance from P to the circular path. By geometrical relationships, the value of y^* is given by:

$$y^* = \frac{R(y-y_c)}{\sqrt{(x-x_c)^2+(y-y_c)^2}} - y_c \qquad \ldots\ldots\ldots\ldots \quad (18)$$

The inclination angle θ_2^* of the tangent to the circular trajectory at point P* represents the desired instantaneous inclination angle of the trailer and can be determined by geometrical relationships as:

$$\theta_2^* = \tan\left(\frac{x_c-x}{y-y_c}\right) \qquad \ldots\ldots\ldots\ldots\ldots\ldots \quad (19)$$

Differentiating equation (19) with respect to time and replacing the expressions of $\dot{x}$ and $\dot{y}$ given by equations (1) and (2) the expression of the desired truck-trailer angle θ_{12}^* is obtained as:

$$\theta_{12}^* = \operatorname{atan}\left(\frac{L_2}{R}\right) \qquad \ldots\ldots\ldots\ldots\ldots\ldots \quad (20)$$

It is noted that the value of θ_{12}^* is constant and does not depend on robot coordinates or angles. This result is expected considering the truck and trailer relative position required to describe circular trajectories, and it is the same as the result presented in Michalek et al. (2012).

Finally, considering that $\dot{\theta}_{12}^*=0$ and replacing the values of θ_{12}^* and $\dot{\theta}_{12}^*$ in equation (6), the value of δ^* is obtained which is also constant as it is expected for circular trajectories:

$$\delta^* = \operatorname{atan}\left(\frac{L_1}{\sqrt{L_2^2+R^2}}\right) \qquad \ldots\ldots\ldots\ldots\ldots\ldots \quad (21)$$

Figure 10 (a) and (b) show the trajectories of the mobile robot following a circular trajectory with center in point (80,-40) and radius R=50, starting from two different initial positions. In both cases, the robot asymptotically converges to the desired circular trajectory without steady-state error which verifies the effectiveness of the proposed control strategy.

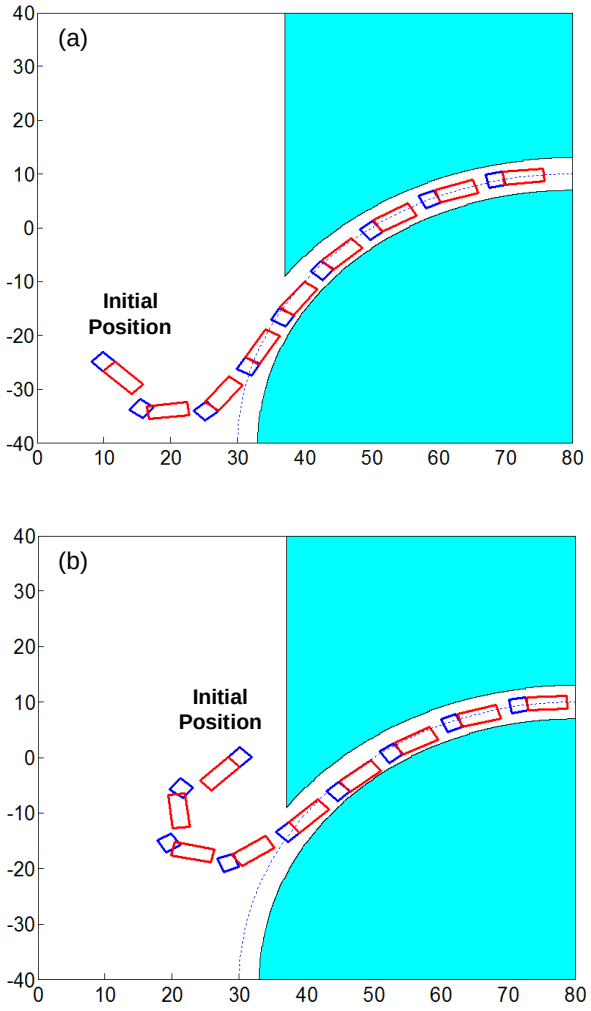


Fig. 10. Trajectories of mobile robot from two different initial positions converging into the circular desired trajectory.
Initial position (a) x=15, y=-30, θ_2=-45°, θ_{12}=0°
(b) x=25, y=-5, θ_2=-135°, θ_{12}=0°

4. EFFECT OF FUZZY PARTITIONS

As it was presented in Section 2, the proposed controller integrates three partitions of truck-trailer angle θ_{12}: *Negative Big, Zero* and *Positive Big*. Considering that the approximate linearized model of equation (10) is defined for partition *Zero* (small values of θ_{12}), the range of this partition plays an important role on the control performance. This effect will be analyzed through two fuzzy controllers, Fuzzy 1 and Fuzzy 2, whose partitions and membership functions are shown in Figure 11: controller Fuzzy 1 with a wider range of partition *Zero* and controller Fuzzy 2 with a narrower partition.

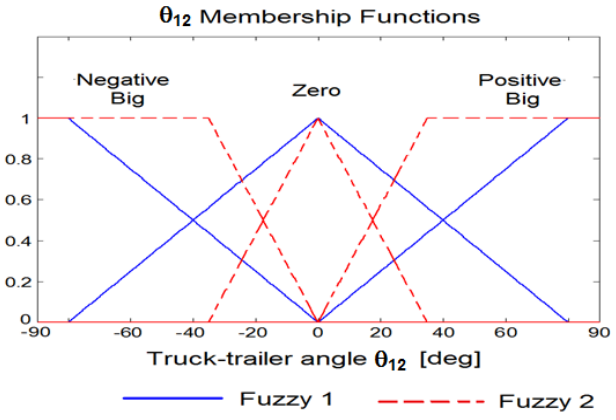


Fig. 11. Partitions and membership functions of truck-trailer angle θ_{12} for controllers Fuzzy 1 y Fuzzy 2.

Membership functions (MF) for controller Fuzzy 1 are described by the following equations:

Negative Big:

IF $-90^o \leq \theta_{12} \leq -80^o$ MF = 1
IF $-80^o < \theta_{12} \leq 0^o$ MF = $-\theta_{12}/80$
IF $0^o < \theta_{12} \leq 90^o$ MF = 0

Zero:

IF $-90^o \leq \theta_{12} \leq -80^o$ MF = 0
IF $-80^o < \theta_{12} \leq 0^o$ MF = $\theta_{12}/80 + 1$
IF $0^o < \theta_{12} \leq 80^o$ MF = $-\theta_{12}/80 + 1$
IF $80^o \leq \theta_{12} \leq 90^o$ MF = 0

Positive Big:

IF $-90^o \leq \theta_{12} \leq 0^o$ MF = 0
IF $0^o < \theta_{12} \leq 80^o$ MF = $\theta_{12}/80$
IF $80^o < \theta_{12} \leq 90^o$ MF = 1

Membership functions (MF) for controller Fuzzy 2 are described by the following equations:

Negative Big:

IF $-90^o \leq \theta_{12} \leq -35^o$ MF = 1
IF $-35^o < \theta_{12} \leq 0^o$ MF = $-\theta_{12}/35$
IF $0^o < \theta_{12} \leq 90^o$ MF = 0

Zero:

IF $-90^o \leq \theta_{12} \leq -35^o$ MF = 0
IF $-35^o < \theta_{12} \leq 0^o$ MF = $\theta_{12}/35 + 1$
IF $0^o < \theta_{12} \leq 35^o$ MF = $-\theta_{12}/35 + 1$
IF $35^o \leq \theta_{12} \leq 90^o$ MF = 0

Positive Big:

IF $-90^o \leq \theta_{12} \leq 0^o$ MF = 0
IF $0^o < \theta_{12} \leq 35^o$ MF = $\theta_{12}/35$
IF $35^o < \theta_{12} \leq 90^o$ MF = 1

Figure 12 shows the trajectory of the mobile robot for both controllers starting from the same initial position and moving toward the same fixed desired position. Although both controllers are able to conduct the mobile robot to the goal position, the robot with controller Fuzzy 1 converges faster to the desired coordinate y^*=0 and describes a trajectory with smaller turning radius. These results are explained by the fact that extreme partitions *Negative Big* and *Positive Big*, having lower membership values in controller Fuzzy 1, impose lesser restrictions on truck-trailer angle θ_{12}, which results in higher values but without reaching unwanted jack-knife positions.

Figure 13 shows the time response of truck inclination angle θ_1, trailer inclination angle θ_2, truck-trailer angle θ_{12}, and steering angle δ corresponding to the trajectories showed in Figure 12. It is noted that the response for controller Fuzzy 1 converges faster than controller Fuzzy 2 at the expense of higher values of truck-trailer angle θ_{12} and steering angle δ. In both cases jack-knife positions are avoided. From these results it is concluded that smaller ranges of the central partition *Zero* result in lower values of angle θ_{12}. These results validate the coherence of the proposed control strategy.

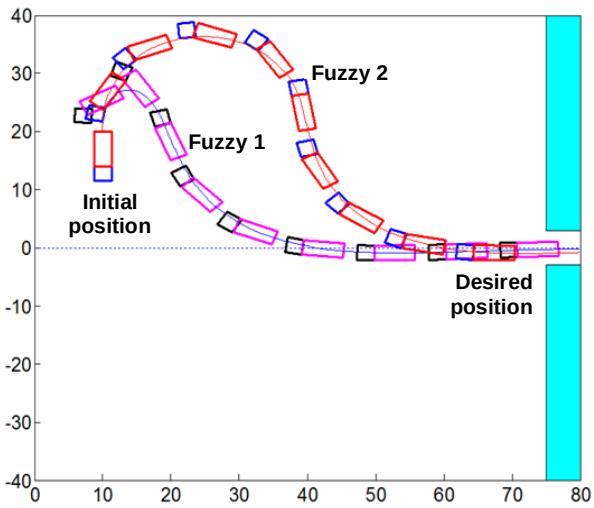


Fig. 12. Trajectories of mobile robot with controllers Fuzzy1 and Fuzzy 2 starting from the same initial position x=10, y=20, θ_2=90°, θ_{12}=0°

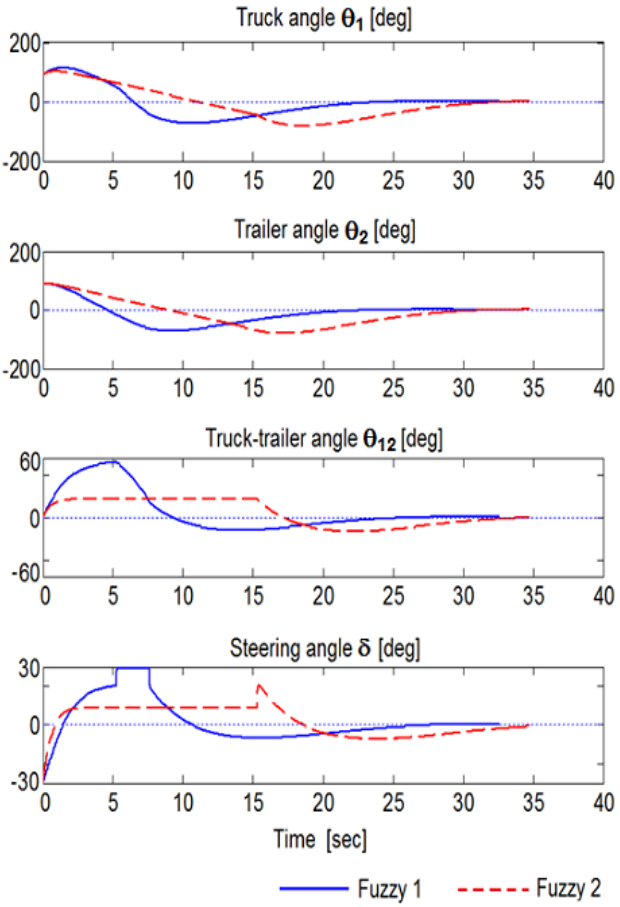


Fig. 13. Time response of truck inclination angle θ_1, trailer inclination angle θ_2, truck-trailer angle θ_{12} and steering angle δ for controllers Fuzzy 1 and Fuzzy 2.

5. CONCLUSIONS

The kinematical model and nonholonomic constraints of truck-trailer mobile robots have been derived and analyzed. A novel control strategy integrating linear controllers in a fuzzy logic approach has been proposed, assuring the robot achieve goal positions avoiding jack-knifing. The effectiveness of the trajectory following control strategies have been verified for linear and circular trajectories where the mobile robot converges to the desired trajectories with bounded values of the steering angle. The effect of the size of truck-trailer angle partitions in fuzzy control was analyzed and it was found that a wider range of the central partition results in faster convergence of the mobile robot at the expense of higher values of truck-trailer angle θ_{12} and steering angle δ. One of the main attributes of the proposed control strategy is the easy incorporation of jack-knife avoidance into the controller designing process.

6. REFERENCES

Altafini, C, Speranzon, A, and Wahlberg, B. (2001), A Feedback Control Scheme for Reversing a Truck and Trailer Vehicle. *IEEE Transactions on Robotics and Automation*, Vol.17, No.6, pp.915-922.

Cheng J., Zhang Y. and Wang Z. (2009), Backward Tracking Control of Mobile Robot with One Trailer Via Fuzzy Line-of-Sight Method, *Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, Vol.6, pp.66-70.

David, J. and Manivannan, P.V. (2014). Control of Truck-Trailer Mobile Robots: A Survey. *Intelligent Service Robotics Journal*, Vol.7, pp.245-258, Springer Verlag.

Fierro, R. and Lewis, F.L. (1995). Control of Nonholonomic Mobile Robot: Backstepping Kinematics into Dynamics. *Proceedings of the 34th IEEE Control and Decision Conference*, Vol.4, pp.3805-3810.

Kinjo, H., Wang,B., and Yamamoto, T. (2000). Backward Movement Control of a Trailer Truck System Using Neuro-Controllers Evolved by Genetic Algorithm. *Proceedings of 26th Conference of IEEE Industrial Electronics Society IECON 2000*, Vol.1, pp.253-258.

Kong, S.G. and Kosko, B. (1992). Adaptive fuzzy systems for backing up a truck-and-trailer. *IEEE Transactions on Neural Networks*, Vol.3, No.2,pp.211-223.

Laumond, J.P., Sekhavat, S. and Lamiraux, F. (1998). Guidelines in Nonholonomic Motion Planning for Mobile Robots. *Lectures Notes in Control and Information Sciences* No. 229, Springer.

Michalek, M. (2012). Tracking Control Strategy for the Standard N-trailer Mobile Robot – A Geometrically Motivated Approach. *Robot Motion Control*, pp.39-51, Springer-Verlag.

Moran, A. (2004). Intelligent Car Parking Using Fuzzy-Neural Networks. *Proceedings of 2nd International Latin American and Caribbean Conference for Engineering and Technology LACCEI*, 2004, Miami, USA.

Nguyen, D. and Widrow, B. (1989). The Truck Backer-Upper: An Example of Self-Learning in Neural Networks, *Proceedings of International Joint Conference on Neural Networks IJCNN*, Vol.2, pp.357-363.

Rouchon, P., Fliess, M., Levine J. and Martin, P. (1993). Flatness, Motion Planning and Trailer Systems, *Proceedings of 32nd IEEE Conference on Decision and Control*, Vol.3, pp.2700-2705.

Sordalen, O.J. (1993). Conversion of the Kinematics of a Car with N Trailers into a Chained Form, *Proceedings of IEEE International Conference on Robotics and Automation*, pp.382-387.

Tanaka, K. and Sano, M. (1994). A Robust Stabilization Problem of Fuzzy Control Systems and its Application to Backing-up Control of a Truck Trailer, *IEEE Transactions on Fuzzy Systems*, Vol.2, No.2, pp.119-134.

Trajectory tracking controller for a nonlinear bioprocess

María N. Pantano*, María C. Fernández*, Mario E. Serrano*, Oscar A. Ortiz*, Gustavo J. Scaglia*.

** Instituto de Ingeniería Química, Universidad Nacional de San Juan (UNSJ), CONICET, Av. Lib. San Martín Oeste 1109, San Juan J5400ARL, Argentina (e-mail: npantano@unsj.edu.ar)*

Abstract: This paper aims to develop a simple but efficient control technique based on a linear algebra approach for tracking optimal profiles of a nonlinear multivariable fed-batch bioprocess. The methodology proposed allows, knowing the desired states, to find the values for the control actions by solving a system of linear equations. Its main advantage is that the condition for the tracking error tends to zero. The efficiency of the proposed controller is tested through several simulations. The optimal controller parameters are selected through Montecarlo Randomized Algorithm in order to minimize a cost index.

Keywords: Multivariable control, trajectory tracking, nonlinear systems, algebraic approaches, optimal trajectories.

1. INTRODUCTION

Fed-batch processes are widely used in the biotechnological industry, which is demanding for more efficient, reliable and safer processes to optimize production and improve power quality (De Battista et al., 2012). In a fed-batch operation, one or more nutrients are gradually supplied to the bioreactor, but no product is withdrawn until the process is finished. Its main advantages are the avoidance of substrate overfeeding which can inhibit the growth of microorganisms and catabolite repression. On the other hand, from the control engineer's viewpoint, the fed-batch fermentation is characterized for a large number of obstacles: complex dynamic behavior of microorganisms, the process model usually contains strongly time-varying parameters, changes in initial conditions, input saturation, external disturbances and the stiffness and nonlinearity of the model equations (Rani and Rao, 1999, Bayen and Mairet, 2013, Renard et al., 2006, Lee et al., 1999, Johnson, 1987).

Several control techniques are studied today associated with optimization and control of bioprocess, such as: bio-inspired algorithms (Rocha et al., 2014), genetic algorithms (Sarkar and Modak, 2003, Sarkar and Modak, 2004), robust control (Renard et al., 2006, Renard and Wouwer, 2008), nonlinear fuzzy control (Cosenza and Galluzzo, 2012),evolutionary algorithms (Ronen et al., 2002), model predictive control (MPC) (Ashoori et al., 2009) and nonlinear MPC (Craven et al., 2014, Santos et al., 2012), adaptive stochastic algorithms (Carrasco and Banga, 1997), neural network model (Saint-Donat et al., 1991, Tholudur and Ramirez, 1996), etc. Most of the control literature for fed-batch cultures focuses on open-loop operation owing to the highly nonlinear and inherently difficult dynamic behavior (Berber, 1996). These methods have good results in biological processes; however, they have limitations regarding the need for advanced specific knowledge, the difficulty of mathematical processing (especially in nonlinear systems), trouble with real-time implementation and the need for a complicated database of the processes (Alford, 2006). Besides that, in the open-loop control strategies, the main disadvantage is that no compensation is made for modeling mismatch or random disturbances during the process operation (Lee et al., 1999, Chang, 2003, Chung et al., 2006, Soni and Parker, 2004). It is therefore important to design a controller to track the optimal policy considering disturbance compensation for the closed-loop control problem.

The aim of this work is to solve the problem of tracking optimal profiles of an important biological system, which has a complex dynamics and a strong nonlinearity. The proposed methodology to achieve the stated objective is based on solving a system of linear equations. One of the key features of this technique is its simple approach, which suggests that knowing the value of the desired state, analyzing the conditions for a system to have an exact solution and then solving the system of linear equations; it can find the values for the control actions, which forces the system to move from its current state to the desired one.

The main advantages of this method are its simplicity, versatility and accuracy even under parametric uncertainty. The methodology for the controller design is very simple, nonlinear model is used; thus, its performance is independent of the operating point, and has an excellent performance against the set point changes. The optimal controller parameters are selected through Montecarlo Experiments in order to minimize a proposed cost index. The computing power required to perform the mathematical operations is low. Furthermore, the developed algorithm is easier to implement in a real system because the use of discrete equations allows direct adaptation to any computer system or programmable device. Moreover, because its simplicity and the mathematical tools that it use, this methodology is applicable to many systems, not only to bioprocesses.

The case study proposed for control is the Lee-Ramirez fed-batch bioreactor (Lee and Ramirez, 1992), developing a mathematical model for the induced foreign protein production by recombinant bacteria in a fed-batch bioreactor.

The advantage of using this system is that it has been already used by a number of researchers using different techniques, so the available data can be used to assess other methods.
The controller efficiency is tested through simulations using Matlab® software. The assays include a simulation in normal operation conditions and then, the control system under parametric uncertainty is analyzed through a Montecarlo randomized algorithm.
The paper is organized as follows. In Section 2, the mathematical model of the proposed system is presented and the optimal profiles are defined. Then, the controller design is considered in Section 3. The results of the simulation tests to demonstrate the efficiency of the controller are shown in Section 4. Finally, Section 5 outlines the conclusions of the work.

2. MATHEMATICAL MODEL

The mathematical model used here is taken of Balsa Canto et al. (Balsa-Canto et al., 2000). Although simple, it can effectively describe the dynamics of the bioprocess.
The original model was developed by Lee and Ramirez (Lee and Ramirez, 1992), who described the dynamics of the process of induced foreign protein production by recombinant bacteria and then used it to obtain an optimal control policy to maximize the foreign protein production with a nutrient and inducer feeding strategy (Lee and Ramirez, 1994). The same problem was studied by Tholudur and Ramirez (Tholudur and Ramirez, 1996) using neural network parameter function models. Carrazco and Banga (Carrasco and Banga, 1997) used adaptive stochastic algorithms to obtain better results. Since the performance index exhibits a very low sensitivity with respect to the controls, Tholudur and Ramirez (Tholudur and Ramirez, 1997) constructed a modified parameter function set to increase the sensitivity to the controls. Balsa-Canto also used the same parameter function set (Balsa-Canto et al., 2000). A genetic algorithm to optimize the same system considering multiple control variables was presented in (Sarkar and Modak, 2004).
The operation of the fed-batch bioreactor considering two control variables (u_1 and u_2, nutrient and inducer feed rates) is described by seven differential equations (1).
The state variables are the reactor volume x_1 *(L)*, the cell density x_2 *(g/L)*, the nutrient concentration x_3 *(g/L)*, the foreign protein concentration x_4 *(g/L)*, the inducer concentration x_5 *(g/L)*, the inducer shock factor on the cell growth rate x_6, and the inducer recovery factor on the cell growth rate x_7 (both dimensionless).
The model parameters were described by Lee and Ramirez (Lee and Ramirez, 1992). The concentration of nutrient feed stream is N, I is the concentration of inducer in the inducer feed stream, and Y is the growth yield coefficient. In addition, g is the specific growth rate, R is the foreign protein production rate, p is a Monod-type constant and K_1, K_2 are the shock and recovery parameters respectively.

$$\left.\begin{aligned}
\dot{x}_1 &= u_1 + u_2 \\
\dot{x}_2 &= x_2 g - \frac{u_1 + u_2}{x_1} x_2 \\
\dot{x}_3 &= \frac{u_1 N}{x_1} - \frac{u_1 + u_2}{x_1} x_3 - \frac{g}{Y} x_2 \\
\dot{x}_4 &= x_2 R - \frac{u_1 + u_2}{x_1} x_4 \\
\dot{x}_5 &= \frac{u_2 I}{x_1} - \frac{u_1 + u_2}{x_1} x_5 \\
\dot{x}_6 &= -K_1 x_6 \\
\dot{x}_7 &= K_2 (1 - x_7)
\end{aligned}\right\} \quad (1)$$

Where,

$$g = \left(\frac{x_3}{14.35 + x_3 \left(1 + \frac{x_3}{111.5} \right)} \right) \left(x_6 + \frac{x_7 0.22}{0.22 + x_5} \right) \quad (2)$$

$$R = \left(\frac{0.233 x_3}{14.35 + x_3 \left(1 + \frac{x_3}{111.5} \right)} \right) \left(\frac{0.0005 + x_5}{0.022 + x_5} \right) \quad (3)$$

$$K_1 = K_2 = \frac{x_5 p}{0.034 + x_5} \quad (4)$$

The two control variables are the glucose feeding rate, u_1 *(L/h)*, and inducer feeding rate u_2 *(L/h)* to the fed-batch bioreactor. The desired variables to follow are the reactor volume x_1, the cell density x_2, and the foreign protein concentration x_4.
It may be noted that the desired trajectories to track are directly the optimal profiles of controlled variables (x_1, x_2, and x_4). These trajectories were obtained by an open-loop simulation of the bioprocess using the optimal feeding policies achieved by Balsa Canto et al. (Balsa-Canto et al., 2000).

3. METHODOLOGY FOR CONTROLLER DESIGN

The proposed controller methodology is based on approximating the differential equations of the mathematical model (1) through the Euler method. Hence, the control problem for tracking optimum profiles of volume (x_1), cell density (x_2), and protein concentration (x_4) is reduced to the resolution of a system of linear equations.
To achieve the control goal, the feed flow rate of nutrient (u_1) and inducer (u_2) are available to be used as control actions. Therefore, the goal is to find the values of u_1 and u_2 such that the variables x_1, x_2 and x_4 follow paths desired with minimal tracking error.

3.1. Controller methodology

The first step for this technique, is rearrange the system of equations (1) in matrix form as $A\boldsymbol{u}=b$. The matrix $\boldsymbol{u}$ is composed of the control variables, for this model, u_1 and u_2:

$$\overbrace{\begin{pmatrix} 1 & 1 \\ -1 & -1 \\ (N-x_3) & -x_3 \\ -1 & -1 \\ -1 & (I-x_5) \end{pmatrix}}^{A} \overbrace{\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}}^{u} = \overbrace{\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \frac{x_1}{x_2} - x_1 g \\ \dot{x}_3 x_1 + \frac{g}{Y} x_2 x_1 \\ \frac{\dot{x}_4 x_1}{x_4} - \frac{R x_2 x_1}{x_4} \\ \dot{x}_5 x_1 \end{pmatrix}}^{b} \quad (5)$$

Remark 1. The variables x_6 and x_7 are not directly related with the control variables, therefore, are not considered in the controller design.

According to the rule of numerical integration of Euler, the differential equations can be approximated as follow:

$$\dot{x}_i = \frac{x_{i,n+1} - x_{i,n}}{T_0} \quad (6)$$

Where T_0 is the sample time, $x_{i,n}$ represents the state variable i in n instant and $x_{i,n+1}$ the state variable i in $(n+1)$ instant.

Defining the following expression:

$$\overbrace{x_{i,ref,n+1} - x_{i,n+1}}^{error_{i,n+1}} = k_i \overbrace{(x_{i,ref,n} - x_{i,n})}^{error_{i,n}} \quad (7)$$

Where, $x_{i,ref,n}$ and $x_{i,ref,n+1}$ are the reference values in the n instant and the next sample time respectively, the constant k_i is the controller parameter for the variable i.

Then, the immediately reachable value of each state variable is:

$$x_{i,n+1} = x_{i,ref,n+1} - k_i \overbrace{(x_{i,ref,n} - x_{i,n})}^{error_{i,n}} \quad (8)$$

Therefore, the values of the real state variables in the next sample time ($x_{i,n+1}$) are function of the reference profiles, the actual state variable and the controller parameters. So, all values are known.

Consequently, substituting (8) in (6):

$$\dot{x}_i = \frac{\overbrace{\left[x_{i,ref,n+1} - k_i (x_{i,ref,n} - x_{i,n})\right]}^{x_{i,n+1}} - x_{i,n}}{T_0} \quad (9)$$

Now, replacing (9) in each differential expression appearing in (5), the process model can be rewritten, see (10).

The values of x_1, x_2, and x_4 are the references to follow, therefore are known (the reference values as well as the real system values). Note that, the Eq. (10) is a system of five equations and two unknowns, which normally has no solution. Therefore, the unknown variables of this system are defined as "*sacrificed variables*" and are written as $x_{i,ez}$ corresponding, in this case, to $x_{3,ez}$ and $x_{5,ez}$. The key of this technique is that the values adopted by such variables forces the equation system (10) to have exact solution, which implies error not only be minimal, but equal zero.

$$\begin{pmatrix} 1 & 1 \\ -1 & -1 \\ (N-x_{3,n}) & -x_{3,n} \\ -1 & -1 \\ -1 & (I-x_{5,n}) \end{pmatrix} \begin{pmatrix} u_{1,n} \\ u_{2,n} \end{pmatrix} =$$

$$\begin{pmatrix} \left(\frac{x_{1,ref,n+1} - k_1 (x_{1,ref,n} - x_{1,n}) - x_{1,n}}{T_0}\right) \\ \left(\frac{x_{2,ref,n+1} - k_2 (x_{2,ref,n} - x_{2,n}) - x_{2,n}}{T_0}\right) \frac{x_{1,n}}{x_{2,n}} - x_{1,n} g \\ \left(\frac{x_{3,ez,n+1} - k_3 (x_{3,ez,n} - x_{3,n}) - x_{3,n}}{T_0}\right) x_{1,n} + \frac{g}{Y} x_{2,n} x_{1,n} \\ \left(\frac{x_{4,ref,n+1} - k_4 (x_{4,ref,n} - x_{4,n}) - x_{4,n}}{T_0}\right) \frac{x_{1,n}}{x_{4,n}} - R x_{2,n} \frac{x_{1,n}}{x_{4,n}} \\ \left(\frac{x_{5,ez,n+1} - k_5 (x_{5,ez,n} - x_{5,n}) - x_{5,n}}{T_0}\right) x_{1,n} \end{pmatrix} \quad (10)$$

To simplify the mathematical treatment, the equations system is expressed as follows:

$$\begin{pmatrix} a_{31} & a_{32} \\ a_{51} & a_{52} \\ a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{41} & a_{42} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} b_3 \\ b_5 \\ b_1 \\ b_2 \\ b_4 \end{pmatrix} \Rightarrow Au = b \quad (11)$$

Remark 2: the order of the rows are altered to the Gaussian elimination because the system presents three linearly dependents rows.

To accomplish the target, the system (10) must have exact solution. Then, the vector ***b*** must be contained in the space formed by the columns of ***A***, ie, the vector ***b*** must be a linear combination of the column vectors of matrix ***A***.

In order to find the values of the $x_{i,ez}$ so that the system to have exact solution, the Gauss elimination process is carried out.

Then, the *necessary* and *sufficient condition* for the system to have exact solution is:

$$\begin{aligned} 0 &= (a_{31}a_{52} - a_{32}a_{51})(a_{31}b_1 - b_3a_{11}) - (a_{31}b_5 - b_3a_{51})(a_{31}a_{12} - a_{32}a_{11}) \\ 0 &= (a_{31}a_{52} - a_{32}a_{51})(a_{31}b_2 - b_3a_{21}) - (a_{31}b_5 - b_3a_{51})(a_{31}a_{22} - a_{32}a_{21}) \\ 0 &= (a_{31}a_{52} - a_{32}a_{51})(a_{31}b_4 - b_3a_{41}) - (a_{31}b_5 - b_3a_{51})(a_{31}a_{42} - a_{32}a_{41}) \end{aligned} \quad (12)$$

This equations system is solved for each sampling period by iteration methods, where the unknown variables $x_{3,ez,n+1}$ and $x_{5,ez,n+1}$ (*sacrificed variables*) are calculated.

Once the values of $x_{3,ez,n+1}$, $x_{5,ez,n+1}$ are found, the matrix ***A*** and ***b*** are completely known at (n) time. Therefore, the control variables $u_{1,n}$ and $u_{2,n}$ (***u*** vector) can be calculated solving the system (10) by the least squares method:

$$(A^T A)u = A^T b \Rightarrow u = (A^T A)^{-1} A^T b \quad (13)$$

The solution allows finding the control actions ($u_{1,n}$ and $u_{2,n}$) to be applied at time n to follow the desired trajectories with a minimal error.

Remark 3. The following constraints on the control variables are considered (Balsa-Canto et al., 2000): $0.0 \leq u_1 \leq 1.0$ and $0.0 \leq u_2 \leq 1.0$.

Remark 4. In Eq. (8) note that:

- If $k_i = 0$, the reference trajectory is reached in only one step.

Remark 5. The parameters k_i, $i = \{1,2,3,4,5\}$, satisfied $0 < k_i < 1$, which allows the tracking error tends to zero.

Remark 6. The tracking error is the value of the difference between the reference and real trajectory, and is calculated as:

$$\|e_n\| = \sqrt{e_{1,n}{}^2 + e_{2,n}{}^2 + e_{4,n}{}^2} \; ; \; E = T_0 \sum_n \|e_n\| \qquad (14)$$

$$e_{1,n} = \frac{x_{1,ref,n} - x_{1,n}}{x_{1,\max}} \qquad (15)$$

$$e_{2,n} = \frac{x_{2,ref,n} - x_{2,n}}{x_{2,\max}} \qquad (16)$$

$$e_{4,n} = \frac{x_{4,ref,n} - x_{4,n}}{x_{4,\max}} \qquad (17)$$

Where: $x_{1,max} = 1{,}9\ L$, $x_{2,max} = 13{,}92\ g/L$, and $x_{4,max} = 3{,}1\ g/L$

Note that the tracking error is dimensionless.

Theorem 1. If the system behavior is ruled by (10) and the controller is designed by (13), then, $e_{i,n} \to 0$, $n \to \infty$, when profile tracking problems are considered. The proof of this theorem is not shown because space reasons.

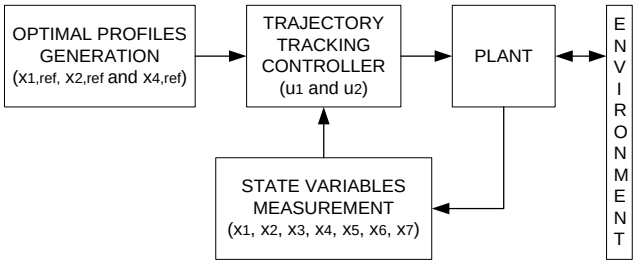


Figure 1. Architecture of the trajectory tracking controller.

Figure 1 shows the architecture of the control system proposed herein. In this work, the optimal profiles are taken from literature (Balsa-Canto et al., 2000), the focus is in the trajectory tracking of such profiles.

4. RESULTS AND DISCUSSION

In order to evaluate the performance of the controller, various simulation tests employing Matlab® were carried out:

- A randomized algorithm to synthetize the optimal controller parameters (k_i).
- Simulation under normal operating conditions using the optimal values of the controller parameters found in previous section.
- Simulation considering parametric uncertainty.

4.1. Optimal controller parameters.

The aim in this subsection is to find the values of the controller parameters, for which the tracking error is minimal (the bioreactor behavior directly depends on the adjustment of parameters k_i). To achieve the target, a Montecarlo experiment is performed.

In the field of systems and control, Montecarlo methods have been found useful especially for problems related to robustness of uncertain systems (Tempo and Ishii, 2007).

In this work, the Montecarlo experiment consisted on randomize the controller parameters and then simulate the process, this is repeated a large number of trials (N) and the tracking error is calculated in every one.

Now, considering by definition that Montecarlo randomized algorithm (MCRA) is a randomized algorithm that may produce an incorrect result, but the probability of such an incorrect result is bounded (Motwani and Raghavan, 1995), the number of simulations necessary to ensure a certain degree of confidence and accuracy (confidence boundaries) is achieved using the following expression (Tempo and Ishii, 2007):

$$N \geq \left\lceil \frac{\log \frac{1}{\delta}}{\log \frac{1}{1-\varepsilon}} \right\rceil \qquad (18)$$

Where δ= confidence and ε= accuracy.

It is fixed $\delta = 0.01$ and $\varepsilon = 0.005$. Therefore, $N \geq 920$.

The initial conditions used to simulate the system are shown in Table 1.

Feeding concentrations and parameters values can be seen in Table 2.

Table 1 Initial conditions for the state variables [*g/L*].

$x_{1,0}$	$x_{2,0}$	$x_{3,0}$	$x_{4,0}$	$x_{5,0}$	$x_{6,n}$	$x_{7,0}$
1.0	*0.1*	*40.0*	*0.01*	*0.01*	*1.0*	*0.01*

Table 2 Feeding concentrations and parameters (Tholudur and Ramirez, 1997).

N (g/L)	***I(g/L)***	***Y***	***p (h^{-1})***
40	*100*	*0.51*	*0.09*

The final time for the process is $T_f = 10\ h$ and the sample time for simulations is $T_0 = 0.1\ h$

The k_i values found for the minimum tracking error after 1000 simulations (total simulation time: 75 min) are presented in Table 3.

Table 3 Optimal k values after Montecarlo experiment.

k_1	k_2	k_3	k_4	k_5
0.9501	*0.5463*	*0.1459*	*0.1563*	*0.2624*
E=0.0014 – Iteration Nº 202				

4.2. Normal conditions operation.

This section shows the results of the simulation of the closed loop control using the controller proposed in this work without environmental disturbances. The optimal controller parameters achieved in the previous subsection and the state variables initial values are used. The evolution of the controlled system can be seen in the following figures.

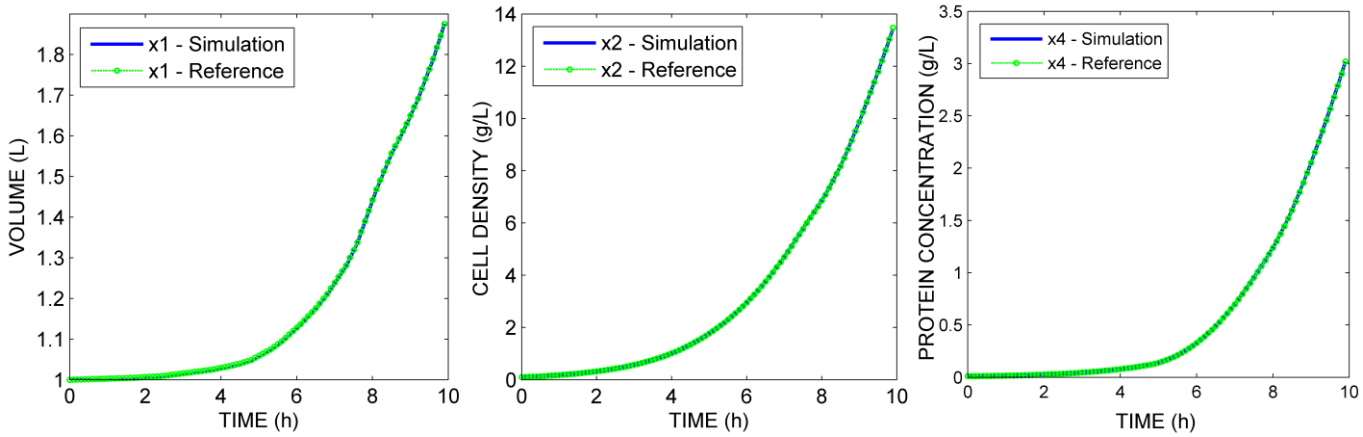


Figure 2. Tracking optimal profiles of desired variables (x_1, x_2, and x_4) in normal operation conditions.

A very good controller performance can be observed, the optimal desired profiles are successfully tracking as can be seen in Fig. 2.

Note that the tracking error defined by (14), remains low and acceptably bounded as shows Fig. 3.

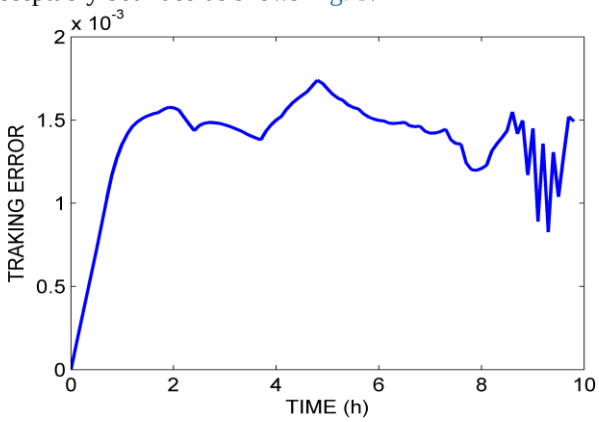


Figure 3. Tracking error *(e)* in normal operation conditions.

It is noteworthy that not only the desired variables follow the optimal profiles, but also the other variables track their respective reference profiles (it is not show here for space reasons).

4.3. Parametric uncertainty.

In this subsection, the MC Method is applied to make an analysis of the system in case of appearing modeling errors. For this, a determined error is introduced in the model parameters (15% above and below their nominal values) and perform 500 simulations (N = 500). In each simulation the parameters are chosen in a random way by MC sampling experiment (Auat Cheein et al., 2013). The parameters under disturbance are the concentration of nutrient feed stream ***N***, the concentration of inducer in the inducer feed stream ***I***, the Monod-type constant ***p*** and the growth yield coefficient ***Y***. The initial conditions and the sampling time are the same used in the previous subsection.

The number of simulations taken to carry out this test is considering the confidence $\delta = 0.01$ and accuracy $\varepsilon = 0.01$. Therefore, replacing in (18), $N \geq 500$.

Figures 4 shows the behavior of the system with the controller proposed in this work, which presents a good response against parametric uncertainty.

It is noteworthy that the controller design is focused in the tracking optimal profile of the desired variables, not of control variables. This is one of the advantages, since it allows control of the system even under disturbances.

The simulation tests carried out along this work show a very good performance of the control law developed.

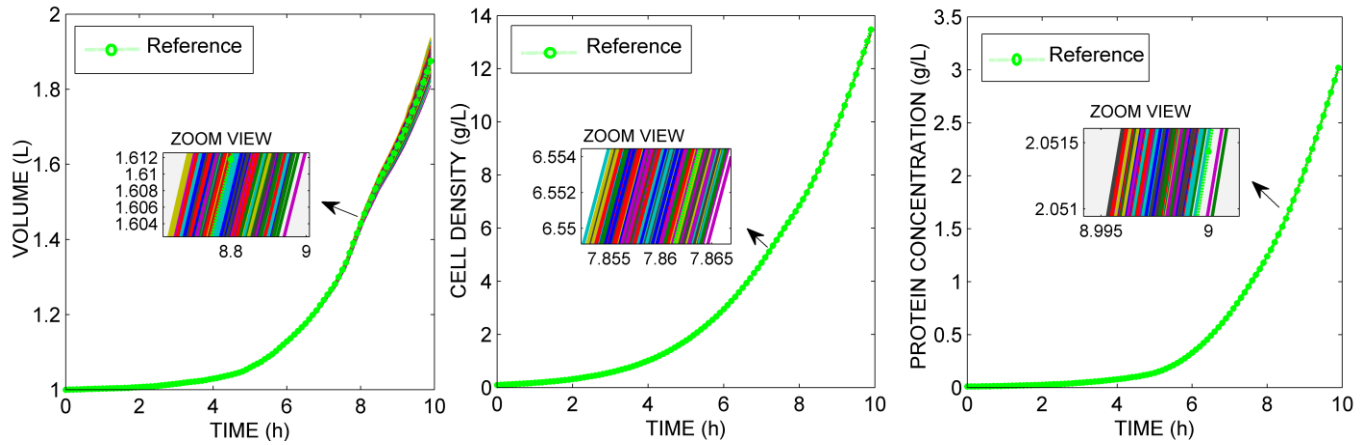


Figure 4. Tracking optimal profiles (desired variables x_1, x_2, and x_4) under parametric uncertainty (±15%, 500 simulations).

5. CONCLUSIONS

The controller design of an important biological system consisting on seven differential equations with three desired variables and two control actions was presented in this work. The advantages of the methodology employed is its simplicity and accuracy since reduces the controller design to a resolution of a linear equations system for the calculation of the control actions, which achieve the tracking optimal trajectories of desired variables with a minimal tracking error, even under parametric uncertainties. This methodology can be implemented only with basic knowledge of linear algebra. The optimal controller parameters were successfully found through a Montecarlo experiment. Moreover, through closed-loop simulation tests, this proposed control structure was shown to be simple and efficient, even considering the mismatches in the model parameters.

ACKNOWLEDGMENT

This work was funded by the *Consejo Nacional de Investigaciones Científicas y Técnicas* (CONICET- *National Council for Scientific Research*) and *Instituto de Ingeniería Química, Facultad de Ingeniería - Universidad Nacional de San Juan*, Argentina.

REFERENCES

ALFORD, J. S. 2006. Bioprocess control: Advances and challenges. *Computers & Chemical Engineering,* 30**,** 1464-1475.

ASHOORI, A., MOSHIRI, B., KHAKI-SEDIGH, A. & BAKHTIARI, M. R. 2009. Optimal control of a nonlinear fed-batch fermentation process using model predictive approach. *Journal of Process Control,* 19**,** 1162-1173.

AUAT CHEEIN, F. A., PEREIRA, F. M. L., DI SCIASCIO, F. & CARELLI, R. 2013. Autonomous Simultaneous Localization and Mapping driven by Monte Carlo uncertainty maps-based navigation. *The Knowledge Engineering Review,* 28**,** 35-57.

BALSA-CANTO, E., BANGA, J. R., ALONSO, A. A. & VASSILIADIS, V. S. 2000. Efficient optimal control of bioprocesses using second-order information. *Industrial and Engineering Chemistry Research,* 39**,** 4287-4295.

BAYEN, T. & MAIRET, F. 2013. Minimal time control of fed-batch bioreactor with product inhibition. *Bioprocess and Biosystems Engineering,* 36**,** 1485-1496.

BERBER, R. 1996. Control of Batch Reactors-A Review (Reprinted from Methods of Model Based Process Control, 1995). *Chemical engineering research & design,* 74**,** 3-20.

CARRASCO, E. F. & BANGA, J. R. 1997. Dynamic optimization of batch reactors using adaptive stochastic algorithms. *Industrial & engineering chemistry research,* 36**,** 2252-2261.

COSENZA, B. & GALLUZZO, M. 2012. Nonlinear fuzzy control of a fed-batch reactor for penicillin production. *Computers & Chemical Engineering,* 36**,** 273-281.

CRAVEN, S., WHELAN, J. & GLENNON, B. 2014. Glucose concentration control of a fed-batch mammalian cell bioprocess using a nonlinear model predictive controller. *Journal of Process Control,* 24**,** 344-357.

CHANG, D. M. 2003. The Snowball Effect in Fed-Batch Bioreactions. *Biotechnology progress,* 19**,** 1064-1070.

CHUNG, Y.-C., CHIEN, I.-L. & CHANG, D.-M. 2006. Multiple-model control strategy for a fed-batch high cell-density culture processing. *Journal of Process Control,* 16**,** 9-26.

DE BATTISTA, H., PICÓ, J. & PICÓ-MARCO, E. 2012. Nonlinear PI control of fed-batch processes for growth rate regulation. *Journal of Process Control,* 22**,** 789-797.

JOHNSON, A. 1987. The control of fed-batch fermentation processes—a survey. *Automatica,* 23**,** 691-705.

LEE, J., LEE, S. Y., PARK, S. & MIDDELBERG, A. P. J. 1999. Control of fed-batch fermentations. *Biotechnology Advances,* 17**,** 29-48.

LEE, J. & RAMIREZ, W. F. 1992. Mathematical modeling of induced foreign protein production by recombinant bacteria. *Biotechnology and bioengineering,* 39**,** 635-646.

LEE, J. & RAMIREZ, W. F. 1994. Optimal fed-batch control of induced foreign protein production by recombinant bacteria. *AIChE Journal,* 40**,** 899-907.

MOTWANI, R. & RAGHAVAN, P. 1995. Randomized algorithms (cambridge international series on parallel computation). Cambridge University Press.

RANI, K. Y. & RAO, V. R. 1999. Control of fermenters–a review. *Bioprocess Engineering,* 21**,** 77-88.

RENARD, F. & WOUWER, A. V. 2008. Robust adaptive control of yeast fed-batch cultures. *Computers & Chemical Engineering,* 32**,** 1238-1248.

RENARD, F., WOUWER, A. V., VALENTINOTTI, S. & DUMUR, D. 2006. A practical robust control scheme for yeast fed-batch cultures–an experimental validation. *Journal of Process Control,* 16**,** 855-864.

ROCHA, M., MENDES, R., ROCHA, O., ROCHA, I. & FERREIRA, E. C. 2014. Optimization of fed-batch fermentation processes with bio-inspired algorithms. *Expert Systems with Applications,* 41**,** 2186-2195.

RONEN, M., SHABTAI, Y. & GUTERMAN, H. 2002. Optimization of feeding profile for a fed-batch bioreactor by an evolutionary algorithm. *Journal of biotechnology,* 97**,** 253-263.

SAINT-DONAT, J., BHAT, N. & MCAVOY, T. J. 1991. Neural net based model predictive control. *International Journal of Control,* 54**,** 1453-1468.

SANTOS, L. O., DEWASME, L., COUTINHO, D. & WOUWER, A. V. 2012. Nonlinear model predictive control of fed-batch cultures of micro-organisms exhibiting overflow metabolism: assessment and robustness. *Computers & Chemical Engineering,* 39**,** 143-151.

SARKAR, D. & MODAK, J. M. 2003. Optimisation of fed-batch bioreactors using genetic algorithms. *Chemical Engineering Science,* 58**,** 2283-2296.

SARKAR, D. & MODAK, J. M. 2004. Optimization of fed-batch bioreactors using genetic algorithm: multiple control variables. *Computers & Chemical Engineering,* 28**,** 789-798.

SONI, A. S. & PARKER, R. S. 2004. Closed-loop control of fed-batch bioreactors: A shrinking-horizon approach. *Industrial & engineering chemistry research,* 43**,** 3381-3393.

TEMPO, R. & ISHII, H. 2007. Monte Carlo and Las Vegas Randomized Algorithms for Systems and Control: An Introduction. *European Journal of Control,* 13**,** 189-203.

THOLUDUR, A. & RAMIREZ, W. F. 1996. Optimization of Fed-Batch Bioreactors Using Neural Network Parameter Function Models. *Biotechnology Progress,* 12**,** 302-309.

THOLUDUR, A. & RAMIREZ, W. F. 1997. Obtaining smoother singular arc policies using a modified iterative dynamic programming algorithm. *International Journal of Control,* 68**,** 1115-1128.

IMPACT AND ADVANCES OF AUTOMATIC CONTROL IN LATINAMERICA

EDITORS

Oscar Camacho • Danilo Chávez • Gerardo Espinosa-Pérez • O. Lucia Quintero M • Gustavo Scaglia

REVIEWERS

Adriana Natacha Amicarelli	INAUT - Argentina
Andrea Angel Zea	EMGESA - Colombia
Antonio Morán	PUCP - Perú
Carlos Alberto Cadavid Moreno	Universidad EAFIT - Colombia
Carlos Andrés Sánchez López	CELSIA - Colombia
Ceso de la Cruz Casaño	Universidad Continental – Perú
César A Uribe	University of Illinois at Urbana-Champaign
Daniel Sierra-Sosa	Universidad EAFIT – Colombia
David Ortiz	Universidad EAFIT – Colombia
Emanuel Serrano	IIQ- UNSJ – Argentina
Fabian Leonardo Jaramillo Palacios	Universidad de Cuenca – Ecuador
Flavio Roberti	INAUT – Argentina
Gustavo Pérez	PUCP – Perú
Héctor Antonio Botero Castro	UNAL – Colombia
Jan. H. van Schuppen	TUDelft – The Netherlands
Javier Sotomayor M	PUCP – Perú
Jesús Antonio Hernandez Riveros	UNAL – Colombia
Juan Fernando García Tirado	ITM – Colombia
Juan Carlos Rivera	Universidad EAFIT – Colombia
Juan Diego Sánchez Torres	ITESO – México
Juan Guillermo Paniagua Castrillón	ITM – Colombia
Juan Marcos Toibero	INAUT – Argentina
Manuel Betancur Betancur	UPB – Colombia
Maria Gulnara Baldoquín de la Peña	Universidad EAFIT – Colombia
Mario Fernández-Fernández	Universidad de Talca – Chile
Natalia Martina López	GATEME – UNSJ- Argentina
Pablo Santiago Rivadeneira	UNAL – Colombia
Ricardo Carelli	INAUT – Argentina
Sergio Ponce	UTN - Argentina

From the Conference proceedings of the
XVII LATIN AMERICAN CONFERENCE IN AUTOMATIC CONTROL
Universidad EAFIT
School of Sciences
2016