THE 10TH INTERNATIONAL CONFERENCE ON INTEGRATED MODELING AND ANALYSIS IN APPLIED CONTROL AND AUTOMATION

SEPTEMBER 18 – 20, 2017
BARCELONA, SPAIN

IMAACA

EDITED BY
AGOSTINO BRUZZONE
GENEVIÈVE DAUPHIN-TANGUY
SERGIO JUNCO

PRINTED IN RENDE (CS), ITALY, SEPTEMBER 2017

ISBN 978-88-97999-96-6 (Paperback)
ISBN 978-88-97999-88-1 (PDF)

ISBN 978-88-97999-96-6 (Paperback)
ISBN 978-88-97999-88-1 (PDF)

The 10th International Conference On Integrated Modeling And Analysis In Applied Control And Automation, IMAACA 2017

September 18 – 20, 2017

Barcelona, Spain

Organized by

DIME – University of Genoa

Liophant Simulation

Simulation Team

IMCS – International Mediterranean & Latin American Council of Simulation

DIMEG, University of Calabria

MSC-LES, Modeling & Simulation Center, Laboratory of Enterprise Solutions

Autonomous University of Barcelona

Modeling and Simulation Center of Excellence (MSCOE)

Latvian Simulation Center – Riga Technical University

Logisim

LSIS – Laboratoire des Sciences de l'Information et des Systemes

MIMOS – Movimento Italiano Modellazione e Simulazione

MITIM Perugia Center – University of Perugia

Brasilian Simulation Center, LAMCE-COPPE-UFRJ

MITIM - McLeod Institute of Technology and Interoperable Modeling and Simulation – Genoa Center

M&SNet - McLeod Modeling and Simulation Network

Latvian Simulation Society

Ecole Superieure d'Ingenierie en Sciences Appliquees

Facultad de Ciencias Exactas. Ingegneria y Agrimensura

University of La Laguna

CIFASIS: CONICET-UNR-UPCAM

INSTICC - Institute for Systems and Technologies of Information, Control and Communication

National Russian Simulation Society

CEA - IFAC

University of Bordeaux

University of Cyprus

Dutch Benelux Simulation Society

I3M 2017 Industrial Sponsors

CAL-TEK SRL

LIOTECH LTD

MAST SRL

SIM-4-FUTURE

I3M 2017 MEDIA PARTNERS

INDERSCIENCE PUBLISHERS

INDERSCIENCE PUBLISHERS – INTERNATIONAL JOURNAL OF SIMULATION AND PROCESS MODELING

INDERSCIENCE PUBLISHERS – INTERNATIONAL JOURNAL OF OIL, GAS AND COAL TECHNOLOGY

INDERSCIENCE PUBLISHERS – INTERNATIONAL JOURNAL OF SERVICE AND COMPUTING ORIENTED MANUFACTURING

IGI GLOBAL
DISSEMINATOR OF KNOWLEDGE

IGI GLOBAL – INTERNATIONAL JOURNAL OF PRIVACY AND HEALTH INFORMATION MANAGEMENT

HalldaleGroup

mst MILITARY SIMULATION & TRAINING MAGAZINE

HALLDALE MEDIA GROUP: MILITARY SIMULATION AND TRAINING MAGAZINE

MedicalTraining magazine

HALLDALE MEDIA GROUP: THE JOURNAL FOR HEALTHCARE EDUCATION, SIMULATION AND TRAINING

SAGE

SAGE
SIMULATION TRANSACTION OF SCS

DE G DE GRUYTER

DE GRUYTER
INTERNATIONAL JOURNAL OF FOOD ENGINEERING

EDITORS

AGOSTINO BRUZZONE
MITIM-DIME, UNIVERSITY OF GENOA, ITALY
agostino@itim.unige.it

GENEVIÈVE DAUPHIN-TANGUY
ECOLE CENTRALE DE LILLE, FRANCE
genevieve.dauphin-tanguy@ec-lille.fr

SERGIO JUNCO
UNIVERSIDAD NACIONAL DE ROSARIO, ARGENTINA
sjunco@fceia.unr.edu.ar

THE INTERNATIONAL MULTIDISCIPLINARY MODELING AND SIMULATION MULTICONFERENCE, I3M 2017

GENERAL CO-CHAIRS

AGOSTINO BRUZZONE, *MITIM DIME, UNIVERSITY OF GENOA, ITALY*
MIQUEL ANGEL PIERA, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*

PROGRAM CO-CHAIRS

FRANCESCO LONGO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
YURY MERKURYEV, *RIGA TECHNICAL UNIVERSITY, LATVIA*

THE 10TH INTERNATIONAL CONFERENCE ON INTEGRATED MODELING AND ANALYSIS IN APPLIED CONTROL AND AUTOMATION, IMAACA 2017

CONFERENCE CHAIRS

SERGIO JUNCO, *UNIVERSIDAD NACIONAL DE ROSARIO, ARGENTINA*
GENEVIÈVE DAUPHIN-TANGUY, *ECOLE CENTRALE DE LILLE, FRANCE*

IMAACA 2017 International Program Committee

Jorge Baliño, Univ. of São Paulo, Brazil
Wolfgang Borutzky, BRS-Univ. Applied Sciences, Sankt Augustin, Germany
Agostino Bruzzone, University of Genoa, Italy
Nicolaï Christov, LAGIS, UST Lille, France
Nicolaï Christov, LAGIS, UST Lille, France
Geneviève Dauphin-Tanguy, École Centrale de Lille, France
Jean-Yves Dieulot, Polytech'Lille, France
Alejandro Donaire, The University of Newcastle, Australia
Tulga Ersal, University of Michigan, USA
Jose Luis Godoy, INTEC, UNL, Argentina
Hussein Ibrahim, ITMI, Quebec, Canada
Laurentiu Hetel, CNRS-EC-Lille, France
Adrian Ilinca, Université du Québec à Rimouski, Canada
Mayank Shekhar Jha, C. Ader Institute, France
Sergio Junco, Universidad Nacional de Rosario, Argentina
Fabrizio Leonardi, FEI, São Bernardo do Campo, Brazil
Francesco Longo, University of Calabria, Italy
Loucas S. Louca, University of Cyprus, School of Engineering, Cyprus
Marina Massei, Liophant Simulation, Italy
Matías Nacusse, FCEIA, UNR, Rosario, Argentina
letizia nicoletti, University of Calabria, Italy
Norberto Nigro, CIMEC, UNL, S. Fe, Argentina
Rachid Outbib, LSIS, Marseille, France
Pushparaj Mani Pathak, IIT, Roorkee, India
Ricardo Pérez Correa, PUC, Chile
Drishtysingh Ramdenee, ITMI, Québec, Canada
Xavier Roboam, INP Toulouse, France
Mónica Romero, FCEIA, UNR, Rosario, Argentina
Christophe Sueur, Centrale Lille, France
Vladimir Turetsky, ORT Braude College, Israel
Costas Tzafestas, NTUA, Greece
Daniel Viassolo, Schlumberger, Houston, TX, USA
Rafic Younes, The Lebanese University, Beirut, Lebanon
Aníbal Zanini, Facultad de Ingeniería, Universidad de Buenos Aires, Argentina

CHAIRS' MESSAGE

WELCOME TO IMAACA 2017!

On behalf of the International Program Committee we are extremely pleased to welcome all the authors, representatives and attendees to IMAACA 2017, the 10th International Conference on Integrated Modeling and Analysis in Applied Control and Automation!

We are happy to be once again part of I3M, the International Multidisciplinary Modeling and Simulation Multiconference, which this year celebrates its 14th edition. On the one hand, I3M provides a superb context to all its member conferences, offering a plentiful of points of view and methodological approaches to modeling and simulation, as well as a great variety of application domains. On the other hand, as always, we benefit from the wonderful organization of the technical and social programs provided by the I3M Organization Committee and the associated team. To all of them our warmest thanks! We also express our gratitude to the local Organizing Committee for providing us with such a pleasant and exchange-inviting working atmosphere in the culturally and historically rich beautiful Barcelona.

As chairs of IMAACA we know that most of the success of the conference also depends on the input of its IPC members. We thank all of them for their support through the in-depth and duly reviewing of the papers submitted to the conference. And of course, last but not least, our deepest thanks go to all the authors for their contributions and attendance to the conference. Finally, let us wish you a fully enjoyable conference in both its technical and social aspects.

Sergio Junco,
Universidad Nacional de Rosario
Argentina

Geneviève Dauphin-Tanguy,
Ecole Centrale de Lille
France

ACKNOWLEDGEMENTS

The IMAACA 2017 International Program Committee (IPC) has selected the papers for the Conference among many submissions; therefore, based on this effort, a very successful event is expected. The IMAACA 2017 IPC would like to thank all the authors as well as the reviewers for their invaluable work.
A special thank goes to Prof. Miquel Angel Piera from Autonomous University of Barcelona, as Local Organizer and to all the organizations, institutions and societies that have supported and technically sponsored the event.

I3M 2017 INTERNAL STAFF

MATTEO AGRESTA, *SIMULATION TEAM, ITALY*
MARGARITA BAGAMANOVA, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
LUIGI BRUNO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
AGOSTINO G. BRUZZONE, *DIME, UNIVERSITY OF GENOA, ITALY*
ALESSANDRO CHIURCO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
RICCARDO DI MATTEO, *SIMULATION TEAM, ITALY*
CATERINA FUSTO, *CAL-TEK SRL, ITALY*
THIMJO KOÇA, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
FRANCESCO LONGO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
GIANLUCA MAGLIONE, *SIMULATION TEAM, ITALY*
MARINA MASSEI, *DIME, UNIVERSITY OF GENOA, ITALY*
ROMUALDO MORENO, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
MERCEDES NARCISO, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
LETIZIA NICOLETTI, *CAL-TEK SRL, ITALY*
ANTONIO PADOVANO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
MIQUEL ANGEL PIERA, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
MARKO RADANOVIC, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
JUAN JOSE RAMOS, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
CATALDO RUSSO, *CAL-TEK SRL, ITALY*
NINA SCHEFERS, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*
MARCO VETRANO, *CAL-TEK SRL, ITALY*

This International Workshop is part of the I3M Multiconference: the Congress leading **Simulation around the World and Along the Years**

INDEX

Unknown Input Observer for MIMO Systems with Stability

Joel A. Gonzalez & Christophe Sueur

Ecole Centrale de Lille, CRIStAL UMR CNRS 9189, CS 20048
59651, Villeneuve d'Ascq Cedex France

joelabrahamgv@gmail.com, christophe.sueur@centralelille.fr

Abstract

This paper presents a solution for the state and unknown input estimator of linear MIMO systems with a structural approach. Compared to the classic Input-Output Decoupling problem, a systematic procedure is initially proposed for the analysis stage by analysing the finite and infinite structures of the modelled system from a structural approach. Afterwards, the observer is directly implemented from the original model at the synthesis stage in a graphical approach with some added terms, and can be represented by a Bond Graph model. The observer is tested by a particular and illustrative example which considers a real torsion-bar model.

Keywords: Unknown Input Observer, Bond Graph, Structural Analysis, MIMO Systems

1 Introduction

The unknown input estimation and state observability problem (UIO) is a well known problem. Different approaches give solvability conditions and constructive solutions for this problem.

At the analysis stage, before design, most of the approaches require the analysis of the structural invariants of the model. The knowledge of zeros (finite structure) is an important issue because these zeros are directly related to stability conditions of the observer ((Hautus 1983)) and of the controlled system. The infinite structure of the model is related to solvability conditions (see Appendix A).

For LTI models, constructive solutions with reduced order observers are first proposed with the geometric approach, see (Bhattacharyya 1978), (Basile and Marro 1973) or based on generalized inverse matrices like in (Miller and Mukunden 1982) and (Hou and Muller 1992). Full order observers are then proposed in a similar way (based on generalized inverse matrices), see (Darouach 2009) for some works related with this issue. Other approaches based on canonical forms, the algebraic approach or sliding mode observers are not recalled here.

This paper proposes an extension of some works dedicated to the UIO problem when the model contains some non strictly stable invariant zeros. The contribution consists on the definition of a new estimation of the disturbance variables which takes into account some integrals of the measured variables in order to augment the number of assigned poles.

2 Unknown Input Observer

Consider a linear perturbed system described by (1), where $x \in \mathbb{R}^n$ is the state vector, $y \in \mathbb{R}^p$ is the vector of measurable variables. Vector $u \in \mathbb{R}^m$ represents the known input variables, whereas $d(t) \in \mathbb{R}^q$ is the vector which represents the unknown input variables. A, B, F, C are known constant matrices of appropriate dimensions.

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + Fd(t) \\ y(t) = Cx(t) \end{cases} \tag{1}$$

Generally, the state vector $x(t)$ cannot be entirely measured and the system is often subject to unknown inputs $d(t)$ (disturbance or failure...) which must be estimated.

2.1 MIMO Systems

In this section, the UIO estimation for MIMO systems without null invariant zero is recalled. For some works related with the design of UIO for linear Bond Graph models, see (Yang et al. 2013), (Tarasov et al. 2013).

The UIO problem is recalled for the multi-variable case with two unknown input variables ($q = 2$) and two measured outputs variables ($p = 2$). It can easily be extended to any square model with $p = q$. It is supposed that system $\Sigma(C, A, F)$ is controllable, observable and invertible. The equation (1) can be written as (2).

$$\left\{\begin{array}{l} x(t) = A^{-1}\dot{x}(t) - A^{-1}Bu(t) - A^{-1}Fd(t) \\ y(t) = CA^{-1}\dot{x}(t) - CA^{-1}Bu(t) - CA^{-1}Fd(t) \end{array}\right. \quad (2)$$

The matrix $\Omega_d = CA^{-1}F$ is invertible if model $\Sigma(C,A,F)$ has no null invariant zeros. Matrix Ω_d is similar to decoupling matrix Ω in control theory. In the classical input-output decoupling problem, the decoupling matrix Ω defined in equation (8) is used with matrix B instead of matrix F (with the control input variables). From (2), the disturbance vector $d(t)$ and its estimation $\hat{d}(t)$ are written in equation (3), and the disturbance equation error in equation (4). The estimation of the state vector is written in equation (5).

$$\left\{\begin{array}{l} d(t) = -\Omega_d^{-1}[y(t) - CA^{-1}\dot{x}(t) + CA^{-1}Bu(t)] \\ \\ \hat{d}(t) = -\Omega_d^{-1}[y(t) - CA^{-1}\dot{\hat{x}}(t) + CA^{-1}Bu(t)] \end{array}\right. \quad (3)$$

$$d(t) - \hat{d}(t) = \Omega_d^{-1}CA^{-1}(\dot{x}(t) - \dot{\hat{x}}(t)) \quad (4)$$

$$\begin{aligned} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + F\hat{d}(t) - \cdots \\ -AK\left[\begin{array}{c} y_1^{(n_1)}(t) - \hat{y}_1^{(n_1)}(t) \\ y_2^{(n_2)}(t) - \hat{y}_2^{(n_2)}(t) \end{array}\right] \end{aligned} \quad (5)$$

Ω_d is defined in case of the Bond Graph representation with a derivative causality assignment (BGD) and related to the infinite structure of the BGD.

Consider vector $e(t)$ defined as $e(t) = x(t) - \hat{x}(t)$. It has been proved in (Tarasov et al. 2013) that vector $e(t)$ verifies equation (6) with matrix N_{CL} defined in (7). In that case, the state estimation error doesn't depend on the disturbance variable.

$$e(t) = N_{CL}\dot{e}(t) \quad (6)$$

$$\left\{\begin{array}{l} N_{OL} = A^{-1} - A^{-1}F\Omega_d^{-1}CA^{-1} \\ \\ N_{CL} = A^{-1} - A^{-1}F\Omega_d^{-1}CA^{-1} - K\begin{bmatrix} C_1A^{n_1-1} \\ C_2A^{n_2-1} \end{bmatrix} \end{array}\right. \quad (7)$$

If matrix N_{CL} is invertible, a classical pole placement is studied with matrix K used for pole placement. A necessary condition for the existence of the state estimator is proposed in Proposition 1.

Suppose that $\{n_1, n_2\}$ and $\{n_1', n_2'\}$ are the set of row infinite zero orders and global infinite zero orders respectively, of system $\Sigma(C,A,F)$ (see Appendix A). Matrix Ω is necessary in this estimation problem.

$$\Omega = \left[\begin{array}{c} C_1A^{n_1-1}F \\ C_2A^{n_2-1}F \end{array}\right] \quad (8)$$

Proposition 1 *A necessary condition for matrix N_{CL} to be invertible is that matrix Ω is invertible.*

Proof 1 *See appendix B*

It has been proved in (Tarasov et al. 2013) that the eigenvalues of matrix N_{OL} defined in (7) are the inverse of the invariant zeros of system $\Sigma(C,A,F)$ $(n - (n_1 + n_2))$ modes) plus $n_1 + n_2$ eigenvalues equal to 0. The null eigenvalues can be assigned to new values and the invariant zeros are the fixed poles as recalled in the two following propositions.

Proposition 2 *((Tarasov et al. 2013))In matrix N_{CL} defined in (7), $n_1 + n_2$ poles can be chosen with matrix K.*

Proposition 3 *((Tarasov et al. 2013)) The fixed poles of the estimation equation error defined in (6) are the invariant zeros of system $\Sigma(C,A,F)$.*

The proposed approach can then be applied for systems with *strictly stable invariant zeros*.

If the state equation (1) is written from a Bond Graph model, it is possible to design a Bond Graph model for the state estimation defined in (5) because the equation (5) is very close to the initial state equation. Some signal bonds must be added for the disturbance equation defined in (3). The structure of the observer is proposed in Fig.(1), where BG_{sys} is Bond Graph model of the system and BG_{obs} is for the observer Bond Graph model. A simplified block diagram in Fig.1 represents the structure of simulation to estimate unknown variables. As written in equation (5), output variables must be derived. Some remarks about this feature are made in the following.

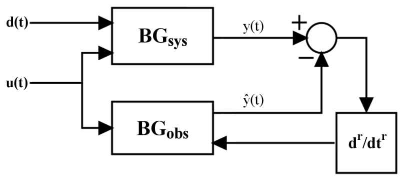


Figure 1: Structure of the simulation to estimate variables.

2.2 Simplified Case with Null Invariant Zeros

In this section, an extension of UIO observer is proposed when the model contains null invariant zeros. In this paper, we consider the case with one null invariant zero, it can be easily extended to more general situations.

The equation (1) can be written as (9)

$$\left\{\begin{array}{l} x(t) = A^{-1}\dot{x}(t) - A^{-1}Bu(t) - A^{-1}Fd(t) \\ y_1(t) = C_1[A^{-1}\dot{x}(t) - A^{-1}Bu(t) - A^{-1}Fd(t)] \\ y_2(t) = C_2[A^{-1}\dot{x}(t) - A^{-1}Bu(t) - A^{-1}Fd(t)] \end{array}\right. \quad (9)$$

Consider, without restriction, that $C_1A^{-1}F = 0$. In that case, in the BGD, the causal path length between the output detector and the two disturbance inputs is at least equal to 1 (it is supposed to be equal to 1 in order to simplify the theoretical development). In (9), the mathematical expression of $y_1(t)$ and of its primitive is then

$$\left\{\begin{array}{l} y_1(t) = C_1A^{-1}\dot{x}(t) - C_1A^{-1}Bu(t) \\ \int y_1(t)\mathrm{d}t = C_1A^{-1}x(t) - C_1A^{-1}B\int u(t)\mathrm{d}t \end{array}\right. \quad (10)$$

Thus

$$\int y_1(t)\mathrm{d}t = C_1A^{-2}\dot{x}(t) - C_1A^{-2}Bu(t) - \cdots \\ - C_1A^{-1}B\int u(t)\mathrm{d}t - C_1A^{-2}Fd(t) \quad (11)$$

If model $\Sigma(C,A,F)$ has only one null invariant zero, matrix $C_1A^{-2}F \neq 0$ and matrix $\Omega_d = [(C_1A^{-2}F)^t, (C_2A^{-1}F)^t]^t$ is invertible. A new expression of vector $d(t)$ can be written, as well for $\hat{d}(t)$ from equation (11) in the same manner as in the classical case and the error equation is written in (14).

$$d(t) = -\Omega_d^{-1}\left[\begin{array}{l} \int y_1(t)\mathrm{d}t - C_1A^{-2}\dot{x}(t) + \gamma(u) \\ y_2(t) - C_2A^{-1}(\dot{x}(t) - Bu(t)) \end{array}\right] \quad (12)$$

$$\hat{d}(t) = -\Omega_d^{-1}\left[\begin{array}{l} \int y_1(t)\text{text}t - C_1A^{-2}\dot{\hat{x}}(t) + \gamma(u) \\ y_2(t) - C_2A^{-1}(\dot{\hat{x}}(t) - Bu(t)) \end{array}\right] \quad (13)$$

Where $\gamma(u) = C_1A^{-2}Bu(t) + C_1A^{-1}B\int u(t)\mathrm{d}t$, and then

$$d(t) - \hat{d}(t) = \Omega_d^{-1}\begin{bmatrix} C_1A^{-2} \\ C_2A^{-1} \end{bmatrix}(\dot{x}(t) - \dot{\hat{x}}(t)) \quad (14)$$

The estimation of the state vector $x(t)$ is the same as in equation (17) as well as for the state estimation error equation defined in (6). Nevertheless, expressions of matrices N_{OL} and N_{CL} have changed since the model has one null invariant zero and thus matrix N_{OL} contains $(n_1 + n_2 + 1)$ null eigenvalues. New expressions are written in (15)

$$\left\{\begin{array}{l} N_{OL} = A^{-1} - A^{-1}F\Omega_d^{-1} \quad \begin{bmatrix} C_1A^{-2} \\ C_2A^{-1} \end{bmatrix} \\ N_{CL} = A^{-1} - A^{-1}F\Omega_d^{-1} \quad \begin{bmatrix} C_1A^{-2} \\ C_2A^{-1} \end{bmatrix} \cdots \\ \qquad -K\begin{bmatrix} C_1A^{n_1-1} \\ C_2A^{n_2-1} \end{bmatrix} \end{array}\right. \quad (15)$$

A new proposition can be written.

Proposition 4 *The fixed poles of the estimation equation error defined in (15) are the strictly stable invariant zeros of system* $\Sigma(C,A,F)$. $(n - (n_1 + n_2 + 1))$ *fixed poles.*

Proof 2 *See appendix C*

As said before, this approach can be easily extended to MIMO models with several null invariant zeros. The idea consists in applying integration on the output variables. It is applied to the torsion-bar system.

2.3 Output Differentiation and noise

Numerical differentiation of measurable signals is a classical problem in signal processing and automation, and many problems have been solved by creating algorithms for approximation of derivatives. The numerical methods for approximation of derivatives of measurable signals can be used to obtain signals which are not known through measurements and reconstruct the missing system data.

There exist many ways in the literature to derivate signals. Some common features are the precision between derivative estimation and noise sensibility and perturbations. These noises or perturbations are the principal trouble for developing derivation algorithms. Most of them assume some features of signal derived and noise (perturbation) of this derivation.

Different approaches are used for different situations such as Linear Systems in (Luenberger 1971), (Carlsson, Ahlen, and Sternad 1991), (Diop et al. 1994), (Al-Alaoui 1993), (Dabroom and Khalil 1997), (Levant 1998), (Levant 2003), (Mehdi 2010). These approaches can be classified by two principal classes: *a*) Model Approach or *b*) Signal Approach. In this case of study, the simulations are performed using the *Matlab* and *20-SIM* softwares. Therefore, for a numerical differentiation of the output signal in simulations, will be used some own blocks of these softwares which are called direct derivative with a noise-filter inside. In future works will be used others approaches implemented over the real bar system.

3 Torsion-bar System

3.1 Experimental System description

The experimental setup in the Fig. 2 is the real torsion bar system with its main parts.

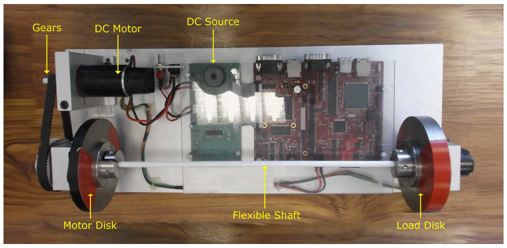


Figure 2: Real torsion bar system.

A functional schematic model of the torsion bar system is presented in Fig.3. According to Fig.3, the system consists of the following components: a DC Power Source, a classical DC Motor which is modelled by an electrical part (Inductance L_a and Resistance R_a) and a mechanical part (Inertia J_m is supposed negligible), a transmission element which transfer the rotation from the motor to the motor disk with a transmission ratio (k_{belt}), a first rotational disk (Motor Disk) with an inertial parameter J_1 and a friction coefficient R_1, a flexible shaft modelled as a spring-damper element (Spring c and Damper R_{shaft}), and a second rotational disk (Load Disk) with an inertial parameter J_2 and a friction coefficient R_2.

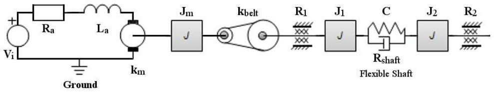


Figure 3: Schematic model of the real torsion bar system.

The simplified Bond Graph model of the system is shown in the Fig.4.

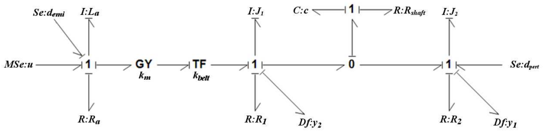


Figure 4: Simplified Bond Graph of the torsion bar system.

For the experimental system in Fig.4, the controlled input voltage is represented by a modulated effort source $MSe:u$. Moreover, y_1 and y_2 are speed rotational variables represented in the bond graph model by flow output detectors $Df:y_1$ and $Df:y_2$ respectively. These sensors are used to estimate the state variables and the unknown inputs $d(t)$ modeled by the source $Se:d_{pert}$ and d_{emi} modeled by the source $Se:d_{emi}$, which represents a torque applied to the first rotational disk(J_1) and a electromagnetic interference respectively. The numerical values for each element of the system are given in Table 1.

Table 1: Parameters for the experimental system

Element	Symbol	Value
Inductance	L_a	$0.34 \cdot 10^{-3}\ H$
Inertia of motor disk	J_1	$9.07 \cdot 10^{-4}\ kg \cdot m^2$
Inertia of load disk	J_2	$1.37 \cdot 10^{-3}\ kg \cdot m^2$
Spring compliance	C	$0.543\ N \cdot m/rad$
Resistance	R_a	$1.23\ \Omega$
Motor disk friction	R_1	$0.005\ N \cdot m \cdot s/rad$
Load disk friction	R_2	$25 \cdot 10^{-6}\ N \cdot m \cdot s/rad$
Damping spring	R_{shaft}	$5 \cdot 10^{-4}\ N \cdot s/rad$
Motor constant	k	$38.9 \cdot 10^{-3}\ N \cdot m/A$
Transmission ratio	k_{belt}	3.75

According to the Bond Graph model, a state space representation is performed as described in the form (1). The state vector $x = [x_1, x_2, x_3, x_4]^t$, is energy storage variables: $x_1 = q_c = q_{c_{shaft}}$ (angular displacement), $x_2 = p_{J_2}$, $x_3 = p_{J_1}$ (angular momentums), and $x_4 = p_{L_a}$ (flux linkage). The output matrix C can be written as $C = [{C_1}^t, {C_2}^t]^t$. The state equations are written as (16). The poles of the model (eigenvalues of matrix A) are equal to -3617.5, $-2.15 \pm 58j$, -2.2523.

$$\begin{cases} \dot{x}_1 = -\frac{1}{J_2}x_2 + \frac{1}{J_1}x_3 \\ \dot{x}_2 = \frac{1}{C}x_1 + a_{2,2}\,x_2 + \frac{R_{Shaft}}{J_1}x_3 + d_{pert} \\ \dot{x}_3 = -\frac{1}{C}x_1 + \frac{R_{Shaft}}{J_2}x_2 + a_{3,3}\,x_3 + \frac{k}{L_a.k_{belt}}x_4 \\ \dot{x}_4 = -\frac{k}{J_1 \cdot k_{belt}}x_3 - \frac{R_a}{L_a}x_4 + u + d_{emi} \\ \quad a_{2,2} = (-\frac{R_2}{J_2} - \frac{R_{Shaft}}{J_2})\ ;\ a_{3,3} = (-\frac{R_1}{J_1} - \frac{R_{Shaft}}{J_1}) \\ \quad y_1 = \frac{1}{J_2}x_2\ ;\ y_2 = \frac{1}{J_1}x_3 \end{cases} \tag{16}$$

3.2 Structural Analysis

From the Bond Graph model of Fig.4 the causal path between the output variable y_1 and the disturbance input d_{pert} is $Df: y_1 \rightarrow I: J_2 \rightarrow Se: d_{pert}$. the length of the causal path is equal to 1, then $n_1 = 1$. The causal path between the output y_2 and the disturbance input d_{emi} is $Df: y_2 \rightarrow I: J_1 \rightarrow TF: k_{belt} \rightarrow GY: k_m \rightarrow I: L_a \rightarrow Se: d_{emi}$. The length of the causal path is equal to 2. Another causal path between the second output detector and the set of unknown inputs is $Df: y_2 \rightarrow I: J_1 \rightarrow R: R_{shaft} \rightarrow I: J_2 \rightarrow Se: d_{pert}$. The length of the causal path is equal to 2, then $n_2 = 2$ and matrix Ω defined in equation (8) is invertible. With the infinite structure, the state estimation is written as equation (17).

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + F\hat{d}(t) - \cdots$$
$$- AK \begin{pmatrix} {y_1}^{(1)}(t) - \hat{y_1}^{(1)}(t) \\ {y_2}^{(2)}(t) - \hat{y_2}^{(2)}(t) \end{pmatrix} \quad (17)$$

Null invariant zeros are studied with the bond graph model with a derivative causality assignment, drawn in Fig.5.

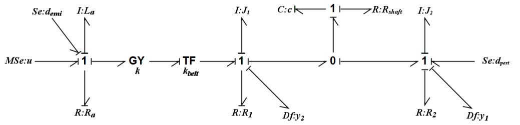


Figure 5: Bond Graph model with derivative causality.

In the BGD, the causal path between the output variable y_1 and the disturbance input d_{pert} is $Df : y_1 \rightarrow R : R_1 \rightarrow Se : d_{pert}$. The length of the causal path is equal to 0, then $n_{d_1} = 0$. The causal path between the output variable y_2 and the disturbance input d_{emi} is $Df : y_2 \rightarrow R : R_1 \rightarrow TF : k_{belt} \rightarrow GY : k_m \rightarrow R : R_a \rightarrow Se : d_{emi}$. The length of the causal path is equal to 0, then $n_{d_2} = 0$. These two paths are however not disjoint. Thus this model is not invertible for the BGD (infinite structure defined for the BGD) and matrix Ω_d is not invertible. There is a null invariant zero and the previous UIO cannot be used for this example. A solution to construct the UIO is presented below.

4 Unknown Input Observer: Torsion-Bar Application

4.1 Analysis

The torsion bar model has one null invariant zero because matrix $CA^{-1}F$ is not invertible. In that case neither matrix $C_1A^{-1}F$ nor $C_2A^{-1}F$ is equal to zero. The previous methodology cannot be directly applied. Nevertheless, it is quite similar because matrices $C_1A^{-1}F$ and $C_2A^{-1}F$ are proportional and linear combination between variables $y_1(t)$ and $y_2(t)$ can be applied in order to obtain a null row in matrix $CA^{-1}F$ before applying an integration. The procedure is now applied.

From the causal paths gains of the BGD model, or directly from matrix $CA^{-1}F$, it can be shown that the two rows are equal and then $y_1(t) - y_2(t)$ is written as in (18).

$$y_1(t) - y_2(t) = y_{1,2}(t) = (C_{1-2})A^{-1}\dot{x}(t) - \cdots$$
$$- (C_{1-2})A^{-1}Bu(t) \quad (18)$$

Where $C_{1-2} = (C_1 - C_2)$.
The primitive of variable $y_{1,2}(t)$ is thus (19)

$$\int y_{1,2}(t)\mathrm{d}t = y^*(t) = (C_{1-2})A^{-2}\dot{x}(t) - \cdots$$
$$- (C_{1-2})A^{-2}Bu(t) - (C_{1-2})A^{-2}Fd(t) - \cdots$$
$$- (C_{1-2})A^{-1}B \int u(t)\mathrm{d}t \quad (19)$$

We consider now two equations with $y_1(t)$ and $y^*(t)$ in (20). Note that it is possible to choose $y_2(t)$ and $y^*(t)$ as well. In that case, matrix Ω_d^* defined as $[(C_1A^{-1}F)^t, ((C_{1-2})A^{-2}F)^t]^t$ is invertible.

$$\left\{ \begin{array}{rl} y_1(t) = C_1[A^{-1}\dot{x}(t) - A^{-1}Bu(t) & -A^{-1}Fd(t)] \\ y^*(t) = (C_{1-2})[A^{-2}\dot{x}(t) - A^{-2} & Bu(t)\cdots \\ -A^{-2}Fd(t) - A^{-1} & B\int u(t)\mathrm{d}t] \end{array} \right. \quad (20)$$

According to (19), the disturbance equation error (4) can be rewritten as (21).

$$d(t) - \hat{d}(t) = [\Omega_d^*]^{-1} \begin{bmatrix} C_1A^{-1} \\ (C_{1-2})A^{-2} \end{bmatrix} (\dot{x}(t) - \dot{\hat{x}}(t)) \quad (21)$$

Therefore, N_{LC} for this system is rewritten as (22), and $n_1 + n_2 + 1 = 4$ which indicates that the four poles can be assigned.

$$N_{CL} = A^{-1} - A^{-1}F[\Omega_d^*]^{-1} \begin{bmatrix} C_1A^{-1} \\ (C_{1-2})A^{-2} \end{bmatrix} - K \begin{bmatrix} C_1 \\ C_2A \end{bmatrix} \quad (22)$$

4.2 Simulation

The estimation of the unknown input variables is defined in (23). For pole placement the matrix K is obtained (24). The four poles are chosen as $-1/2000, -1/2100, -1/2200$ and $-1/2200$ for the matrix N_{CL}, but they are the inverse of the classical estimation error equation.

$$\hat{d}(t) = -[\Omega_d^*]^{-1} \left[\begin{bmatrix} y_1 \\ y^* \end{bmatrix} \begin{bmatrix} C_1A^{-1} \\ (C_{1-2})A^{-2} \end{bmatrix} \dot{x}(t) + \cdots \right.$$
$$\left. + \begin{bmatrix} C_1A^{-1}B \\ (C_{1-2})A^{-2}B \end{bmatrix} u(t) \right] \quad (23)$$

$$K = \begin{bmatrix} -5.96 \cdot 10^{-11} & 1.08 \cdot 10^{-10} \\ 6.227 \cdot 10^{-7} & 8.8 \cdot 10^{-24} \\ -3.41 \cdot 10^{-10} & 6.18 \cdot 10^{-10} \\ -2.352 \cdot 10^{-8} & 4.266 \cdot 10^{-8} \end{bmatrix} \quad (24)$$

Different variables are simulated in *20-sim*. The input is chosen as $u(t) = 1v$(step response), unknown inputs are

$d_{emi} = 0.1A$ with action between 1 to 2sec. and $d_{pert} = 0.01N.m$ with time action between 3 to 4 sec.

First, the signal d_{pert} and its estimation $\hat{d}_{pert}$ are displayed in Fig.6. In Fig.7 is possible to see a zoom that shows more closely the behavior of the signals d_{pert} and $\hat{d}_{pert}$.

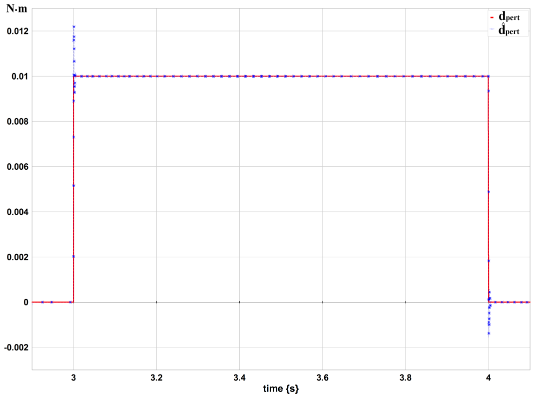


Figure 6: Unknown input d_{pert} and its estimation.

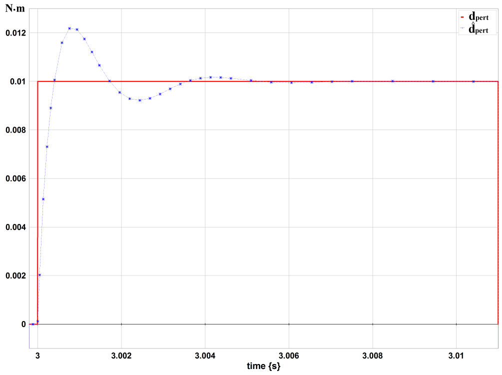


Figure 7: Zoom of unknown input d_{pert} and its estimation.

The signal d_{emi} and its estimation are shown in Fig.8, and a zoom of the signals is displayed in the Fig.9. It is possible to see the estimator reaction in this zoom.

The Fig.10a shows the output signal y_1 and its estimation $\hat{y}_1$ is displayed in Fig.10b.

Finally, the Fig.11 represents the output signal y_1 in (a) and the estimation $\hat{y}_2$ in (b).

These results show that the outputs estimations are well obtained and the estimations for disturbances have a quickly and very close responses at the disturbances.

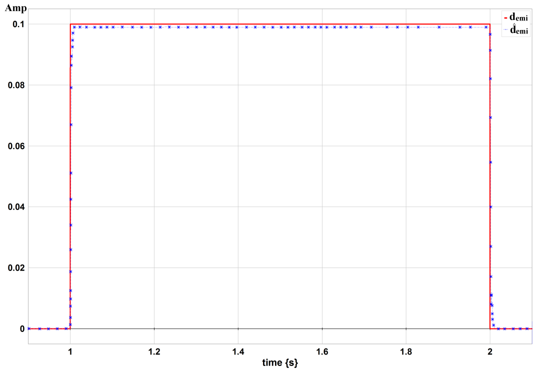


Figure 8: Perturbation signal d_{emi} and its estimation $\hat{d}_{emi}$.

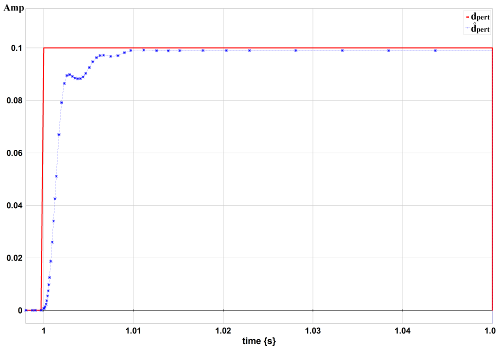


Figure 9: Zoom of perturbation signal d_{emi} and its estimation $\hat{d}_{emi}$.

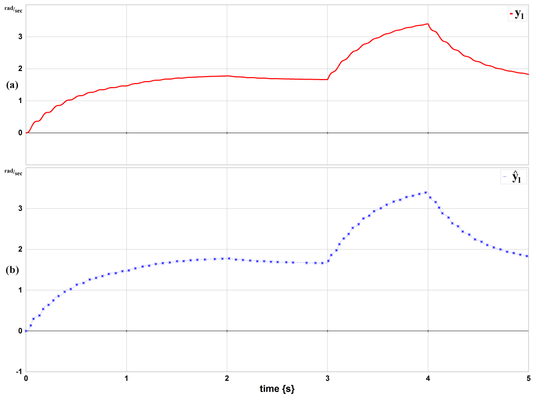


Figure 10: Output signal y_1 and the estimation $\hat{y}_1$.

5 Conclusion

An effective performance estimation system is essential in any controlled system subject to unknown inputs, failures or no well known parameters. In the literature, many solutions are based on the infinite structure of the model for the solvability conditions of the UIO and on the finite structure for the solutions with stability.

In this paper, a structural approach is proposed for the study of MIMO models which contains null invariant zeros. The innovative point is that estimation of the unknown input variables is also based on the integrals of measured

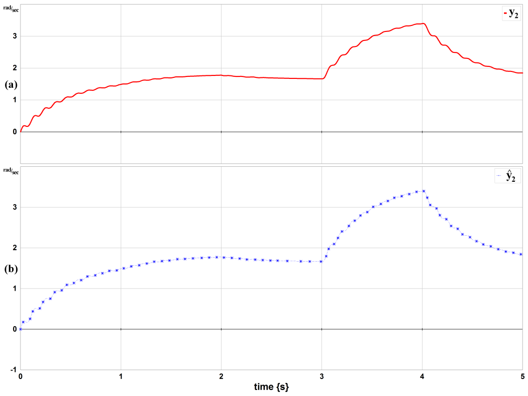


Figure 11: Output signal y_2 and the estimation $\hat{y}_2$.

variables, which allow to assign more modes in the state equation error estimation.

The simulations on an experimental bar system proved the effectiveness of the proposed UIO.

APPENDIX

A Structural Properties

A.1 Finite and infinite structures

Consider an invertible square model $\Sigma(C,A,B)$. The infinite structure of the multivariable linear model is characterized by different integer sets: $\{n'_i\}$ is the set of infinite zero orders of the global model $\Sigma(C,A,B)$ and $\{n_i\}$ is the set of row infinite zero orders of the row sub-systems $\Sigma(C_i,A,B)$. The row infinite zero order n_i verifies condition $n_i = \min\left\{k|C_iA^{(k-1)}B \neq 0\right\}$. n_i is equal to the number of derivations of the output variable $y_i(t)$ necessary for at least one of the input variables to appear explicitly. The global infinite zero orders are equal to the minimal number of derivations of each output variable necessary so that the input variables appear explicitly and independently in the equations. The infinite structure is also pointed out with the Smith-McMillan form at infinity of the transfer matrix. The finite structure of a linear model $\Sigma(C,A,B)$ is characterized by different polynomial matrices. Invariant zeros are pointed out with the Smith form of the System matrix associated to the state space representation.

A.2 Finite and infinite structures of bond graph models

Causality and causal paths are useful for the study of properties, such as controllability, observability and systems poles/zeros. Bond graph models with integral causality assignment (BGI) can be used to determine reachability conditions and the number of invariant zeros by studying the infinite structure. The rank of the controllability matrix is derived from bond graph models with derivative causality (BGD).

A LTI bond graph model is controllable iff the two following conditions are verified (Sueur and Dauphin-Tanguy 1991): first there is a causal path between each dynamical element and one of the input sources and secondly each dynamical element can have a derivative causality assignment in the bond graph model with a preferential derivative causality assignment (with a possible duality of input sources). The observability property can be studied in a similar way, but with output detectors. Systems invariant zeros are poles of inverse systems. Inverse systems can be constructed by bond graph models with bicausality (BGB) which are thus useful for the determination of invariant zeros.

The concept of causal path is used for the study of the infinite structure of the model. The causal path length between an input source and an output detector in the bond graph model is equal to the number of dynamical elements met in the path. Two paths are different if they have no dynamical element in common. The order of the infinite zero n_i for the row sub-system $\Sigma(C_i,A,B)$ is equal to the length of the shortest causal path between the i^{th} output detector z_i and the set of input sources. The global infinite structure is defined with the concepts of different causal paths. The orders of the infinite zeros of a global invertible linear bond graph model are calculated according to equation (25), where l_k is the smallest sum of the lengths of the k different input-output causal paths.

$$\left\{\begin{array}{l} n'_1 = l_1 \\ n'_k = l_k - l_{k-1} \end{array}\right. \tag{25}$$

The number of invariant zeros is determined by the infinite structure of the BGI model. The number of invariant zeros associated to a controllable, observable, invertible and square bond graph model is equal to $n - \sum n'_i$.

B Proof proposition 1

Consider the matrix product $N_{CL}F = A^{-1}F - A^{-1}F\Omega_d^{-1}\Omega_d - K[(C_1A^{n_1-1})^t(C_2A^{n_2-1})^t]^tF = -K\Omega$. Suppose that matrix Ω is not invertible. In this case, $\{n_1,n_2\} \neq \{n'_1,n'_2\}$, i.e. the row infinite structure of system $\Sigma(C,A,F)$ is different of its global infinite structure. The rank of matrix $N_{CL}.F$ is equal to 1, thus matrix N_{CL} is not invertible and the observer cannot be synthesized.

C Fixed Poles for the MIMO case, with a null invariant zero

Matrix N_{CL} in this MIMO problem is in equation (15). Pole placement is studied with the observability property of system $\Sigma([(C_1A^{n_1-1})^t,(C_2A^{n_2-1})^t]^t,N_{OL})$. Because the number of modes which can be assigned is equal to the rank of this observability matrix. The rows of the observability matrix of this system are calculated, firstly with the row matrix $C_1A^{n_1-1}$ associated with the null invariant zero, then with matrix matrix $C_2A^{n_2-1}$.

$$\begin{cases} C_1A^{n_1-1} \\ C_1A^{n_1-1}N_{OL} = C_1A^{n_1-1}\Big(A^{-1}-A^{-1}F\cdot \\ \qquad \cdot\begin{bmatrix} C_1A^{-2}F \\ C_2A^{-1}F \end{bmatrix}^{-1}\begin{bmatrix} C_1A^{-2} \\ C_2A^{-1} \end{bmatrix}\Big) = C_1A^{n_1-2} \\ C_1A^{n_1-1}(N_{OL})^2 = C_1A^{(n_1-3)} \\ \vdots \\ C_1A^{n_1-1}(N_{OL})^{n_1-2} = C_1A \\ C_1A^{n_1-1}(N_{OL})^{n_1-1} = C_1 \\ C_1A^{n_1-1}(N_{OL})^{n_1} = C_1A^{-1} \\ C_1A^{n_1-1}(N_{OL})^{n_1+1} = 0 \\ \vdots \\ C_1A^{n_1-1}(N_{OL})^{n-1} = 0 \end{cases} \tag{26}$$

A similar result is obtained for the row matrix $C_2A^{n_2-1}$, but $C_2A^{n_2-1}(N_{OL})^{n_2} = 0$.

Therefore, the non null rows of the observability matrix of system $\Sigma([(C_1A^{n_1-1})^t,(C_2A^{n_2-1})^t]^t,N_{OL})$ are thus: $[(C_1A^{-1})^t, C_1^t, (C_1A)^t, ..., (C_1A^{n_1-1})^t, C_2^t, (C_2A)^t, ...$ $, (C_2A^{n_2-1})^t]^t$. The rank of this matrix is equal to n_1+n_2+1 because model $\Sigma(C,A,F)$ is observable and for each output variable, the observability index is greater or equal to the row infinite zero order. The non null rows of the observability matrix of system $\Sigma([(C_1A^{n_1-1})^t,(C_2A^{n_2-1})^t]^t,N_{OL})$ are thus one part of the independent rows of the observability matrix of system $\Sigma(C,A)$. This rank can also be studied with the invariant subspaces defined in the geometric approach.

References

Al-Alaoui, M.A. (1993). “Novel digital integrator and differentiator”. In: *Electronics Letters* 29.4, pp. 376–378. ISSN: 0013-5194.

Basile, G. and G. Marro (1973). “A new characterization of some structural properties of linear systems: Unknown-input observability, invertibility and functional controllability”. In: *International Journal of Control* 17(5), pp. 931–943.

Bhattacharyya, S.P. (1978). “Observer design for linear systems with unknown inputs”. In: *IEEE Transactions on Automatic Control* 23, pp. 1483–484.

Carlsson, B., A. Ahlen, and M. Sternad (1991). “Optimal differentaition based on stochastic signal models”. In: *IEEE Transactions on Signal Processing* 39.2, pp. 341–353.

Dabroom, A. and H.K. Khalil (1997). “Numerical differentiation using high-gain observers”. In: *Proceedings of the 36th IEEE Conference onDecision and Control*. Vol. 5. San Diego, California USA, pp. 4790–4795.

Darouach, M. (2009). “Complements to full order observer design for linear systems with unknown inputs”. In: *Applied Mathematics Letters* 22, pp. 1107–1111.

Diop, S. et al. (1994). “Interpolation and numerical differentiation for observer design”. In: *American Control Conference*. Vol. 2, pp. 1329–1333.

Hautus, M. L. J. (1983). “Strong detectability and observers”. In: *Linear Algebra and its Applications* 50, pp. 353–368.

Hou, M. and P.C. Muller (1992). “Design of observers for linear systems with unknown inputs”. In: *IEEE Trans. Automat. Control* 37, pp. 871–875.

Levant, A. (1998). “Robust exact differentiation via sliding mode technique”. In: *Automatica* 34.3, pp. 379–384.

– (2003). “Higher order sliding modes, differentiation and output feedback control”. In: *International Journal of Control* 76.9, pp. 924–941.

Luenberger, D.G. (1971). “An introduction to observers”. In: *IEEE Trans. on Automatic Control* Vol. AC-16.No. 6, pp. 596–602.

Mehdi, Dridi (2010). “Derivation numerique : synthese , application et integration”. PhD thesis. Ecole Centrale de Lyon.

Miller, B.J. and R. Mukunden (1982). “On designing reduced-order observers for linear time-invariant systems subject to unknown inputs”. In: *Internat. J. Control* 35, pp. 183–188.

Sueur, C. and G. Dauphin-Tanguy (1991). “Bond-graph approach for structural analysis of MIMO linear systems”. In: *Journal of the Franklin Institute* 328, pp. 55–70.

Tarasov, E. et al. (2013). “State and unknown input observer: analysis and design”. In: *7th IMAACA’13, part of 10th I3M2013, Athens, September 25-27, Greece*.

Yang, D. et al. (2013). “New Unknown Input Observer for control design: a bond graph approach”. In: *SSSC-IFAC 2013, Grenoble, France*.

A NEW APPROACH TO THE DERIVATION OF A SINGLE SET OF IMPLICIT STATE EQUATIONS FROM A FIXED CAUSALITY BOND GRAPH OF A HYBRID MODEL

W. Borutzky
Department of Computer Science
Bonn-Rhein-Sieg University of Applied Sciences, St. Augustin, Germany

ABSTRACT

This paper proposes a novel approach to the generation of state equations from a bond graph (BG) of a mode switching linear time invariant model. Fast state transitions are modelled by ideal or non-ideal switches. Fixed causalities are assigned following the Standard Causality Assignment Procedure such that the number of storage elements in integral causality is maximised. A system of differential and algebraic equations (DAEs) is derived from the BG that holds for all system modes. It is distinguished between storage elements with mode independent causality and those that change causality due to switch state changes.

A matrix based approach is used for the general case. For illustration, small circuit examples with a special feature are considered. Equations are directly derived from their BG. Switch state commutations may change the index of the DAE system. This is addressed by modelling and simulating one of the examples in the OpenModelica environment.

Keywords: Hybrid models, bond graphs, state equations generation, numerical solution.

1. INTRODUCTION

Various BG representations of hybrid models have been proposed in the literature. A survey may be found in (Borutzky, 2010, Chapter 7). Strömberg et. al. (Strömberg et al, 1993) introduced the ideal switch as another basic BG element excepting that computational causalities at least in some parts of a BG become mode dependent. In the same year, Asher introduced so-called causality resistors (Asher, 1993) which compensate for the causality change at a switch port so that other elements are not affected. In particular, storage elements in preferred integral causality can keep their causality. Disadvantages, however, are that causality resistors can lead to widely different time constants. Their resistance value has to be chosen with care. Buisson et. al. used ideal switches and presented a matrix-based approach to the generation of an implicit state equation from the BG of a mode switching linear time-invariant (LTI) model for a reference mode. Equations for any other system mode are derived from the set of equations for the reference mode (Buisson et al, 2002; Buisson, 1993). However, for systems with n ideal switches there can be quite a number, n_f, of physically feasible switch state combinations $n_f \leq 2^n$ and for each combination a set of implicit state equations must be derived. Moreover, the use of ideal switches means that the preferred integral causality at storage ports is not preserved. The dimension of the state vector becomes mode-dependent. When a closing switch directly connects two storage elements then they become dependent. Their states are algebraically related. As a consequence, simulation software must detect such commutation events, must re-initialise state variables because of a discontinuous jump of their values and continue the computation of the dynamic behaviour with a new set of equations and a smaller number of state variables.

Margetts (Margetts, 2013) also follows the variable causality approach but uses controlled junctions introduced by Mosterman (Mosterman, 1997) for modelling the abstraction of instantaneous discrete switching, marks causalities in the BG that change due to switch state changes by additional dashed causal strokes that indicate the causal configuration after the commutation of some switches and distinguishes between static and dynamic causalities. Other than Buisson et. al. she derives a single set of implicit state equations that holds for all system modes.

In this paper, devices or phenomena with fast state transitions such as electronic diodes and transistors, clutches, or hard mechanical stops are modelled by ideal or non-ideal switches. The unmodified Standard Causality Assignment Procedure (SCAP) is used to once assign *fixed* static causalities so that the number of storage elements in integral causality is maximised. (There may be some storage elements in static derivative causality independent of any switch state changes.) Additional causal strokes are not assigned to bonds. As a result, the BG reflects the configuration only for a specific system mode. Actually, causality at some storage ports may change due to the commutation of some switches. That is, some storage ports change into derivative causality in some system modes, i.e. the number of state variables may be mode-dependent. Therefore, as in (Buisson et al, 2002; Margetts, 2013) it is distinguished between storage elements with mode independent causality and those that change causality due to switch state changes. The latter ones can be identified by inspecting causal paths between storage elements and switches. It is shown that by considering causal paths between storage elements and switches a single mode-dependent DAE system can be derived from a BG with static causalities that hold for all system modes.

As in (Buisson et al, 2002), the presented matrix-based equations formulation starts from the well known equivalent block diagram of a general BG extended by a field of switches and a partitioning of the junction structure (JS) matrix. However, other than in (Buisson et al, 2002), a BG, once causally augmented by means of the unmodified SCAP, is used for all modes of operation, i.e. for all (physically feasible) switch state configurations. That is, an actual discrete switch states configuration is not graphically expressed by causal strokes but is taken into account by the values of the discrete switch states. The latter ones are annotated to the switch symbols, i.e. Sw : *m*. (The non-standard symbol Sw : *m* denotes a switch that may be either ideal or non-ideal.) This representation may be compared to the *fixed* causality representation of non-ideal switch models composed of a Boolean modulated transformer, MTF : *b*, and an ON-resistor, R : R_{on}, in fixed conductance causality, in which the value of the transformer modulus, *b*, accounts for the switch state (Ducreux et al, 1993; Borutzky, 2015).

In this paper, the causality the switches have received by application of the SCAP is disregarded. Instead, the commutation of the switches is always described by an *implicit* equation. Equations obtained from the partitioning of the JS are reformulated so that the outputs of the JS into the switch field can be inserted into the switch equation resulting in an algebraic equation for the outputs of the switches. Its solution is used in the ordinary differential equations (ODEs) for the state variables.

The approach is illustrated by three small circuit examples with each of them having a special feature. Their equations are directly derived from the equivalent BG by following causal paths. Depending on the switch states, the differential equation (ODE) for some storage elements can turn into an algebraic equation which has an affect on the numerical solution of the model. This is addressed in the last section by modelling and simulating one of the examples in the OpenModelica software environment (OpenModelica Consortium, n.d.).

2. EQUATIONS GENERATION

For simplicity it is assumed that the BG of a mode switching LTI model does not contain storage elements that are of static derivative causality independent of any switch states. That is, all storage elements are in integral causality after application of the SCAP. However, there may be causal paths between storage and switch ports which means that a change of a switch state would change the causality of a storage port connected to the switch through a causal path. Let $\boldsymbol{x}_{ii}$ denote the state of all storage elements that are not affected by any switch state changes and remain in integral causality and let $\boldsymbol{x}_{id}$ be the state of all those storage elements in integral causality that would change causality due to a state change of a switch which is not explicitly expressed in the BG as causalities remain fixed once they have been assigned following the SCAP. Nevertheless, the components of the vector $\boldsymbol{x}_{id}$ can be determined by inspecting the BG for causal paths between a storage element in integral causality and a switch. The complementary state variables are grouped into the vectors $\boldsymbol{z}_{ii}$, $\boldsymbol{z}_{id}$ respectively and $\boldsymbol{x}_i = [\boldsymbol{x}_{ii}^T\ \boldsymbol{x}_{id}^T]^T$, $\boldsymbol{z}_i = [\boldsymbol{z}_{ii}^T\ \boldsymbol{z}_{id}^T]^T$.

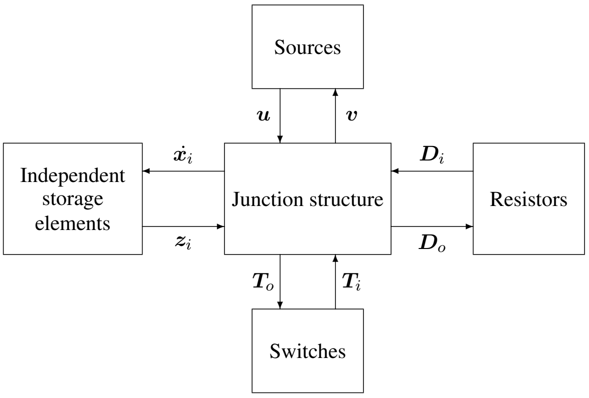


Figure 1: Equivalent block diagram of a general BG with no storage elements in derivative causality

Fig. 1 depicts the well known equivalent block diagram of a general BG with no storage elements in static derivative causality extended by a field of ideal switches.

According to the input and output signals of the junction structure in Fig. 1 the following equations can be set up if it can be assumed that there is no unity gain loop in the junction structure.

$$\underbrace{\begin{bmatrix}\dot{\boldsymbol{x}}_{ii}\\ \dot{\boldsymbol{x}}_{id}\end{bmatrix}}_{\dot{\boldsymbol{x}}_i} = \underbrace{\begin{bmatrix}\mathbf{S}_{11} & \mathbf{S}_{12}\\ -\mathbf{S}_{12}^T & \mathbf{0}\end{bmatrix}}_{\mathbf{A}_{11}} \underbrace{\begin{bmatrix}\boldsymbol{z}_{ii}\\ \boldsymbol{z}_{id}\end{bmatrix}}_{\boldsymbol{z}_i} + \underbrace{\begin{bmatrix}\mathbf{S}_{13}\\ \mathbf{S}_{23}\end{bmatrix}}_{\mathbf{A}_{12}} \boldsymbol{D}_i + \underbrace{\begin{bmatrix}\mathbf{S}_{14}\\ \mathbf{S}_{24}\end{bmatrix}}_{\mathbf{A}_{13}} \boldsymbol{T}_i + \underbrace{\begin{bmatrix}\mathbf{S}_{15}\\ \mathbf{S}_{25}\end{bmatrix}}_{\mathbf{A}_{14}} \boldsymbol{u} \tag{1}$$

$$\boldsymbol{D}_o = \underbrace{\begin{bmatrix}-\mathbf{S}_{13}^T & -\mathbf{S}_{23}^T\end{bmatrix}}_{\mathbf{A}_{21}} \begin{bmatrix}\boldsymbol{z}_{ii}\\ \boldsymbol{z}_{id}\end{bmatrix} + \mathbf{S}_{33}\boldsymbol{D}_i + \mathbf{S}_{34}\boldsymbol{T}_i + \mathbf{S}_{35}\boldsymbol{u} \tag{2}$$

$$\boldsymbol{T}_o = \underbrace{\begin{bmatrix}-\mathbf{S}_{14}^T & -\mathbf{S}_{24}^T\end{bmatrix}}_{\mathbf{A}_{31}} \begin{bmatrix}\boldsymbol{z}_{ii}\\ \boldsymbol{z}_{id}\end{bmatrix} + \mathbf{S}_{43}\boldsymbol{D}_i + \mathbf{S}_{44}\boldsymbol{T}_i + \mathbf{S}_{45}\boldsymbol{u} \tag{3}$$

The constitutive equation of the resistive field reads

$$\boldsymbol{D}_i = \mathbf{L}\boldsymbol{D}_o \tag{4}$$

Substituting (4) into (2) gives

$$(\mathbf{I} - \mathbf{S}_{33}\mathbf{L})\boldsymbol{D}_o = \mathbf{A}_{21}\boldsymbol{z}_i + \mathbf{S}_{34}^T\boldsymbol{T}_i + \mathbf{S}_{35}\boldsymbol{u} \tag{5}$$

The commutations of all n_s ideal switches is expressed by the *implicit* equation

$$\mathbf{M}\boldsymbol{T}_o + \bar{\mathbf{M}}\boldsymbol{T}_i = \mathbf{0} \tag{6}$$

where $\mathbf{M}$ is a diagonal matrix with entries $m_{jj} \in \{1,0\}$, $j = 1,\ldots,n_s$ and $\bar{\mathbf{M}} := \mathbf{I} - \mathbf{M}$. The assignment of fixed static causalities for all elements in a BG of a hybrid model including the switches and (6) are a key point in the presented approach.

Substituting (5) into (4) and the result into (3) assuming that $\mathbf{H} := \mathbf{L}(\mathbf{I} - \mathbf{S}_{33}\mathbf{L})^{-1}$ is non-singular and replacing $\boldsymbol{T}_o$ in the switch equations (6) yields an implicit algebraic equation for $\boldsymbol{T}_i$.

$$\begin{aligned}\underbrace{\left[\bar{\mathbf{M}} + \mathbf{M}(\mathbf{S}_{44} + \mathbf{S}_{43}\mathbf{H}\mathbf{S}_{34})\right]}_{\mathbf{N}_1}\boldsymbol{T}_i = \\ -\underbrace{\left[\mathbf{M}(\mathbf{A}_{31} + \mathbf{S}_{34}^T\mathbf{H}\mathbf{A}_{21})\right]}_{\mathbf{N}_2}\boldsymbol{z}_i \\ -\underbrace{\left[\mathbf{M}(\mathbf{S}_{45} + \mathbf{S}_{43}\mathbf{H}\mathbf{S}_{35})\right]}_{\mathbf{N}_3}\boldsymbol{u}\end{aligned} \tag{7}$$

Replacing $\boldsymbol{D}_i$ in (1) gives

$$\begin{aligned}\dot{\boldsymbol{x}}_i = {} & (\mathbf{A}_{11} + \mathbf{A}_{12}\mathbf{H}\mathbf{A}_{21})\boldsymbol{z}_i + (\mathbf{A}_{13} + \mathbf{A}_{12}\mathbf{H}\mathbf{S}_{34})\boldsymbol{T}_i \\ & + (\mathbf{A}_{14} + \mathbf{A}_{12}\mathbf{H}\mathbf{S}_{35})\boldsymbol{u}\end{aligned} \tag{8}$$

Equations 8, 7 constitute a unique DAE system for all modes of operation derived from a BG with fixed static causalities. The complementary state vector $\boldsymbol{z}_i$ can be replaced in (7), (8) by the constitutive equation of the storage field.

$$\begin{bmatrix}\boldsymbol{z}_{ii} \\ \boldsymbol{z}_{id}\end{bmatrix} = \underbrace{\begin{bmatrix}\mathbf{F}_{11} & \mathbf{F}_{12} \\ \mathbf{F}_{12}^T & \mathbf{F}_{22}\end{bmatrix}}_{\mathbf{F}}\begin{bmatrix}\boldsymbol{x}_{ii} \\ \boldsymbol{x}_{id}\end{bmatrix} \tag{9}$$

where $\mathbf{F}$ is symmetric positive definite.

For the states of all storage elements with mode independent causality the ODE reads

$$\begin{aligned}\dot{\boldsymbol{x}}_{ii} = {} & \underbrace{(\mathbf{S}_{11} - \mathbf{S}_{13}\mathbf{H}\mathbf{S}_{13}^T)}_{\mathbf{A}_1}\boldsymbol{z}_{ii} + \underbrace{(\mathbf{S}_{12} - \mathbf{S}_{13}\mathbf{H}\mathbf{S}_{23}^T)}_{\mathbf{A}_2}\boldsymbol{z}_{id} \\ & + \underbrace{(\mathbf{S}_{14} + \mathbf{S}_{13}\mathbf{H}\mathbf{S}_{34})}_{\mathbf{A}_3}\boldsymbol{T}_i + \underbrace{(\mathbf{S}_{15} + \mathbf{S}_{13}\mathbf{H}\mathbf{S}_{35})}_{\mathbf{A}_4}\boldsymbol{u}\end{aligned} \tag{10}$$

Substituting (5) into (4) and the result into (1) gives another equation for $\boldsymbol{T}_i$.

$$\begin{aligned}\underbrace{(\mathbf{S}_{24} + \mathbf{S}_{23}\mathbf{H}\mathbf{S}_{34})}_{:=\mathbf{K}_1}\boldsymbol{T}_i = {} & \dot{\boldsymbol{x}}_{id} + \underbrace{(\mathbf{S}_{12}^T + \mathbf{S}_{23}\mathbf{H}\mathbf{S}_{13}^T)}_{:=\mathbf{K}_2}\boldsymbol{z}_{ii} \\ & + \underbrace{\mathbf{S}_{23}\mathbf{H}\mathbf{S}_{23}^T}_{:=\mathbf{K}_3}\boldsymbol{z}_{id} \\ & - \underbrace{(\mathbf{S}_{25} + \mathbf{S}_{23}\mathbf{H}\mathbf{S}_{35})}_{:=\mathbf{K}_4}\boldsymbol{u}\end{aligned} \tag{11}$$

or

$$\mathbf{K}_1\boldsymbol{T}_i = \dot{\boldsymbol{x}}_{id} + \underbrace{[\mathbf{K}_2\ \mathbf{K}_3]}_{\mathbf{K}}\boldsymbol{z}_i - \mathbf{K}_4\boldsymbol{u} \tag{12}$$

Given that $\mathbf{K}_1$ can be assumed to be non-singular then substitution of (12) into (10) results in the first part of a final DAE system.

$$\begin{aligned}\dot{\boldsymbol{x}}_{ii} - \mathbf{A}_3\mathbf{K}_1^{-1}\dot{\boldsymbol{x}}_{id} = {} & [\mathbf{A}_1 + \mathbf{A}_3\mathbf{K}^{-1}\mathbf{K}_2]\boldsymbol{z}_{ii} \\ & + [\mathbf{A}_2 + \mathbf{A}_3\mathbf{K}^{-1}\mathbf{K}_3]\boldsymbol{z}_{id} \\ & + (\mathbf{A}_4 - \mathbf{A}_3\mathbf{K}^{-1}\mathbf{K}_4)\boldsymbol{u}\end{aligned} \tag{13}$$

Substituting (12) into (7) gives the second part of the final implicit DAE system

$$\begin{aligned}\mathbf{N}_1\mathbf{K}_1^{-1}\dot{\boldsymbol{x}}_{id} = {} & [\mathbf{N}_2 - \mathbf{N}_1\mathbf{K}_1^{-1}\mathbf{K}]\boldsymbol{z}_i \\ & - [\mathbf{N}_3 - \mathbf{N}_1\mathbf{K}_1^{-1}\mathbf{K}_4]\boldsymbol{u}\end{aligned} \tag{14}$$

Equations 13 and 14 constitute an implicit unique DAE system for all system modes derived from a BG with fixed causalities. Accounting for the constitutive equation of the storage field, (9), the final implicit DAE system is of the form

$$\begin{bmatrix}\mathbf{I} & -\mathbf{A}_3\mathbf{K}_1^{-1} \\ \mathbf{0} & \mathbf{N}_1\mathbf{K}_1^{-1}\end{bmatrix}\underbrace{\begin{bmatrix}\dot{\boldsymbol{x}}_{ii} \\ \dot{\boldsymbol{x}}_{id}\end{bmatrix}}_{\dot{\boldsymbol{x}}_i} = rhs(\boldsymbol{x}_i, \boldsymbol{u}) \tag{15}$$

where *rhs* denotes the right hand side of the DAE system.

In (14), the variables are multiplied by mode dependent matrices. That is, in some cases, the ODE can turn into an algebraic equation, or the right hand side can become zero.

3. ILLUSTRATIVE EXAMPLES

The matrix based equations formulation presented in the previous section can be implemented in a script or coded in a programming language and can be used for an automatic equations generation from BGs of large scale mode switching LTI models. In this section, equations are derived manually from the BGs of three small illustrative example systems by following causal paths and are inserted into the switch equations. For the first two examples with one single switch, the derivation of equations leads to a DAE system with a *mode-dependent index*. In one of the two modes of operation, the system is structurally singular.

3.1 Circuit with a single switch in series with an inductor

As a first example, consider the simple circuit in Fig. 2 with an ideal switch and an inductor in series.

In case the switch is open and fully disconnects inductance L_1 from the two right hand side storage elements, i.e. its OFF conductance is assumed to be zero,

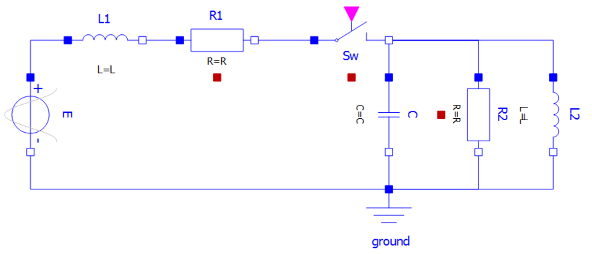


Figure 2: Circuit with an ideal switch and an inductor in series

then the model order is two. In case the switch is closed the model order is three. That is, the number of state variables is mode-dependent. If a non-zero ON resistance of the switch is taken into account and if the switch is modelled as a mode switching resistor with a *fixed* conductance causality then the latter one either causes a causal conflict with the integral causality at the inductor I : L_1 or the inductor would be forced into derivative causality.

The proposed approach adopts an ideal switch model and applies the unmodified SCAP. As a result, all storage elements are in integral causality. By consequence, the switch receives an effort out causality. That is, the causal BG depicted in Fig. 3 reflects the configuration when the switch is closed. The causal path between the switch and the inductor highlighted in red indicates that their causalities change oppositely, i.e. if the open switch state would be taken into account by a flow out causality, then the inductor would be correctly forced into derivative causality.

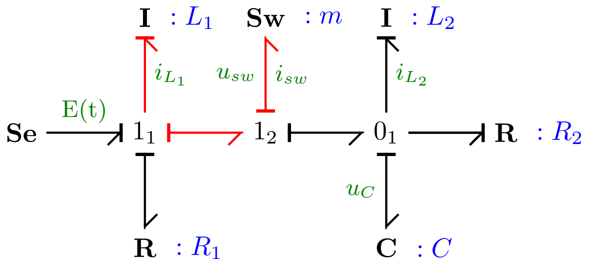


Figure 3: BG of the circuit in Fig. 2

Nevertheless, a single DAE system can be derived from the BG with fixed causalities that holds for both system modes.

$$Sw: \qquad 0 = m u_{\text{sw}} + \bar{m} i_{\text{sw}} \tag{16}$$

$$L_1: \qquad L_1 \frac{d}{dt} i_{L_1} = E - R_1 i_{L_1} - u_{\text{sw}} - u_C \tag{17}$$

$$C: \qquad C \dot{u}_C = i_{\text{sw}} - i_{L_2} - \frac{u_C}{R_2} \tag{18}$$

$$L_2: \qquad L_2 \frac{d}{dt} i_{L_2} = u_C \tag{19}$$

Only inductor I : L_1 is affected by a causality change at the switch. This is reflected by the causal path between the inductor I : L_1 and the switch. Accordingly, let $\boldsymbol{x}_{ii} := [u_C, i_{L_2}]^T$ and $\boldsymbol{x}_{id} := [i_{L_1}]$. Substituting (17) into the switch equation (16) and accounting for $i_{\text{sw}} = i_{L_1} =: i$ yields an implicit DAE that holds for both modes of operation.

$$\left[\begin{array}{cc|c} C & 0 & 0 \\ 0 & L_2 & 0 \\ \hline 0 & 0 & mL_1 \end{array}\right] \frac{d}{dt} \left[\begin{array}{c} u_c \\ i_{L_2} \\ \hline i \end{array}\right] = \left[\begin{array}{cc|c} -1/R_2 & -1 & 1 \\ 1 & 0 & 0 \\ \hline -m & 0 & \bar{m} - mR_1 \end{array}\right] \left[\begin{array}{c} u_c \\ i_{L_2} \\ \hline i \end{array}\right] + \left[\begin{array}{c} 0 \\ 0 \\ \hline m \end{array}\right] E \tag{20}$$

In (20), the ODE for the inductor current i in the last row turns into the correct algebraic constraint $i = 0$ in case $m = 0$ (open switch).

The ideal switch could also be replaced by a non-ideal switch in fixed conductance causality with an ON resistance R_{on} (forcing inductor $I : L_1$ into derivative causality. Replacing u_{sw} in the new explicit switch equation $i = m\, u_{\text{sw}}/R_{\text{on}}$ then gives

$$m L_1 \frac{di}{dt} = -(R_{\text{on}} + mR_1) i - m u_C + mE \tag{21}$$

which is also correct for both system modes.

3.2 Circuit with dependent resistors and dependent storage elements

If the switching element in the circuit of Fig. 4 is modelled by an ideal switch then one obtains a higher index problem in case the switch is closed. That is, the index of the DAE system derived from the circuit schematic or from the BG in Fig. 5 is *mode dependent*. The same holds true if a clutch between two inertia is modelled by an ideal switch.

Moreover, as can be seen from the BG in Fig. 5, there is a causal path between the resistors R : R_1 and R : R_2 highlighted in blue indicating that there is an algebraic loop. The latter one can be solved if the constitutive relations of the resistors are linear.

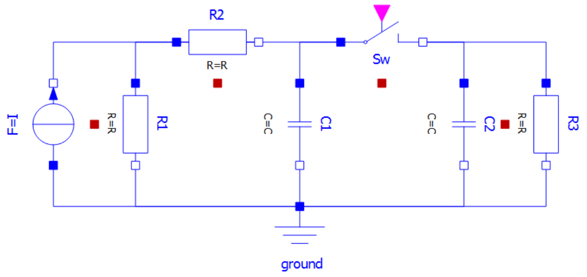


Figure 4: Switched circuit with two dependent resistors and two dependent capacitors

From the BG in Fig. 5, the following equations can be directly derived.

$$u_1 = R_1 (F - i_2) \tag{22}$$

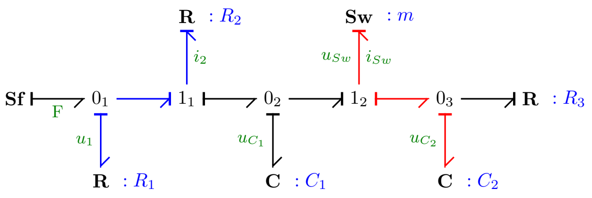


Figure 5: BG of the circuit in Fig. 4

$$\begin{aligned} i_2 &= \frac{1}{R_2}(u_1 - u_{C_1}) && (23)\\ C_1\,\dot u_{C_1} &= i_2 - i_{\mathrm{Sw}} && (24)\\ C_2\,\dot u_{C_2} &= i_{\mathrm{Sw}} - \frac{u_{C_2}}{R_3} && (25)\\ u_{\mathrm{Sw}} &= u_{C_1} - u_{C_2} && (26)\\ 0 &= m\,u_{\mathrm{Sw}} + \bar m i_{\mathrm{Sw}} && (27) \end{aligned}$$

There is another causal path between capacitor $C : C_1$ and the switch and another one between $C : C_2$ and the switch. However, if the switch changes its state, the causality at only one of the two capacitors is affected. The causality at the other capacitor remains fixed. Let $x_{ii} := u_{C_1}$ and $x_{id} := u_{C_2}$. Substituting u_{Sw} and i_{Sw} into the switch equation gives

$$\left[\begin{array}{c|c} C_1 & C_2 \\ 0 & \bar m C_2 \end{array}\right] \left[\begin{array}{c} \dot u_{C_1} \\ \hline \dot u_{C_2} \end{array}\right] = \left[\begin{array}{c|c} -1/(R_1+R_2) & -1/R_3 \\ \hline -m & (m - \bar m/R_3) \end{array}\right] \left[\begin{array}{c} u_{C_1} \\ \hline u_{C_2} \end{array}\right] + \left[\begin{array}{c} R_1/(R_1+R_2) \\ \hline 0 \end{array}\right] F \qquad (28)$$

As in the previous example, the result is an implicit DAE that holds for both system modes. In case the switch is closed ($m = 1$), the ODE for $\dot u_{C_2}$ turns into the algebraic equation $u_{C_2} = u_{C_1}$.

3.3 Circuit with two independent switches

The third example illustrates the application of the approach to a simple circuit with two ideal switches that may change their state independently. Fig. 6 shows the circuit diagram.

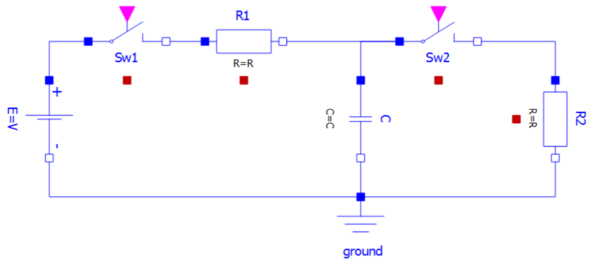


Figure 6: Circuit with two independent switches

From the BG in Fig. 7 it can be seen that there is a causal path from the capacitor $\mathrm{C} : C$ to the switch $\mathrm{Sw} : m_1$. However, the causality of the C element is not affected when the two switches change their state in case linear resistors can be assumed. The path between the C element and the switch $\mathrm{Sw} : m_1$ can be removed by changing the causalities at the switch $\mathrm{Sw} : m_1$ and the resistor $\mathrm{R} : R_1$. The two causal paths from switch $\mathrm{Sw} : m_1$ to resistor $\mathrm{R} : R_1$ and from switch $\mathrm{Sw} : m_2$ to resistor $\mathrm{R} : R_2$ highlighted in blue indicate that the elements in each of the two disjunct paths are algebraically dependent.

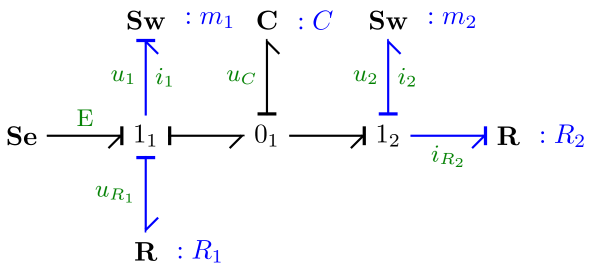


Figure 7: BG of the circuit in Fig. 6

From the BG in Fig. 7, the following equations can be directly derived.

$$\begin{aligned} \dot q = C\dot u_C &= i_1 - i_2 && (29)\\ u_1 &= E - R_1 i_1 - u_C && (30)\\ 0 &= m_1 u_1 + \bar m_1 i_1 && (31)\\ i_2 &= \frac{1}{R_2}(u_C - u_2) && (32)\\ 0 &= m_2 u_2 + \bar m_2 i_2 && (33) \end{aligned}$$

According to the causal paths between a resistor and a switch in the BG, (30), (31) can be solved for i_1 and (32), (33) yields i_2. Hence,

$$\dot q = \frac{m_1}{m_1 R_1 - \bar m_1}(E - u_C) - \frac{m_2}{m_2 R_1 - \bar m_2} u_C \qquad (34)$$

which holds for all four system modes. In case both switches are open ($m_1 = m_2 = 0$), (34) correctly yields $\dot q = 0$.

4. SOLUTION OF THE DERIVED EQUATIONS

Once a single DAE system for all modes of operation has been derived from a BG of a hybrid model, the question is how the equations can be solved numerically. This problem is discussed by considering the circuit in Fig. 4. It is assumed that the switch abruptly closed at $t_0 = 2ms$ and opens abruptly again at $t_1 = 4ms$, i.e. $m = 1\ \forall\, t \in [2ms, 4ms]$.

4.1 Direct formulation of the derived equation in Modelica

If the equations 28 are directly formulated in the Modelica language, then the OpenModelica software will notice that there are as many equations as unknowns, will solve

Table 1: Parameters of the circuit in Fig 4

```
record parameters ”Parameters IdealSwCircuit”
  //
  parameter Real C1 = 1e-6 ”Farad”;
  parameter Real C2 = 1e-6 ”Farad”;
  parameter Real R1 = 1e+3 ”Ohms”;
  parameter Real R2 = 1e+3 ”Ohms”;
  parameter Real R3 = 1e+3 ”Ohms”;
  // source:
  parameter Real tstart1 = 0.0;
  parameter Real h1 = 5e-3 ”Amps”;
  // trigger:
  parameter Real tstart2 = 0.002;
  parameter Real tstop = 0.004;
  parameter Real eps = 1e-10;
  parameter Real h2 = 1.0 - eps;
end parameters;
```

symbolically the second equation for $\dot{u}_{C2}$ and compile the result. The simulation, however, will fail at $t_0 = 2\,ms$ with a *division by zero*. When the switch closes ($\bar{m} = 0$) the ODE for $\dot{u}_{C2}$ turns into the algebraic constraint $u_{C_2} = u_{C_1}$. The new DAE system for $t \in [2ms, 4ms]$ then needs index reduction. However, symbolic index reduction by means of an implementation of the Pantelides algorithm requires that the index of a DAE system does not change (Cellier and Krebs, 2007). Index reduction is not done *at runtime*. OpenModelica assumes that a hybrid DAE representation is *structurally time invariant*. That is, the set of variables and the set of equations remain fixed during a simulation. Discrete switch states, however, can turn some equations off or on so that the number of *active* equations can vary at run-time (Fritzson, 2004). (For a discussion on hybrid models and their numerical solution see also (Borutzky, 2016, Chap. 5))

4.2 One set of ODEs for each system mode

One pragmatic way to overcome this problem of a mode-dependent DAE index in OpenModelica is to differentiate the algebraic constraint in the case the ideal switch closes ($\bar{m} = 0$) and directly connects the two capacitor voltages so that a set of two ODEs can be provided for each of the two modes.

In addition, both state variables must be initialised at the switching instant $t_0 = 2\,ms$ with a joint value to which both capacitor voltages jump when the switch closes (cf. Lines 16 and 17 of the listing in Fig. 8). This value can be obtained by accounting for *charge conservation*. Let q denote the charge of the capacitors, $u_{C_1}(t_0^-)$ the left side limit and $u_{C_1}(t_0^+)$ the right side limit of $u_{C_1}(t)$.

$$q = C_1 u_{C_1}(t_0^-) \; = \; (C_1 + C_2) u_{C_1}(t_0^+) \qquad (35)$$

When the switch is open, only capacitor C_1 is charged. Solving the ODE for u_{C_1} and using the parameters in Table 1 gives $u_{C_1}(t_0^-) = 5 \cdot (1 - e^{-1}) = 3.1606$.

The switching event at $t_1 = 4\,ms$ does not require additional action as both voltages start from the same value.

```
 1 model Circ2
 2   ...
 3 equation
 4   ...
 5   //
 6   if m < 1 then
 7     C1 * der(uC1) + C2 * der(uC2) = (-uC1 / (R1 + R2))
 8     - uC2 / R3 + R1 * F / (R1 + R2);
 9     C2 * der(uC2) = -uC2 / R3;
10   else
11     C1 * der(uC1) + C2 * der(uC2) = (-uC1 / (R1 + R2))
12     - uC2 / R3 + R1 * F / (R1 + R2);
13     der(uC2) = der(uC1);
14   end if;
15   when pre(m) < m then
16     reinit(uC1, 1.5803);
17     reinit(uC2, 1.5803);
18   end when;
19 end Circ2;
```

Figure 8: Formulation of equations 28 in Modelica

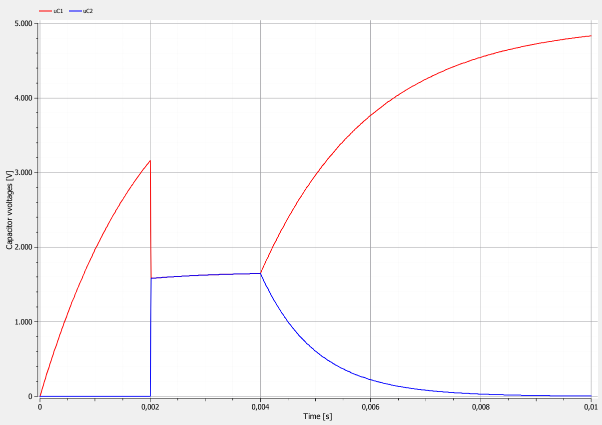


Figure 9: Voltages $u_{C_1}(t)$, $u_{C_2}(t)$ in the circuit of Fig 4 with the switch closed for $t \in [2ms, 4ms]$

Fig. 8 shows part of the Modelica formulation that is accepted by the OpenModelica compiler and for which a simulation provides the correct simulation results displayed in Fig. 9.

4.3 Escape to a non-ideal mode-dependent switch

A DAE system that can be directly formulated in Modelica and solved without problem in OpenModelica can be obtained if the plus sign in the implicit equation of an ideal switch is replaced by a minus sign.

$$0 \; = \; m\, u_{\mathrm{Sw}} - \bar{m}\, i_{\mathrm{Sw}} \qquad (36)$$

with $m \in \{\varepsilon, 1 - \varepsilon\}$ and $0 < \varepsilon \ll 1$.

As a result, in the implicit DAE system, (28), only the coefficients in the second row of the system matrix change their signs. The physical meaning of this modification is that the ideal switch connecting the two parallel capacitors have been replaced by a *non-ideal* switch with a mode-dependent resistance $R_{\mathrm{Sw}}(m) := (1-m)/m$ that is very large when the switch is open and very small when it is closed. Advantages are that the equations obtained can be directly formulated in Modelica. There is no need to determine the joint value to which both voltages instantaneously jump when the switch is closed in order to re-initialise the numerical integration. Simulation results are the same as in Fig. 9. For this simple, but typical example, neither a loss of accuracy nor a loss of performance is noticeable. For larger models, non-ideal switches with a large OFF-resistance and a small ON-resistance may lead to small time constants and to a longer computational run time. However, in general, not all ideal switches need to be replaced by non-ideal ones.

A similar problem arises with the series connection of an inductor and an ideal switch as in the first example circuit (Fig. 2). One way to overcome the problem in this case is to add a resistor of high resistance between the connection node of inductor and switch and ground. Another option is to replace the ideal switch in series with an inductor by a non-ideal one. As a result, in configurations as the two ones considered in Section 3.1 and 3.2, ideal switches should be replaced by on-ideal ones.

In (Cellier and Kofman, 2006), equations are directly read from the circuit schematic of a half-way rectifier with an inductor and a diode in series as an example of a model with variable structure. The relations between variables and equations are analysed by means of a digraph and it is shown that the equations cannot be solved in case the diode is modelled as an ideal switch because a division by zero occurs when the switch opens. Furthermore it is shown that the problem can be overcome by turning the differential equation of the inductor into an algebraic equation by means of an integration formula. This step the authors call inline integration is inspired by the approach implemented in the circuit simulator Spice in which the constitutive equation of a storage element is replaced by an integration formula. The graphical representation is known as the companion model of a storage element.

Approaches to the numerical solution of the generated equations as the two one presented in Sections 4.2 and 4.3 are necessary because software such as OpenModelica currently can perform an index reduction for a given mode-independent DAE system in a symbolic preprocessing phase prior to compilation and subsequent simulation but not during simulation when an instantaneous mode change results in a change of the index. The problem of handling mode-dependent DAE systems of varying index described in Modelica has been discussed in (Mattsson et al (2015); Elmqvist et al (2014)). First approaches have been implemented in a prototype of the commercial modelling and simulation environment Dymola (Dassault Systèmes (2015)). Also, recently, Zimmer has extended concepts of the Modelica language and has developed an experimental DAE processor that can handle changes in the set of equations at run time (Zimmer, 2013).

CONCLUSION

The paper starts from a BG with ideal or non-ideal switches of mode switching LTI models to which *fixed* causalities have been assigned and presents an approach to the generation of a single set of implicit state space equations that hold for all modes of operation. The novelty of the approach is that no additional causal strokes, or no derivation of equations for a new switch state configuration from the equations of a reference configuration is needed. Dynamic equations for a specific switch state configuration are obtained by just setting the values of the discrete switch states in the mode-dependent state equations accordingly. That is, several switches may commutate simultaneously. The procedure for the generation of the single implicit state equations with coefficient depending on the discrete switch states can be implemented in a script or in a programming language.

In some modes of operation, when closing ideal switches directly connect storage elements so that they become dependent, or when an ideal switch is in series with an inductor, some of the ODEs may turn into algebraic constraints so that index reduction would be required during simulation which, however, is not done by the OpenModelica software. In such cases, OpenModelica cannot solve the mode-dependent equations directly and needs some help from an experienced modeller. It is well known that other software, e.g. Spice for circuit simulation also needs sometimes some small smart model modifications to run a simulation successfully.

The problem of solving mode-dependent DAE Systems of varying index is discussed in the framework of the Modelica language and the OpenModelica software by considering two small example circuits.

The presented approach may be extended by lifting the simplifying assumption that there are no storage elements in derivative causality independent of any switch state changes. Moreover, an extension to mode switching systems with some nonlinear elements may be considered.

ACKNOWLEDGEMENTS

The author would like to thank the anonymous reviewers for their comments and for making me aware of some minor typos in some formulae.

REFERENCES

Asher G (1993) The Robust Modelling of Variable Topology Circuits Using Bond Graphs. In: Granda J, Cellier F (eds) International Conference on Bond Graph Modeling, ICBGM'93, Proc. of the 1993 Western Simulation Multiconference, SCS Publishing, pp 126–131, Simulation Series, volume 25, no. 2, ISBN: 1-56555-019-6

Borutzky W (2010) Bond Graph Methodology – Development and Analysis of Multidisciplinary Dynamic System Models. Springer-Verlag, London, UK, ISBN : 978-1-84882-881-0

Borutzky W (2015) Bond Graph Model-based Fault Diagnosis of Hybrid Systems. Springer International Publishing Switzerland

Borutzky W (ed) (2016) Bond Graphs for Modelling, Control and Fault Diagnosis of Engineering Systems, 2nd edn. Springer International Publishing Switzerland, DOI 10.1007/978-3-319-47434-2

Buisson J (1993) Analysis and Characterisation of Hybrid Systems with Bond Graphs. In: 1993 IEEE International Conference on Systems, Man and Cybernetics, vol 1, pp 264–269

Buisson J, Cormerais H, Richard PY (2002) Analysis of the bond graph model of hybrid physical systems with ideal switches. Proc of the Institution of Mechanical Engineers Part I: Systems and Control Engineering 216(1):47–63

Cellier FE, Kofman E (2006) Continuous System Simulation. Springer-Verlag

Cellier FE, Krebs M (2007) Analysis and Simulation of Variable Structure Systems Using Bond Graphs and Inline Integration. In: Proc. ICBGM 07, 8th SCS Intl. Conf. on Bond Graph Modeling and Simulation, San Diego, CA, USA, pp 29–34

Dassault Systèmes (2015) URL `http://www.Dymola.com`

Ducreux J, Dauphin-Tanguy G, Rombaut C (1993) Bond Graph Modelling of Commutation Phenomena in Power Electronic Circuits. In: Granda J, Cellier F (eds) International Conference on Bond Graph Modeling, ICBGM'93, Proc. of the 1993 Western Simulation Multiconference, SCS Publishing, San Diego, CA 92177 U.S.A., pp 132–136, Simulation Series, volume 25, no. 2, ISBN: 1-56555-019-6

Elmqvist H, Mattsson S, Otter M (2014) Modelica extensions for Multi-Mode DAE Systems. In: Proceedings of the 10th International Modelica Conference, Lund, Sweden, pp 183 – 193

Fritzson P (2004) Principles of Object-Oriented Modeling and Simulation with Modelica 2.1. Wiley & Sons, ISBN: 0-471-47163-1

Margetts R (2013) Modelling & analysis of hybrid dynamic systems using a bond graph approach. PhD thesis, University of Bath, UK

Mattsson S, Otter M, Elmqvist H (2015) Multi-mode dae systems with varying index. In: Proceedings of the 11th International Modelica Conference, Versailles, France, pp 89 – 98

Mosterman P (1997) Hybrid Dynamic Systems: A hybrid bond graph modeling paradigm and its application in diagnosis. PhD thesis, Vanderbilt University, Nashville, TN, USA

OpenModelica Consortium (n.d.) OpenModelica. URL `https://www.openmodelica.org/`

Strömberg J, Top J, Södermann U (1993) Variable Causality in Bond Graphs Caused by Discrete Effects. In: Granda J, Cellier F (eds) International Conference on Bond Graph Modeling, ICBGM'93, Proc. of the 1993 Western Simulation Multiconference, SCS Publishing, pp 115–119, Simulation Series, volume 25, no. 2, ISBN: 1-56555-019-6

Zimmer D (2013) A new framework for the simulation of equation-based models with variable structure. SIMULATION 89(8):935–963

DEVELOPMENT OF A RULES-BASED FUZZY ALGORITHM FOR DEVIATION OF OBSTACLES

DIAS, Lucas Alves[a]; SILVA, Roger William de Oliveira[b]; EMANUEL, Paulo César da Silva[c];
FERRUS FILHO, André[d]; BENTO, Rodrigo Teixeira[e]

[a], [b], [c], [d], [e] Faculdade de Tecnologia Termomecanica – FTT, Alvarengas Road, 4.001, São Bernardo do Campo – São Paulo – Brazil

[a] lucasalves61@hotmail.com, [b] roger.wo1995@gmail.com, [c] pro8808@cefsa.edu.br,
[d] pro1724@cefsa.edu.br, [e] rodrigo.bento@ipen.br

ABSTRACT

The objective of this project is to introduce the study of the Fuzzy logic, presenting the development of a navigation system in MATLAB software and its implementation in an autonomous mobile vehicle. The application proposal aims to integrate an Automated Guided Vehicles gifted a navigation system elaborated from Set Theory Fuzzy, in any kind of plan environment with obstacles at random points, so that it is able to identify any obstacles present in the environment in which is inserted and through the developed programming language C based on Fuzzy logic implemented in a ARDUINO MEGA2560 plate, calculating the deviation and executing necessary to avoid a collision with any obstacle encountered in the environment. At the end of deployment tests were conducted to obtain the robot responses and compared with the simulation software, according to the possibilities of reading of the sensors. The responses were satisfactory, following the same trend behavior of the simulations of the Fuzzy controller.

Keywords: Fuzzy Logic, Mobile robot, Autonomous navigation, Artificial Intelligence.

1. INTRODUCTION

One of the characteristics of human thinking is dealing with frequent situations that involve factors such as uncertainty, ambiguity or lack of information for its resolution, for example, where it is necessary to use skills such as experiences, knowledge, intuition and even common sense to manipulate such situations. According to Pasternak (2015), Khatchatourian & Treter (2010) and Reis, Dayr & Pati (2000), Artificial Intelligence (AI) is a computer system performing as a substitute for intelligent functions of human beings

This technology involves using methods based on the intelligent behavior of humans and other animals to solve complex problems, representing characteristics that are associated with intelligence in human behavior and the ways in which nature adapts to imbalances (COPPIN 2010).

The Fuzzy Logic, one of the techniques of AI, transforms the verbal, imprecise or qualitative expressions into numerical values and has wide application in modeling of several systems (Khatchatourian & Treter 2010; Zadeh 1990; Shaw & Simões 1999). This system is known as nebulous or diffuse logic and, as stated by Gomide & Gudwin (1994) and Faria & Romero (2002), has been elaborated in order to solve the problems of imprecision Mathematics, supporting the modes of reasoning that are close to the human, but not exact.

Cox (1994) affirm that while the Boolean Logic proposes that these values be true or false, Fuzzy Logic proposes that this is a matter of degree, making possible the use of approximation and even the inference of something that is necessary.

There is a large number of research on the selection problem within a fixed number of alternatives targeted to specific applications.

Lacevic & Velagic (2011) developed a fuzzy logic based position controller whose membership functions are tuned by genetic algorithm. The main goal is to ensure successful velocity and position trajectories tracking between the mobile robot and the virtual reference cart. Simulation results indicate good performance of position tracking while at the same time a substantial reduction of the control torques is achieved.

Shayestegan et al. (2013) demonstrated a hybrid control methodology for mobile robot navigation of a two-wheeled non-holonomic mobile robot by using fuzzy logic controller (FLC) incorporated with a switching command strategy (SCS). Yoo, Kim & Jeong (2012) proposes a fuzzy integral-based gaze control architecture incorporated with the modified-univector field-based navigation for humanoid robots. Using the partial evaluation values and the degree of consideration for criteria, fuzzy integral is applied to each candidate gaze direction for global evaluation.

Omrane, Masmoudi & Masmoudi (2016) designed and implemented a trajectory tracking controller using fuzzy logic for mobile robot to navigate in indoor environments. Simulation results show the performances of the intelligent navigation algorithms in terms of simulation times and travelled path.

Within this context, this paper propose to develop a Automatic Guided Vehicle (AGV) controlled by Fuzzy Logic, able to move in a known flat environment, providing it with a navigation system that identifies random obstacles in its route so that it is able to calculate and execute the deviations that are necessary to avoid its collision. The parameters of the system were elaborated in MatLab software, based on the proposal of a Fuzzy controller based on the literature.

2. EXPERIMENTAL

2.1. Materials

The robot developed in the study must move forward always and make the necessary deviation to avoid its collision when an obstacle is detected by one of the three ultrasonic sensors arranged at 90º in the front part of the prototype. Its deviation must attend to the rule base created to execute the inference strategically in the Fuzzy controller.

According to Corradini & Orlando (1997) and Yang, Li & Huang (2016), for the decision making of the robot it is necessary that it can detect the positions of the obstacles where the necessary deviations must be made to avoid their collision. In view of the required data and the imposed application, it was decided to use three ultrasonic sensors (S1; S2; S3), model HC-SR04, arranged at 90º, with a range of 2cm to 4m, an effect angle of 15º and a 5V supply in direct current. This sensors easily calculate the distance from the obstacle in relation to the emitter and present good accuracy.

Two motors DC PM Gear Motor, model 37JB6K/3530, were used to drive the two wheels used in the robot. In order to allow the connectivity of the respective motors with a microcontroller to control them, the electronic circuit H-bridge was used, being able to provide the current or voltage necessary for the operation of the motors and also provides an easier way for the possibility of changing the direction of rotation of the motor, only the polarity of the terminals being necessary. The model used in the project was the L298N, which presents in its structure two H-bridges, controlled via the Arduino platform and the C language program, which allows controlling the direction and speed of the motors through the pulse width modulation in the Pulse Width Modulation (PWM) of the board.

The Figure 1 allows to visualize the final mechanical structure of the mobile robot developed, emphasizing the need of the application of two motor wheels and a free wheel to maintain the stability of its structure in relation to the surface.

The Figure 2 presents a diagram of the mobile robot with the detail of the hardware used.


Figure 1: Mechanical structure of the robot.

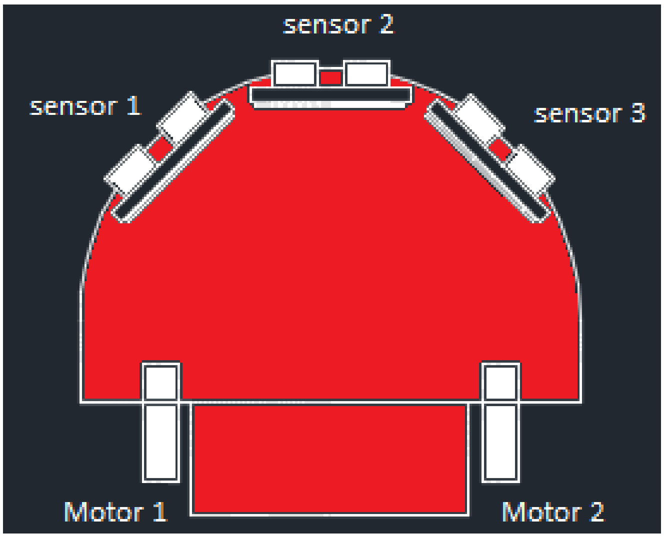


Figure 2: Diagram of the mobile robot hardware.

2.2. Methods

The Fuzzy Logic is a simple system of modeling by allowing its application through linguistic variables and their answers obtained in a short processing time, adapting to the real time application needful. Its application can be done in numerous areas such as industrial, medical and domestic services, replacing jobs that require dexterity, safety and precision (Gomide and Gudwin 1994; Carvalho, Yamakami and Bonfim 2013; Corradini and Orlando 1997).

The basic purpose of a Fuzzy control is to model actions based on expert knowledge, rather than modeling the process itself. According to Takagi & Sugeno (1985), even with the different methods found, controllers can be grouped into two large groups:

- Mandani controller: based on the Fuzzy Implication Functions in Composition Operators for the Fuzzy Output Definition of the Controller;

- Sugeno controller: simplification of the Mamdani controller, dispense with the definition of implication functions and operators for inference.

The Mandani controller the quantitative values into qualitative ones (Fuzzy) and then, in other values also qualitative through inference, after the Defuzzifier is used for the final quantitative response. These controllers are easy to model because they are based on intuition and are good when a rough control is acceptable, and for thinner systems the Sugeno controller presents superior performance, but with a mathematical modeling through the functions of the input language variables (Takagi and Sugeno 1985).

For the implementation of the study an Arduino MEGA 2560 board was used, due to its specifications that successfully meet the needs of the project.

For the virtual evaluation and modeling of the proposed solutions, MatLab was used to construct the Fuzzy controller of the robot navigation system, with the help of the Fuzzy Logic Toolbox.

The system developed for the navigation control was created in such a way that the robot was able to detect the obstacles present in its trajectory, to judge the conditions of the environment around and make decisions through the performance of the engines.

In the Fuzzy system, developed through MatLab, according to Figure 3, three input variables (S1; S2; S3) were determined, relative to the input values of the left, center and right ultrasonic sensors, respectively, located on the front of the robot And two output variables (Left Wheel; Right Wheel) referring to the PWM values that will be assigned, respectively, to the left and right motors.

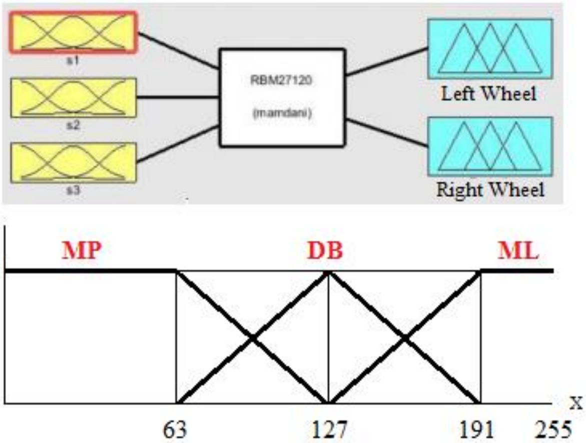


Figure 3: Pertinence functions used in Fuzzy Robot System.

Three inputs of data were used in this study, each one originated by an ultrasonic sensor considering the distance of the sensor to some object. For each input, three pertinence functions were assigned, according to the distance range: MP (Very Close); DB (Good Distance); and ML (Very Far).

Considering the quantities of pertinence functions present in the controller were assigned 27 rules, referring to the possibilities of combining the variables of each entry.

To define the rules, the positions of the sensors and the situations in which the robot can be submitted has been considered, in order to avoid its collision with any obstacle present in the operating environment.

The speed control actions of the robot has been established, working with the PWM values for the two motors, such as: VP (Positive Velocity); V0 (Average Speed) and VN (Negative Speed).

By using the inference method asked for the system, the following reading values were obtained from the sensors to define the pertinence functions: reading values less than 63 considered as Very Close Distance (MP), higher than 191 considered as Very Far Distance (ML) and from 63 to 191 as Good Distance (DB) considering the pertinence degrees of each linguistic value superior to 0.5 to be stored for use in the inference. The robot will move forward when the obstacle is too far away or when it picks up an obstacle on the side sensors (DB or MP) and the control unit is reading an obstacle too far (ML). If there is an obstacle ahead, the robot should deflect to the side where the sensor identifies the longest distance to it. When the two side’s sensors are the same and have no distance ML, it has been established that the curve is made to the left.

Defuzzificador present in the robot navigation system is responsible for converting the fuzzy controller outputs inferred into traditional values, which in this case will be the outputs with the PWM value that will give the desired motions to the motors. The centroid method was chosen as the method for Defuzzification of the controller, where the final output value of the system is given by the position of the centroid of the generated geometry in the output Fuzzy sets from the input inference (Simonsen 2011).

In summary, the first stage is responsible for the communication of the robot with the environment, where it is made the reading and storage of the digital values measured by the ultrasonic sensors, these data are converted to the Fuzzy set of the system (0 to 255) and inserted in the fuzzifiers of the controller where they are transformed into linguistic variables according to the coding performed in degrees of pertinence. Each Fuzzifier has three outputs for each of the pertinence functions. The coded values are evaluated according to the defined rules, in this step the pertinence functions (MP; DB; ML) are stored, according to the inference method, of each sensor to choose the rule to be followed, resulting in a Language value of output (VN, V0; VP). In Defuzzification these values are, again, converted into the range 0 to 255 and sent to the PWM outputs of the board defined in the programming, corresponding to the speed that each motor must act.

Figure 4 shows the diagram of the electrical scheme used for the development of the project, with the connections used for sensor communication, H bridge, DC motors and the Arduino ATmega2560 board.

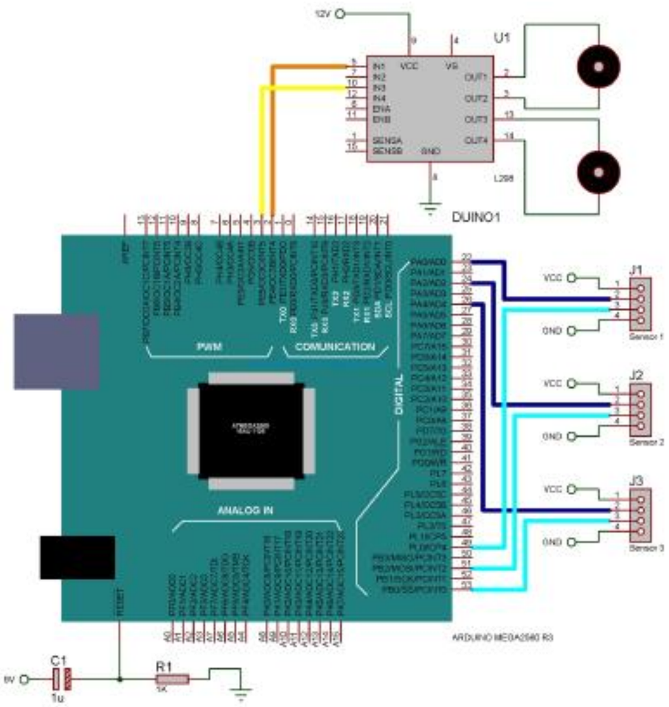


Figure 4: Electrical scheme of the project.

3. RESULTS AND DISCUSSION

Below are presented the results obtained by individually analyzing an input and an output.

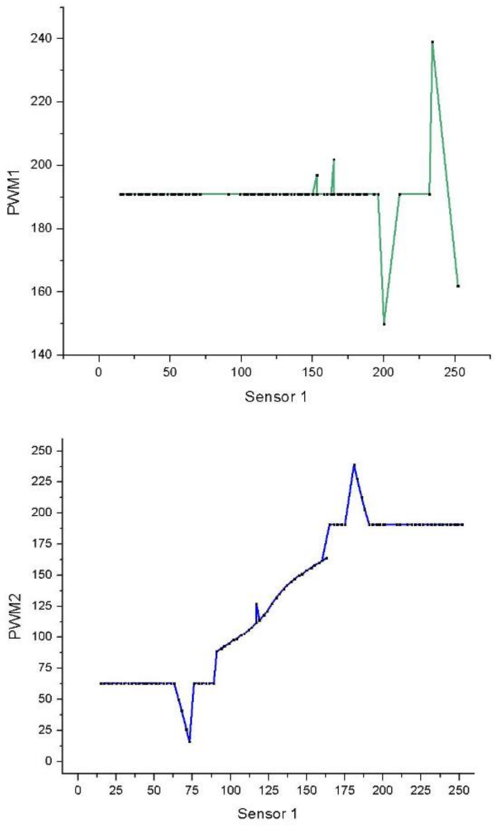


Figure 5: Sensor 1 input (PWM1) and output (PWM2) values.

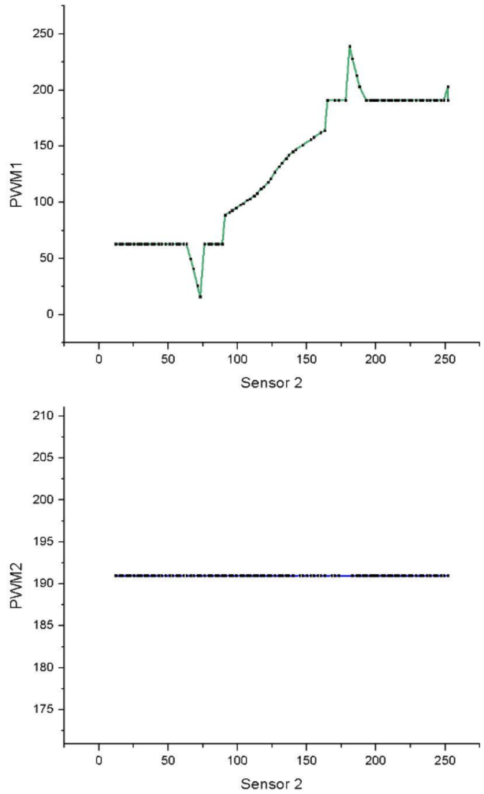


Figure 6: Sensor 2 input (PWM1) and output (PWM2) values.

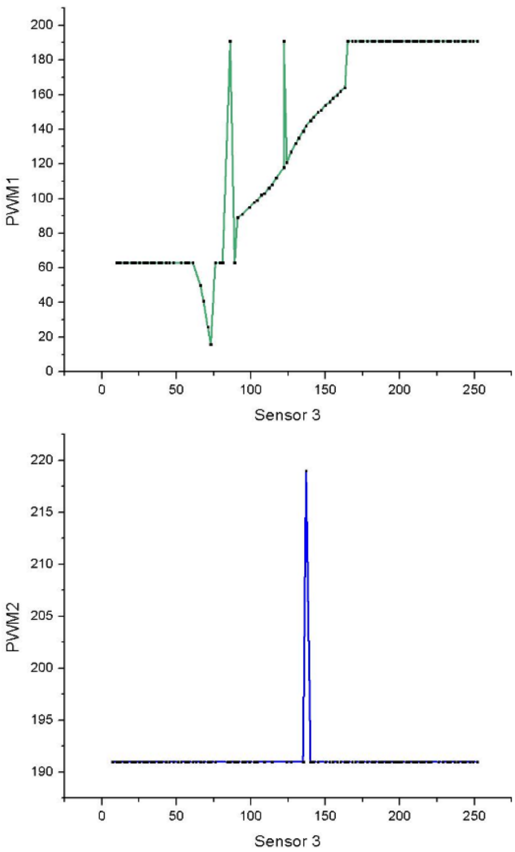


Figure 7: Sensor 3 input (PWM1) and output (PWM2) values.

The output of PWM1 and PWM2 are obtained through the Fuzzification process, inference based on Rules and Defuzzification. In this test, the PWM outputs of the controller for each sensor were analyzed individually, considering that the sensors not used in the test had a non-tangible margin of inference to the process, so that there was no interference of unused sensors in the results obtained.

The respective values were obtained due to the pertinence of each sensor used in the movement of the AGV. The linearity found in Sensor 2 (Figure 6) occurred because of its greater sensitivity, during the change of trajectory, and the tendency of curve. To validate the software it is necessary to compare the results of the sensors with the MatLab simulation.

The first tests were performed on the MatLab controller simulator to evaluate the basic operation of the parameters defined in the software, according to the rules base developed, considering the three data inputs that represented the sensors of the robot and its outputs, so much the input values as output values were defined for a set universe with a range from 0 to 255. Taking into account the curve time of the robot and the range of action of the ultrasonic sensors used in the study, the rules were formulated so that the curves could occur with the greatest difference in the speed of each engine and its actions with a longer distance from the present obstacle.

Analyzing the graphs, the developed in C language for the robot were satisfactory. Other methods would require more computational resources, whose application is seen as lacking in resources. As a consequence of this ness, an attempt was made to find an alternative for the development of the controller, since the simplicity of the system would imply directly in the controller response. The programming language should be low-level, requiring less resources of the physical controller, making the use of Language C the most possible programming mode for the system, compared to other commonly known codes.

In the initial tests with the software were obtained the results of the defuzzified values presented in Table 1, the input values were chosen according to the probability of reading of the three sensors with three functions of each pertinence.

Table 1: MatLab Simulator Results

RULES	S1	S2	S3	LEFT WHEEL	RIGHT WHEEL
1	51	51	51	48,6	206
2	51	51	127	206	48,6
3	51	51	200	206	48,6
4	51	127	51	48,6	206
5	51	127	127	206	48,6
6	51	127	200	206	48,6
7	51	200	51	206	206
8	51	200	127	206	48,6
9	51	200	200	206	48,6
10	127	51	51	48,6	206
11	127	51	127	48,6	206
12	127	51	200	206	48,6
13	127	127	51	48,6	206
14	127	127	127	48,6	206
15	127	127	200	206	127
16	127	200	51	48,6	206
17	127	200	127	206	206
18	127	200	200	206	127
19	200	51	51	48,6	206
20	200	51	127	48,6	206
21	200	51	200	48,6	206
22	200	127	51	48,6	206
23	200	127	127	48,6	206
24	200	127	200	127	206
25	200	200	51	48,6	206
26	200	200	127	127	206
27	200	200	200	206	206

The output values presented in the simulator correctly followed the expected behavior in each controller rule.

After the initial tests, the answers were analyzed with the final algorithm developed, already implemented on the Arduino board with the three ultrasonic sensors connected to provide the input values of the controller (0 to 255). The output values responsible for the movement of the motors are were obtained through the PWM pins of the board which have also been configured to operate in the range of 0 to 255.

The test preparation methodology aimed to obtain the output values in the PWM for each rule produced in the Fuzzy controller. For this evaluation to be carried out, the board with, the algorithm was connected with serial communication to a computer and analyzed through the serial monitor in the software of the Arduino platform, configured through the program created to present in real time the reading value of the sensors and the values present on the PWM pins of the board.

The sensors were placed at a 90º angle between them, so that obstacles could be positioned in each sensor at the desired distance without interfering with the other sensors. Objects were allocated at distances that presented the values of reading of the sensors according to the linguistic variables of the controller, thus, all rules could be tested by varying the distances of the objects in relation to the sensors with the necessary combinations.

Table 2 presents the PWM readings of the plate according to the input values, manipulated through the distance of the objects where: input values 51 belong to the linguistic value MP; 127 to the linguistic value DB; and 200 to the linguistic value ML.

Table 2: Robot Results

RULES	S1	S2	S3	PWM1	PWM2
1	51	51	51	63	191
2	51	51	127	191	63
3	51	51	200	191	63
4	51	127	51	63	191
5	51	127	127	191	63
6	51	127	200	191	63
7	51	200	51	191	191
8	51	200	127	191	63
9	51	200	200	191	63
10	127	51	51	63	191
11	127	51	127	63	191
12	127	51	200	191	63
13	127	127	51	63	191
14	127	127	127	63	191
15	127	127	200	191	127
16	127	200	51	63	191
17	127	200	127	191	191
18	127	200	200	191	127
19	200	51	51	63	191
20	200	51	127	63	191
21	200	51	200	63	191
22	200	127	51	63	191
23	200	127	127	63	191
24	200	127	200	127	191
25	200	200	51	63	191
26	200	200	127	127	191
27	200	200	200	191	191

The results of the robot obtained in bench were also satisfactory, which led to the beginning of the tests with displacement of the robot of autonomous form, aiming to observe its displacement in real situations of application.

As a way of validating the results obtained in the algorithm implemented in the robot, comparative graphs were drawn for each controller output, which could show the behavior of the results obtained for the two wheels in MatLab and in the robot according to the same input values. The Figure 6 and Figure 7 show the comparison charts for the two wheels.

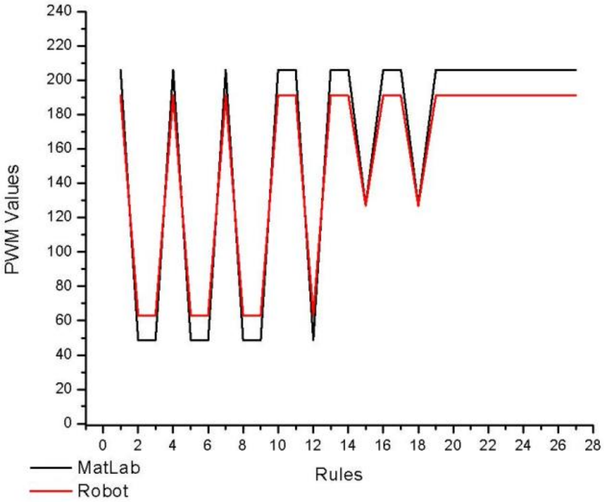


Figure 7: Comparison of the Right Wheel in MatLab and in the Robot.

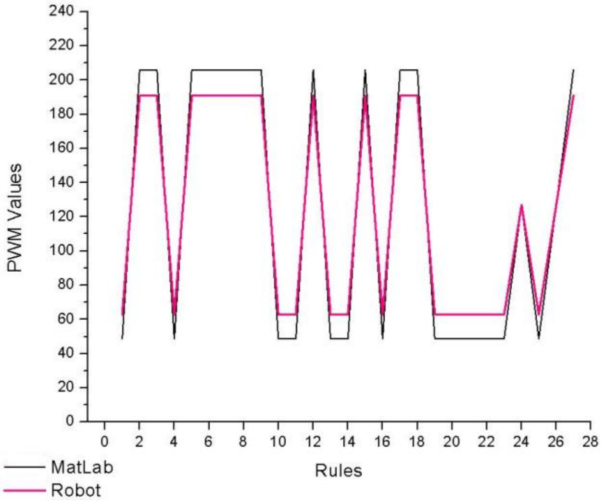


Figure 8: Comparison of the Left Wheel in MatLab and in the Robot.

The behavior of the graphs shows that the results of the algorithm developed in C language for the robot were satisfactory, following the same performance trends according to the situations to which they were submitted. However, the differences identified were the maximum and minimum PWM values. In the results obtained in tests with the robot, the output values for both motors were shown to be lower, at the maximum speed, according to the controller simulation in MatLab and, at minimum speed, presented higher output values than those obtained in simulation.

4 CONCLUSIONS

With the algorithm in operation, it was verified that the Fuzzy logic offers a smoother transition in the movements. In order to validate the efficiency of the navigation system created, simulations were performed with the robot according to the rules inserted in the Fuzzy controller, where the input values of the sensors and the output values in the PWM of the board were analyzed. The results obtained were consistent with the responses given by the simulation in MatLab, following the same trend of behavior. In comparison to the proposed objectives and the desired behavior obtained in virtual simulation, the Fuzzy controller used in the study, in an intelligent navigation system, is feasible for such an application, for presenting coherent answers in the tests made, being able to deviate from the present obstacles.

REFERENCES

Carvalho, M. B.; Yamakami, A.; Bonfim, T. R. Algoritmo híbrido para resolver o problema de escalonamento job shop com incertezas. Tema Tendências em Matemática Aplicada e Computacional. Brazilian Society for Computational and Applied Mathematics (SBMAC), 2013, p.43-55.

Coppin, B. Inteligência Artificial. Rio de Janeiro: LTC, 2010.

Corradini, M. I.; Orlando, G. A. A discrete adaptive variable-structure controller for MIMO systems, and its application to an underwater ROV. Institute of Electrical & Electronics Engineers (IEEE) Transactions On Control Systems Technology, [s.l.], v. 5, n. 3, 1997, p.349-359.

Cox, E. The fuzzy systems handbook: a practitioner’s guide to building, using, and maintaining fuzzy systems. New York: AP Professional, 1994.

Faria, Gedson and Romero, Roseli A. Francelin Navegação de robôs móveis utilizando aprendizado por reforço e lógica fuzzi. Sba Controle & Automação, Set 2002, vol.13, no.3, p.219-230.

Gomide F. A. C.; Gudwin. R. R. Modelagem, controle, sistemas e lógica Fuzzy”. SBA Controle e Automação, Campinas, v. 4, n. 3, 1994, p.97-115.

Khatchatourian, Oleg and Treter, Jaciara Aplicação da lógica fuzzy para avaliação econômico-financeira de cooperativas de produção. JISTEM J.Inf.Syst. Technol. Manag. (Online), 2010, vol. 7, n.1, p. 141-162.

Lacevic, B.; Velagic, J. Journal of Intelligent & Robotic Systems, 2011, Vol. 63(3), p.595-614.

Omrane, H.; Masmoudi, M. S.; Masmoudi, M. K.; Dong, W. Computational Intelligence and Neuroscience, 2016, p. 10.

Pasternak, Jacyr. Robot-assisted needle insertion for venous catheterization. Einstein (São Paulo), Sept 2015, vol.13, no.3, p.475-476.

Reis, Dayr and Pati, Niranjan Applications of artificial intelligence to condition-based maintenance. Rev. adm. empres., June 2000, vol.40, no.2, p.102-107.

Sandri, S.; Correa, C. Lógica Nebulosa. Escola de RN, Julho/1999, ITA – São José dos Campos/SP, 1999.

Shaw, I. S., & Simões, M. G. Controle e Modelagem Fuzzy. São Paulo: Editora Edgard Blucher Ltda, 1999.

Shayestegan, M.; Marhaban, M.; Shafie, S.; Sattar, B D.; Abdul, S. M. International Journal of Robotics & Automation, 2013, Vol. 28(4), p.1.

Silva, R. A. Inteligência Artificial Aplicada a Ambientes de Engenharia de Software: Uma Visão Geral. INFOCOMP. Viçosa, 2005, p. 1 -10.

Simonsen, G. D. M. Resgate utilizando robô real. São Paulo, 2011.

Takagi, T.; Sugeno, M. Fuzzy identification of systems and its applications to modeling and control. Institute of Electrical & Electronics Engineers (IEEE) Transactions On Systems, Man, And Cybernetics, 1985, 15(1), p.116-132.

Yang, S.; Li, C.; Huang, T. Impulsive synchronization for TS fuzzy model of Memristor-based chaotic systems with parameter mismatches. International Journal of Control, Automation and Systems, 2016, 14(3): 854-864.

Yoo, J. K.; Kim, J. H. Y., Jeong, K. IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics, Feb 2012, Vol. 42(1), p.125-139.

Zadeh, L. A. Fuzzy sets and systems. International Journal of General Systems, 1990, 17: 129-138.

AUTHORS BIOGRAPHY

Lucas Alves Dias is Technologist trained by Faculdade de Tecnologia Termomecanica – FTT (Brazil) in 2015. Dias has experience in Basic Electricity, Robotics, Mechatronics and has recently worked in the following development of Fuzzy Logic, Autonomous navigation, Mobile robot and Artificial Intelligence.

Roger Wiiliam de Oliveira Silva is Technologist trained by Faculdade de Tecnologia Termomecanica – FTT (Brazil) in 2015. He has experience in Basic Electricity, Robotics, Mechatronics and has recently worked in the following development of Fuzzy Logic, Autonomous navigation and Mobile robot.

Paulo César da Silva Emanuel is Professor of Automation and Control Engineering and Food Engineering, Faculdade de Tecnologia Termomecanica – FTT (Brazil). Emanuel received the M.S. and Dr. of Biomedical Engineering in 2013 and 2015 from Universidade de Mogi das Cruzes – UMC (Brazil). He is coordinator of Engineering Laboratories at Faculdade ENIAC (Brazil). He has experience in Basic Electricity, Analog Electronics, Digital and Microprocessed Systems, Microcontrolled Devices and Basic Computing.

André Ferrus Filho is Professor of Automation and Control Engineering, Faculdade de Tecnologia Termomecanica – FTT (Brazil), and Professor of Engineering in Faculdade das Américas – FAM (Brazil). He received the M.S. of Biomedical Engineering in 2012 from Universidade de Mogi das Cruzes – UMC (Brazil). Ferrus Filho works as a Technical Consultant of Perfect do Brasil in the area of Research, Quality and Development of New Products and he has experience in Robotics, Mechatronics, Automation and Metallurgy, with emphasis on CNC, CAM and Machining Practices.

Rodrigo Teixeira Bento is Technologist trained by Faculdade de Tecnologia Termomecanica – FTT (Brazil) in 2015. Currently, he studies Master’s degree in Nuclear and Energy Research Institute (IPEN-CNEN/SP – Brazil). Bento has recently worked in the area of Materials and Metallurgical Engineering, in the following development topics: synthesis and characterization of TiO_2 films, semiconductor photocatalysis, Thermochemical treatment and Structure Analysis.

WELL-FORMED PETRI NET BASED PATTERNS FOR MODELING LOGIC CONTROLLERS FOR AUTONOMOUS TRAINS

Yuchen Xie, Manel Khlif-Bouassida, Armand Toguyéni

Centrale Lille, CRIStAL, UMR 9189
59650 Villeneuve d'Ascq, France
Univ. Lille Nord de France, F-59650, Lille, France

{yuchen.xie, manel.khlif-bouassida, armand.toguyeni}@centralelille.fr

ABSTRACT

The automation and the adoption of ERTMS (The European Rail Traffic Management System) are two solutions for railway systems to satisfy the necessity of increasing the capacity of railway lines and enhancing their safety. In this context, this study is to be part of the contribution to a methodology allowing the development of discrete event controllers of autonomous train control system needed for railway automation. This article emphasizes the modeling stage using Colored Petri Nets (CPN) and its extensions. While modeling, both the railway requirements and the necessity to formally verify some crucial properties (e.g., collision-free system) have been taken into account. By proposing several modeling patterns based on Well-Formed Petri Nets (WFN), we solve several technical problems of modeling railway train control system and similar complex systems, making it possible to construct reducible and analyzable models, before being formally verified.

Keywords: Railway System, Autonomous Train Control, DES Modeling, Colored Petri Nets

1. INTRODUCTION

The development of autonomous trains logic controllers has become a priority in railway control system, in order to increase its safety, and to make railway system more competitive with regard to the other means of transportation.

This study is part of a methodology allowing the systematic and rigorous development of logic controllers necessary for railway automation. The methodology we finally develop should formally model the control functions and make it possible to verify essential properties (e.g., safety) of an automated system. The ultimate goal of this methodology is to generate, by model transformation, the code of these functions, which could be implemented on the computers of the ground infrastructure and on the embedded controllers in the trains. This paper concerns the modeling stage. One problematic situation is to deal with the compromise between the modeling power of the selected modeling approaches and the possibility of making formal verification. In this study, we decide to use Well-Formed Petri Nets (WFN) as the modeling formalism to benefit from all its advantages, and we propose several WFN modeling patterns and techniques suitable with the complexity of railway systems.

The paper is structured as follows. In the second section, we give a state-of-the-art of railway system modeling using Colored Petri Net (CPN). In the third section, we discuss the tradeoff between the modeling power and the verification capacity of CPN and WFN approaches, in order to finally justify the reason why we choose WFN as our modeling tool for autonomous train control system. Section 4 presents a railway system structure and some main functions used in this paper; several constrains and assumptions in the modeling stage are also given in section 4. In section 5, we present a brief introduction about the main formalism and characteristics of WFN. In section 6, we propose certain modeling patterns as solutions to some main modeling problems of a railway control system. In section 7, we illustrate the use of these patterns by a case study of automation of railway system. Finally, we end up with conclusions and perspectives of this work in section 8.

2. STATE OF THE ART

For about half a century, Petri Nets (PN) have been used to model concurrent and complex systems. Among its numerous extensions, CPN is the most widely used formalism incorporating data, hierarchy, and time (van der Aalst et al. 2013). This section summarizes some research works using CPN to model and analyze railway control system.

In (Janczura 1999), a whole process of modeling and analyzing a railway network is proposed using CPN. The network considers two types of trains (i.e., express and normal) which move in the same direction. A safety property (each block in the railway line can only be occupied by exactly one train or empty) and four operational properties are analyzed. However, this thesis report only considers a quite simple model and does not respect the ERTMS/ETCS standard.

In (Jansen et al. 1998; Meyer zu Hörste 1999), a CPN hierarchical framework is proposed to model ERTMS/ETCS (mainly in level-2). Several generic modeling paradigms and techniques (e.g., distributed modeling, communication between separate CPNs, synchronization, etc.) are created to build their CPN models and formal methods can be used to analyze these models. This study mainly focus on the hierarchical and structural problems of modeling ETCS specifications instead of the implementation of concrete functional models.

A summary of Petri Nets models of railway stations is given in (Žarnay 2004). Different models are divided into four levels (i.e., technical equipment level, movement level, train processing level and decision-making level) according to their different objectives and different abstract levels.

CPN Tools is a tool for editing, simulating and analyzing CPNs, which is first introduced in (Ratzer et al. 2003; Jensen et al. 2007). After its wide application, more CPN models are proposed by benefiting from the features of this powerful tool.

In (van der Aalst et al. 2013), several CPN design patterns and strategies are proposed using CPN Tools, showing some solutions to several typical design problems in terms of modeling complex processes.

In (Vanit-Anunchai 2009; Vanit-Anunchai 2010; Vanit-Anunchai 2014), railway interlocking table models are proposed using CPN Tools. As a main advantage of these models, the general CPN structure proposed can be reused regardless of variable structures and sizes of railway systems. While, these models store too many data (e.g., geographic information) in colored tokens and the behavior of the models are greatly affected by the data instead of the structure of model. In this case, although we can do some test with the simulation function in CPN Tools, a formal verification is rather difficult to performed on this kind of CPN models.

In our previous work (Xie et al. 2016), we have proposed discrete controller models based on High-level Petri Nets supported by CPN Tools. Our research concentrates on the whole railway system (i.e., both the train controller and the trackside part). In this previous work, these High-level Petri Nets are unconstrained CPNs, allowing a powerful modeling ability of the systems. CPN Tools also offers extra extensions with ML Language to enhance its modeling power, e.g., the possibility of defining a “list” datatype. However, one pays for this capacity of modeling because there are no really, tools or efficient methods, to help analyze and verify these unconstrained CPNs with extensions.

3. COMPARISON OF CPN AND WFN APPROACHES OF MODELING TRAIN CONTROL SYSTEMS

Figure 1 compares the CPN and WFN approaches of modeling the railway system and the possible analyzing methods applicable to the models.

As shown in the left part of Figure 1, the most direct way of analyzing a CPN is to generate its reachability graph which enables to check the required properties. Although it is possible in theory, the application of this method is generally limited by combinatorial explosion problems.

One way to combat this combinatorial explosion would be to reduce the initial CPN model before constructing the reachability graph. However, there is very little work proposing reduction rules applicable to CPNs and each of them has their own applicable constrains. For example, the reduction rules proposed in (Esparza & Hoffmann 2016) are only applicable to free choice workflow nets with an objective of maintaining the soundness property.

In more general cases, the only solution available is to first go through an unfolding operation of the CPN before the application of some existing reduction rules of ordinary PNs (Berthelot & Lri-Iie 1986; Murata 1989). However, in this case, for a complex system e.g. railway system, one is confronted with a combinatorial explosion problem in the unfolding operation.

To solve the problems above, some High-level Petri Nets with constrains are proposed, among which the Well-Formed Petri Nets (Chiola et al. 1991) are of interest to us. It is proved that WFN have the same expression power as CPN (Diaz 2013), which means, every CPN can be transformed into a WFN with the same basic structure, same color domains (possibly partitioned in static subclasses), equivalent arc labeling, and the possible addition of transition predicates (Chiola et al. 1991). It implies that WFNs have at least an equal modeling power compared to general (unconstrained) CPNs defined in (Jensen 1981).

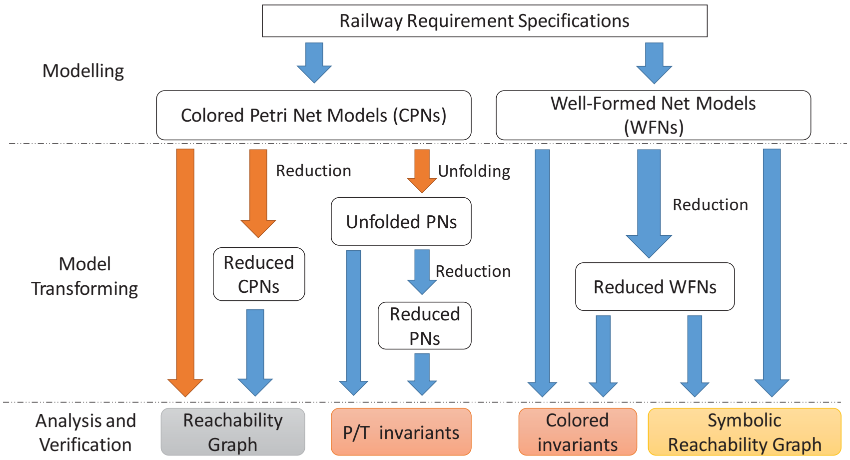


Figure 1 Different Ways of Analysis of Colored Nets

The right part of Figure 1 describes the possible analysis methods applicable to WFN models. To avoid the combinatorial explosion, several reduction rules could first be applied, e.g. the reduction rules in (Haddad 1991). Several reduction rules may also need a calculation of colored invariants (Couvreur & Martínez 1991). Then the models can be analyzed with help of these invariants and/or by building a symbolic reachability graph (Chiola et al. 1991) which will greatly reduce the size of the reachability graph.
As our final objective is to propose appropriate Petri Net patterns whose properties can be checked before the models are implemented, this paper proposes the use of WFN instead of CPN for modeling autonomous railway control systems, in order to benefit from the advantages of analyzing a WFN model.

4. RAILWAY SYSTEM BASIC AND CONTEXT

This study concerns the management of multiple trains in a railway line based on Movement Authorities (MA, permission for a train to move to a specific location with supervision of its speed) generated by trackside infrastructure. We first present the background of the railway models.

4.1. Railway Lines and Blocks

Railway lines are connections of different railway stations. Normally, a single railway line has a fixed direction and all the trains in this railway line run in this direction. A railway line is divided into numerous blocks. Blocks are used to avoid train collisions, ensuring the safe and efficient operations of railway systems.

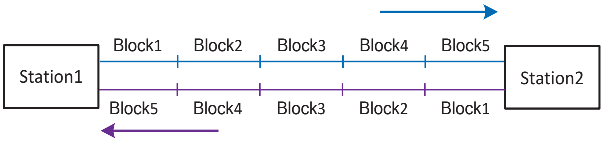


Figure 2 Railway Lines and Blocks

Figure 2 shows an example of lines decomposed into blocks. The railway line from Station 1 to Station 2 and another railway line from Station 2 to Station 1 are divided into several blocks respectively (for simplicity, Figure 2 represents each railway line with 5 blocks). For safety reasons, each block must contain no more than one train. Thus, only after the train occupying the current block (the block is said to be "occupied") has left (the block is then "clear"), another train is authorized to enter this block.

4.2. ETCS-2 Based Train Management in Railway Lines

European railway systems are nowadays equipped with the ERTMS and the European Train Control System (ETCS).
ETCS is specified in four different levels (level 0-3). Currently, ERTMS/ETCS level 2 (ETCS-2) has been put into use on several high-speed railway lines in Europe, which uses Eurobalise to help train locating and uses continuous radio transmission GSM-R (Global System for Mobile Communications - Railway) for data exchanges between trackside infrastructures and onboard equipment. Our study is based on the infrastructure of ETCS-2. Figure 3 illustrates the main functions of train management offered by ETCS-2.

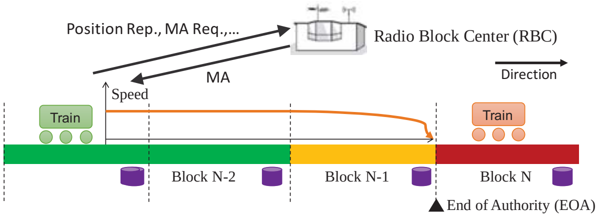


Figure 3 ETCS-2 Based Train Management

Trackside: Radio Block Center (RBC) provides trains with Movement Authorities (MA), taking into account the positions of corresponding trains, signals and switch states as well as the physical line configuration (slopes, curves, etc.);
Onboard: Each train regularly sends its position to RBC and receives MA from RBC. The onboard equipment calculates a speed profile considering the End of Authority (EOA), which is the last block in the MA, and the train characteristics (mass, length, etc.).

4.3. System Simplifications and Assumptions

This study considers a set of simplifying assumptions to manage multiple trains in a railway line. The aim of these simplifications is to reduce the complexity of the models so that the models could be represented in a limited number of pages. The principle assumptions are:

1. Our model does not take into account the length of a train. We only care whether a train occupies a block.
2. The MA message is reduced to the list of blocks that are reserved and assigned to a train. The exact speed limit and the other parameters in a MA are not considered here. However, we assume that a train can always stop at its EOA.
3. A single RBC manages all the trains in the same railway line between two stations. This means that the RBC handover function is beyond this study. The control of railway node/station is not considered.
4. In this paper, a "railway line" has a fixed operation direction and is linked with only 2 stations: the departure and the arrival. The overtaking is not considered. This assumption simplifies the operations we propose later in the paper by always maintaining the same order of the trains as they enter in this railway line.
5. Each time a train enters in a new block, we assume that it receives its current position from a Eurobalise and then sends a position report to the corresponding RBC, instead of considering the specified report format according to ERTMS/ETCS-2 standard.

6. Once a RBC receives a position report from a train, it updates the train's location in its database. The RBC also considers the location report as a MA request. Consequently, it generates a MA response to the train according to the following principle: if the train is the first one in the railway line (there is no preceding train until the end block), its EOA is set to the end block in this railway line, otherwise the EOA is set to the block next to the one occupied by the preceding train.

5. WELL-FORMED PETRI NETS

5.1. Well-Formed Petri Net and its formalism

Well-Formed Petri Nets (WFN) are Colored Petri Nets (Jensen 1981) that satisfy a set of syntactical constraints. In this paper, we only introduce the main features of WFN. A complete formal definition can be found in (Chiola et al. 1991).

5.1.1. WFN Color Classes and Color Domains

A color class can be ordered or unordered, and can be divided into static subclasses. A color class defines the same nature of the tokens of this type. When a color class comprise of several static subclasses, the colors within each static subclass share some similar potential behaviors (batch operation, symmetry, etc.).

A color domain is a Cartesian product of color classes. A neutral color is noted as ε, allowing to define uncolored places or transitions.

Each place and each transition of a WFN is associated with a color class or with a color domain.

5.1.2. WFN Color Functions

Color functions are formal sums of guarded functions built by standard operations (linear combination, composition, etc.) on basic functions.

There are three basic functions: *identity function* is a projection which selects an item of a tuple and is always denoted by a typed variable (e.g. X, Y) in application; *diffusion function* is a constant function which returns the bag composed by all the colors of a class or a subclass and is denoted All(C) where C is the corresponding (sub)class; *successor function* applies on an ordered class and returns the color following the given color, which is denoted as $\oplus$.

5.1.3. Guards

Color functions are formal sums of guarded functions built by standard operations (linear combination, composition, etc.) on basic functions.

An atomic predicate can identify two variables ([X = Y]), compare a variable with another using successor function ($[X = \oplus Y]$), or restrict a variable to be within a static subclass D ([X $\in$ D]).

The constrains above provide WFN with a good structure and simplify its analysis. The formalism of basic functions emphasizes the system symmetries. However, some asymmetric behaviors of objects in a given class are also supported by subclass divisions or by guards on transitions or on color functions, which has strengthened the modeling power of WFN.

5.2. WFN Modeling Tools

CPN-AMI (Kordon & Paviot-Adet 1999) allows users to build and analyze models of AMI-Nets, which are WFNs with a specific syntax.

GreatSPN (Chiola et al. 1995) is a friendly framework allowing the modeling, validation, and performance evaluation of Generalized Stochastic Petri Nets (GSPN) and their colored extension: Stochastic Well-Formed Nets (SWN). This tool also supports timed Petri Net based modeling and implements several efficient analysis algorithms to facilitate complex applications.

Besides these tools supporting WFN, one could also choose from a variety of tools for Colored Petri Nets and High-level Petri Nets to build their WFN models with respect to the WFN definition.

6. WFN MODELING PATTERNS FOR TRAIN CONTROL SYSTEM

In this section, we propose three modeling patterns that could be useful to build WFN models for railway control systems.

1. An equivalent structure in WFN to the arcs using IF-THEN-ELSE expressions defined in CPN Tool (Jensen et al. 2007);
2. The definition and implementation of a successor function;
3. A WFN queue structure with its corresponding management operations (adding item, removing item, modifying item, query, etc.).

We will define these modeling patterns with respect to a practical railway train control model. While these modeling patterns can also be applied to other complex system models.

6.1. IF-THEN-ELSE Arc in WFN

IF-THEN-ELSE is a common alternative structure that facilitates the modeling of some system logic functions. An arc using IF-THEN-ELSE expression is supported by some tools e.g. CPN Tools. Unfortunately, it is not supported in WFN. In this section, we propose two solutions to use IF-THEN-ELSE arc based on guarded functions and guarded transitions respectively.

6.1.1. IF-THEN-ELSE Arc by Guarded Functions

Consider a transition t and a place p. Let $\mathrm{C}(\mathrm{t}) = \mathrm{C}$ and $\mathrm{C}(\mathrm{p}) = C_1 \times C_2 \times \cdots \times C_k$.

We define F_t and F_f two unguarded colored functions, which are sums of tuple of basic functions.

$$F_t = \textstyle\sum_m \langle f_1^t, f_2^t, \dots, f_k^t \rangle,$$
$$F_f = \textstyle\sum_n \langle f_1^f, f_2^f, \dots, f_k^f \rangle.$$

We define a general IF-THEN-ELSE expression which labels an arc connecting a transition t and a place p:

$$\mathrm{W}^*(p,t) = if\ g\ then\ F_t\ else\ \mathrm{F_f}, \text{ where } * \in \{+,-\}.$$

As such an expression $W^*(p,t)$ is not supported by WFN syntax, we define the equivalent function $W_E^*(p,t)$:

$$
\begin{aligned}
\mathrm{W}_{\mathrm{E}}^{*}(\mathrm{p,t}) &= [\mathrm{g}]\, F_t + [\neg \mathrm{g}]\, F_f \\
&= [\mathrm{g}] \sum\nolimits_{\mathrm{m}} \langle \mathrm{f}_1^{\mathrm{t}}, \mathrm{f}_2^{\mathrm{t}}, \ldots, \mathrm{f}_{\mathrm{k}}^{\mathrm{t}} \rangle + [\neg \mathrm{g}] \sum\nolimits_{\mathrm{n}} \langle \mathrm{f}_1^{\mathrm{f}}, \mathrm{f}_2^{\mathrm{f}}, \ldots, \mathrm{f}_{\mathrm{k}}^{\mathrm{f}} \rangle \\
&= \sum\nolimits_{\mathrm{m}} [\mathrm{g}] \langle \mathrm{f}_1^{\mathrm{t}}, \mathrm{f}_2^{\mathrm{t}}, \ldots, \mathrm{f}_{\mathrm{k}}^{\mathrm{t}} \rangle + \sum\nolimits_{\mathrm{n}} [\neg \mathrm{g}] \langle \mathrm{f}_1^{\mathrm{f}}, \mathrm{f}_2^{\mathrm{f}}, \ldots, \mathrm{f}_{\mathrm{k}}^{\mathrm{f}} \rangle
\end{aligned}
$$

where $* \in \{+,-\}$.

Obviously, $W_E^*(p,t)$ respects the definition of WFN standard functions (Chiola et al. 1991) and has the same semantic as the IF-THEN-ELSE expression $W^*(p,t)$.

6.1.2. IF-THEN-ELSE Arc by Guarded Transitions

Some Petri Nets tools do not support the concept of guarded function. In this case, we can use two guarded transitions to model the "then" and "else" clause of the IF-THEN-ELSE arc respectively.

Figure 4 shows an example of an IF-THEN-ELSE arc and its context. G is the guard of transition t (it is possible that G=TRUE) and g is the condition in the IF-THEN-ELSE expression. The other notations will be the same as defined in section 6.1.1.

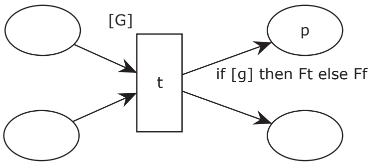


Figure 4 IF-THEN-ELSE Arc by Guarded Transitions

We propose an equivalent structure taking into consideration three cases based on the relationship between G and g.

Case 1: G is stronger than g ($G \subset g$)

In this case, $\{c \in C \mid G(c) = TRUE\} \subset \{c \in C \mid g(c) = TRUE\}$.

According to the firing principles, transition t is not enabled with a color $c \in \{c \in C \mid g(c) = FALSE\}$. Consequently, only the function F_t in the THEN-clause should be considered. Thus, the incidence function $\mathrm{W}^*(p,t) = if\ g\ then\ F_t\ else\ \mathrm{F_f}$ is then rewritten as $\mathrm{W}^*(p,t) = F_t$, where $* \in \{+,-\}$.

Case 2: G and g are disjoint ($\mathrm{G} \cap \mathrm{g} = \emptyset$)

In this case, $\{c \in C \mid G(c) = TRUE\} \cap \{c \in C \mid g(c) = TRUE\} = \emptyset$

In opposition to Case1, transition t is not fireable with any color $c \in \{c \in C \mid g(c) = TRUE\}$, so only the function F_f in the ELSE-clause should be considered. Thus, the incidence function $\mathrm{W}^*(p,t) = if\ g\ then\ F_t\ else\ \mathrm{F_f}$ is then rewritten as $\mathrm{W}^*(p,t) = F_f$, where $* \in \{+,-\}$.

Case 3: general case not belonging to Case 1 nor Case 2

In this case, we partition the colorset satisfying the guard G (i.e. $C_G = \{\mathrm{c} \in \mathrm{C} \mid \mathrm{G(c)} = \mathrm{TRUE}\}$) into two sub-colorsets C_G^T and C_G^T :

$C_G^T = \{\mathrm{c} \in \mathrm{C} \mid \mathrm{G(c)} = \mathrm{TRUE\ and\ g(c)} = \mathrm{TRUE}\}$,

$C_G^F = \{\mathrm{c} \in \mathrm{C} \mid \mathrm{G(c)} = \mathrm{TRUE\ and\ g(c)} = \mathrm{FALSE}\}$,

such that $C_G = C_G^T \cup C_G^F$ and $C_G^T \cap C_G^F = \emptyset$.

Then one models transition t with two transitions t^T and t^F defining $WN' = (P', T', \mathrm{Cl}', \mathrm{C}', W^{-\prime}, W^{+\prime}, \Phi', M_0')$, where

- $P' = P$; $\mathrm{Cl}' = \mathrm{Cl}$; $M_0' = M_0$;
- $T' = \{\, T/\{t\} \,\} \cup \{t^T, t^F\}$;
- $\mathrm{C}'_{t^T} = \mathrm{C}'_{t^F} = \mathrm{C}_t$;
- Assume $\mathrm{W}^*(p,t) = if\ g\ then\ F_t\ else\ \mathrm{F_f}$, $* \neq **$ and $*, ** \in \{+,-\}$, then $W^{*\prime}(p,t^T) = F_t$, $W^{*\prime}(p,t^F) = F_f$, $W^{**\prime}(p,t^T) = W^{**\prime}(p,t^F) = W^{**}(p,t)$;
- $\forall\, p' \neq \mathrm{p} \in P'$, $* \in \{+,-\}$, $W^{*\prime}(\,p', t^T) = W^{*\prime}(\,p', t^F) = W^{*\prime}(\,p', t)$;
- $\forall\, t' \in \{\, T' / \{t^T, t^F\} \,\}$, $\forall\, (p' \in P')$, $W^{+(-)\prime}(\,p', t') = W^{+(-)}(\,p', t')$;
- $\emptyset'(t^T) =$ G and g, $\emptyset'(t^F) =$ G and $(\neg$g).

We will give an example later in section 6.3 when introducing the operation 3 for a train queue structure.

6.2. Predecessor Function and Its WFN Realization

In WFNs the successor function $\oplus X_i^j$ is defined as an elementary function. While in some modeling cases it is also necessary to use a predecessor function, which is not defined in WFNs. This study proposes a method to use predecessor functions that will be noted as $\ominus X_i^j$. This study also gives its application constrains. With respect to these constrains one could always find an equivalent WFN structure which behaves as a predecessor function.

Let $\ominus X_i^j(c)$ be an application from $c \in C = C_1^{e_1} \times \cdots \times C_k^{e_k}$ to the predecessor of c_i^j in C_i, where C_i is an ordered class. It is worth noting that like the successor function, the predecessor of the first item in C_i is the last item.

To benefit from the features of WFN, when analyzing such a colored net using predecessor functions defined above, we could transform it to an equivalent WFN.

Figure 5 shows an example of a colored net using predecessor (Figure 5 (a)) and its equivalent WFN (Figure 5 (b)). In the example $C(t) = C \times C \times C$; X, Y are two identity functions that $X = X^1$, $Y = X^2$; (X-1) and (X+1) are notations of predecessor and successor functions that (X-1) = $\ominus X^1$, (X+1) = $\oplus X^1$.

Figure 5 (a) uses the predecessor function (X-1) in the output arc of transition t. In order to replace this structure using WFN, we do the following two steps:

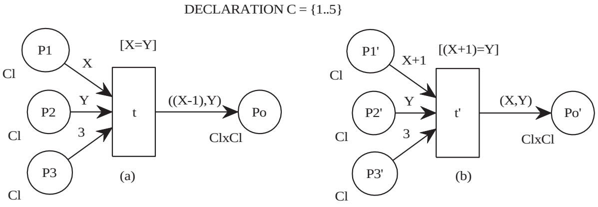


Figure 5 Predecessor and the Equivalent WFN

Step 1: search for all the instances of the identification function X in the “context of transition t”, and replace them with the corresponding successor function (X+1).
It is worth noting that the three atomic predicates defined in WFN are replaced by the following rules respectively:

1. [X = Y] is replaced by [$Y = \oplus X$];
2. [$X = \oplus Y$] is replaced by [$\oplus X = \oplus Y$], which means [X = Y];
3. [X ∈ D] is replaced by [$\oplus X \in$ D], which is not a WFN guard. In this case, let $D = \{x_m \cdots x_n\}$ be a subclass, we define a new subclass $D' = \{x_{m-1} \cdots x_{n-1}\}$ where x_{m-1} and x_{n-1} are the predecessors of x_m and x_n, respectively. Then [$\oplus X \in$ D] is transformed to [$X \in D'$].

In the example, the two instances are found in the guard of transition t and on the output arc from the transition t respectively in Figure 5 (a), which are then replaced by (X+1) in Figure 5 (b).

Step 2: replace the predecessor function (X-1) (in Figure 5 (a)) with the corresponding successor function X (in Figure 5 (b)). In the example, the one on the output arc of transition t is replaced by t’.

Application constrains: In order that the replacement above can be performed, for a color instance X_i^j, if the predecessor function $\ominus X_i^j$ is used, the corresponding successor function $\oplus X_i^j$ cannot appear in the “context of the same transition t”, which includes the arcs connected with transition t and the guard of transition t. In other words, we cannot use the predecessor and the successor function of a same color instance X_i^j simultaneously and in the “context of a transition”.

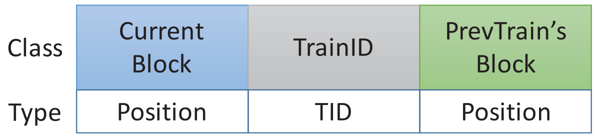
6.3. Queue Structure in WFN

While modeling railway control systems, more exactly the RBC model needs to have a centralized storage of the trains’ queue, i.e., the information of all the trains in the railway line it manages. The information includes at least the trains’ identifications, their positions and the sequence of these trains.
Using WFN, we can use a token of a product domain (e.g., $< TrainID \times Position >$) to illustrate the identification and position of each train. However, it is difficult to establish an ordered relation among these tokens.
In some software for modeling high level Petri nets such as CPN Tools (Ratzer et al. 2003; Jensen et al. 2007), it is possible to use a “list” type, like that defined in most programming languages, to realize this queue of trains. While the use of “list” type color class will lose the convenience of analyzing a WFN (a colored net using “list” type is obviously not a WFN).
This section defines a queue structure in WFN. It establishes an order relation among different elements and supports several operations e.g. insert, removal, query, and update. In addition, a colored net using this WFN-compatible queue structure remains a WFN, maintaining all its advantages for its analysis.

The proposition of this queue structure is faced with the requirements of modeling a practical train control system. Its application will be illustrated with the implementation of the Movement Authority (MA) function as part of the RBC model in Section 7. The implementation of the queue structure (e.g. Operation 3) uses the modeling patterns proposed in section 6.1 and 6.2.
Some basic declarations used in the queue structure are defined as follows:

CLASS POS = <0> ∪ <1, 2, …, N> ∪ <N+1>;
TID = <T(0), T(1), T(2), … , T(M)>;
DOMAIN TRAINITEM = <POS, TID, POS>.

The color class POS is ordered and is divided into three sub-classes. Each position in <1, 2, …, N> represents a particular block in the railway line (which has N blocks; N is consequently a parameter that is bound to a specific value for each real line). The other two sub-classes <0> and <N+1> are for special purposes and will be explained in the following paragraphs. For convenience, we define a constant HEAD = N+1 for the following parts of this paper.
The color class TID enumerates the different identifiers of trains, in which T(0) is reserved as a special value and it does not represent a real train. TID could be an unordered class.
The color domain TRAINITEM is a 3-tuples Cartesian product and has the following practical meaning, as shown in Figure 6.



Class	Current Block	TrainID	PrevTrain’s Block
Type	Position	TID	Position

Figure 6 Structure of TrainItem

Each token (except the token *TrainQueueRear*) of color domain TRAINITEM represents a particular train (*TrainID*) with its current position (*Current Block*). In addition, each train is connected to its previous train by indicating the block where its predecessor is located (*PrevTrain’s block*). The following two special values help to construct the queue structure:

First Train: The first train in the railway line has not a preceding train regarding the actual state of the line. Let us give a special value “HEAD: POS” to its third field. As defined above, the constant HEAD = N+1. The block “N+1: POS” does not exist in the railway infrastructure. This value is used to indicate the first train in the queue.

Train Queue Rear: It doesn’t present a real train, but offers a link to the rear train’s position. The first and second fields of this item are always “0: POS” and “T0: TID”, which is used to identify this rear item. It is worth noting that the block “0:POS” is not a real block, neither T(0) a real train. Its third field indicates the position of the rear train in the queue.

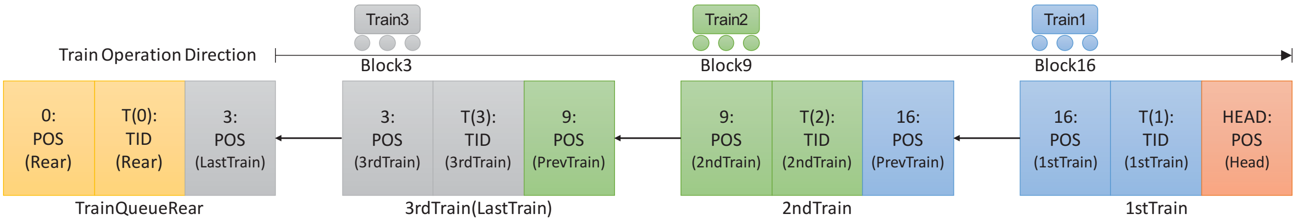


Figure 7 Conceptual List Structure of 3 Trains

The queue structure is then constructed by two places, i.e., place *TrainQueue: TRAINITEM* and place *FreeBlock: POS*.

In place *TrainQueue* there are tokens of color domain “TRAINITEM”. In a special case where there are no trains in the railway line, the place *TrainQueue* is not empty, there still exists a token (*Train Queue Rear* in which the third value is “HEAD: POS”): <0: POS, T(0): TID, HEAD: POS>.

Tokens in Place *FreeBlock* represents the free blocks that are not occupied by a train. Each time a train moves, it will take the new position token from Place *FreeBlock* and release the token of its previous position.

To illustrate how to model a practical queue structure of trains in WFN, here is a general case assuming that there are 3 trains in the railway line, as shown in Figure 7.

Now it is necessary to define some basic operations to manage the queue structure.

Operation 1: Insert Operation

A new train is always inserted from the rear of the queue and it is normally inserted in Block 1. Then the objective of this operation is to insert a new token with <TrainID = tr: TID, CurrentBlock = 1: POS> to the queue and to modify the concerned tokens. This operation is explained with Figure 8, where *tr* is the identifier of the train to insert, and *p_last* is the position of the last train before this inserting operation.

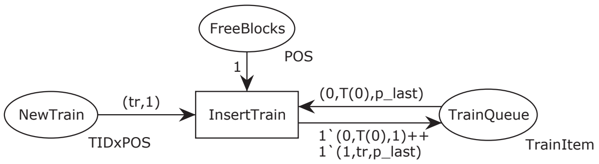


Figure 8 Insert Operation

For Operation 1, it is worth noting that:

- If there is already a train in block 1 before the operation, the token <1: POS> is no longer in place *FreePlace*, the new train to be inserted needs to wait until this block 1 is set free again;
- The operation also considers the case that the railway line is previously empty, i.e. p_last = HEAD: POS.

Operation 2: Removal Operation

When a train arrives at the end block (Block N) of the railway line and then leaves this railway line, its token <N:POS, tr:TID, HEAD:POS> must be removed from place *TrainQueue* and the token of the block <N: POS> must be released to place *FreeBlock*.

Figure 9 shows that the two tokens representing the first train <N, tr, HEAD> and its successor train <p1, t1, N> are involved. The last train token is removed and the “PrevTrain” field of train “*t1*” is updated to “HEAD: POS” as it becomes the first train in the railway line.

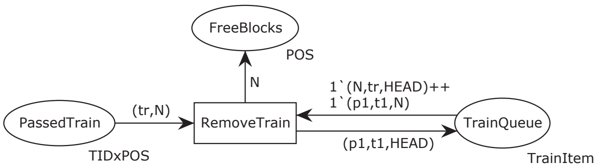


Figure 9 Removal Operation

For Operation 2, it is worth noting that:

- The variable “tr” is in fact, not necessary as it could be implied by only using its position “N”, however we still use it to guarantee the right train we want to remove.
- When the train to be removed is the only train in the queue, its successor train is the rear item, i.e. “p1=0” and “t1=T(0)”. In this case the removal operation will result in the case with a unique token <0: POS, T(0): TID, HEAD: POS >, which means there is no more trains in the railway line.

Operation 3: Request of Movement Authority (MA) for a train.

In order to avoid the collision of trains, each train must request the RBC for MA. By receiving the MA requested, the train knows to which block it can advance safely without any collision risk. In practice, it can advance until the anterior block to the current position of its predecessor train. This authorized position is called the End of Movement Authority (EOA). Normally the train needs to request a new MA regularly before reaching its EOA, in order not to be stopped during its advancement.

- When the considered train is the first train in the railway line, it can advance until the last block of this railway line, so its EOA is position N;
- When the considered train is not the first train on the railway line, and its predecessor train is currently in block “p_pre”, then its EOA should be the block “p_pre - 1”.

Therefore, the EOA position for a particular train with “tid=tr” could be expressed as “IF (p_pre = HEAD) THEN *N* ELSE (p_pre - 1)”, which contains an “IF-

THEN-ELSE" arc and a predecessor function. Such an arc expression is not supported by WFN. While, after the application of equivalent structures proposed in section 6.1.2 and 6.2 (considering the definition HEAD = N+1), the structure of this operation using WFN can be given in Figure 10.

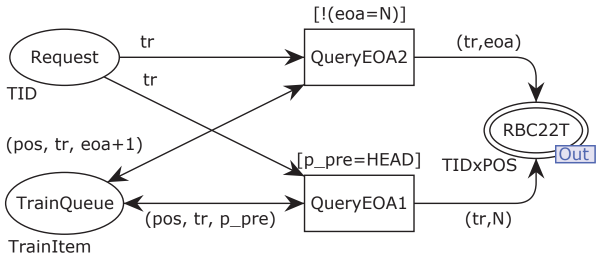


Figure 10 Request of Movement Authority

Operation 4: Update of Train Position

The update of train position does not affect the order relation of trains in the queue. Transition Update1 replaces the train's positon with a new value while transition Update2 deals with its successor train.

Figure 11 illustrates the WFN implementation of this operation. When the position value is updated, the previous position token "p0" is released to place FreeBlocks and the new position value is taken. We use two guarded functions with the guard [p<>p0] to avoid the manipulation to place FreeBlocks when the new value equals the old one.

The update operation is always triggered when RBC receives a position report < tr: TID, p: POS > from a train. After the update in its database, it is necessary to send back an acknowledgement to the train.

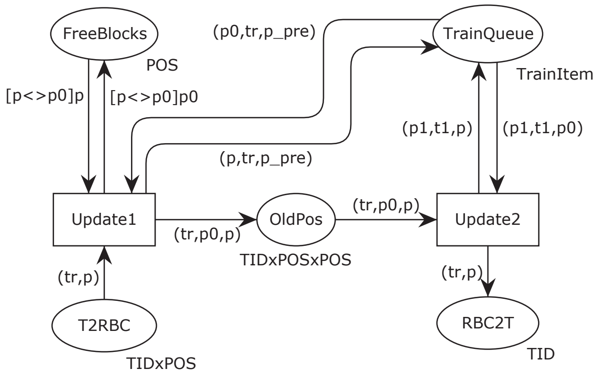


Figure 11 Update of Train Position

7. CASE STUDY

Faced with the practical problem of railway system controllers design, we have built several control models. Three models will be explained in this section. They offer the functions of managing multiple trains in a railway line, and with respect to WFN definitions.

7.1. System Structural Model

Figure 12 shows the model of the system architecture. The models are built in a modular way. The rectangles with double-line borders are modules.

This example model considers two train modules, whose detail is explained in section 7.2. A RBC module is built for the railway line management; and its details will be given in section 7.3.

Place Train2RBC and place RBC2Train represent the wireless interfaces between trains and the RBC module. The tokens in them are messages between different modules. In a similar way, place Balise2T models the Eurobalise interfaces. In our study, the Eurobalises are used to inform the trains of their locations.

The places T1info and T2info define the respective identifier of each train. Bidirectional arcs are used as the identifier tokens should never be consumed or modified. This can also be done for RBCs, in case of the modeling of a line controlled by several RBCs.

As all the train modules are exactly identical (except their TIDs as initial markings), it is possible to add more train modules to the architectural model, as long as these train modules are connected to the suitable interfaces and assigned with a TrainID.

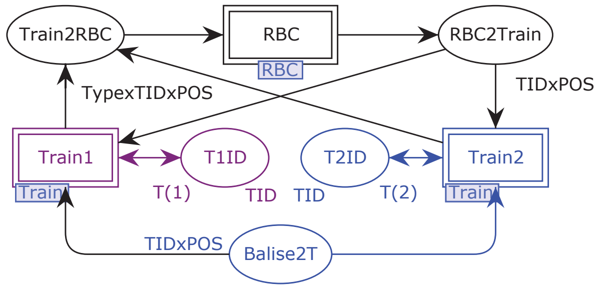


Figure 12 Structural Model of the Case Study

7.2. Train Model

Figure 13 gives the train model integrated with the functions to enter a railway line, to advance with respect to its MA, and finally to pass this railway line.

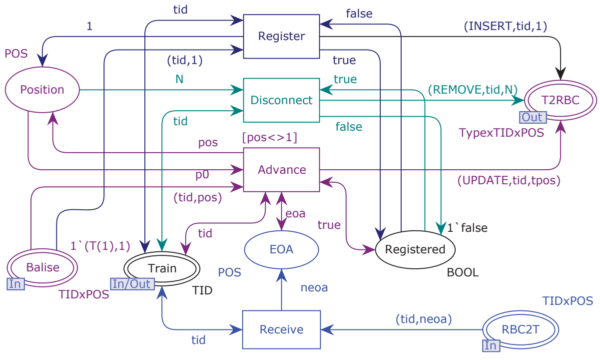


Figure 13 Train Module of Case Study

Let us suppose that initially the train just arrives on the first block (place Balise: 1, place Position is empty, place EOA: 1, place Registered: false). From this initial state, the following functions describes the behavior of the train sequentially.

Register function is to register the train itself to the RBC. To fire the transition Register, the train must be located in block 1 (token "1: POS" in place Balise) and the state

Registered is false. When transition Register fires, it will send a message of type “INSERT” to the RBC, put a token “1: POS” into place Position, and change the state Registered to true.

Advance function is to simulate the advancement of a train already in the blocks. The train can advance as well as it does not arrive on its EOA position. Each time the train passes a block, its new position is received via place Balise so that transition Advance is fired. The token in place Position is updated, and a position report is also sent to inform the RBC of its new position.

Transition Receive can be fired when there is a MA message generated by RBC. Place RBC2T are shared by all the trains, so only the message for this train (tid) will be received. After receiving the message, its new EOA value is then memorized in place EOA.

Disconnect function is to inform the RBC that it has passed the railway line. After passing the last block (block N), the transition Disconnect can be fired. Then the token “N: POS” is removed from place Position, the train sends a message of type “REMOVE” to the RBC and changes its state Registered to false.

7.3. RBC Model

Figure 14 represents the RBC model. The four main functions (e.g., InsertTrain, RemoveTrain, QueryEOA and PositionUpdate) are well explained as the four operations of train queue structure in Section 6.3.
What we need to add in this model is the way to fire different functions. The functions InsertTrain, RemoveTrain and PositionUpdate can be fired after receiving a message from a train. A field “Type” (i.e., INSERT, REMOVE or UPDATE) in the message helps to choose the corresponding functions to fire.
For convenience, the RBC model regards a position report as a MA request. So, each time it receives a position report, the trainID is then put into place Request in order to generate a MA for it. The RBC also generates a MA for a train that is just registered (after transition InsertTrain is fired).

8. CONCLUSIONS AND PERSPECTIVES

In this paper, we have shown that it is possible to use WFN to model complex systems such as railway systems by using several modeling patterns and techniques that we propose. These modeling patterns also make it possible to model some structures and extensions of other types of colored Petri nets such as the CPNs defined in (Jensen, 81), based on the WFN rules (e.g., elementary colored functions). We illustrate our propositions by applying them to the Movement Authority (MA) function modeled in the ECTS-2 context.
The prospects for this work are of course to continue the modeling of other functions of a railway system with a view to its complete automation. We are thinking in particular about the routing function of a train inside a node, which is implemented today in a semi-automatic mode, which requires a man-machine cooperation.
Beyond the modeling stage, it will be necessary to complete this work by the development of a method allowing the formal verification of our models while controlling their combinatory explosion. On the other hand, we want to use the techniques of reductions applicable to the WFN for our developed models. We also plan to directly use the calculation of colored invariants on reduced models but also the construction if necessary of symbolic reachability graph. All these mentioned methods may be complemented by the proposal of a formal model verification such as the assume-guarantee reasoning (Nguyen Huu 2013) in order to ensure that the global model inherits the properties verified on its component modules.

ACKNOWLEDGMENTS

This study is carried out within the framework of the CompRAIL project of ELSAT2020. The ELSAT2020 research program is co-financed by the European Union with the European Regional Development Fund, the French state and the Hauts de France Region Council.

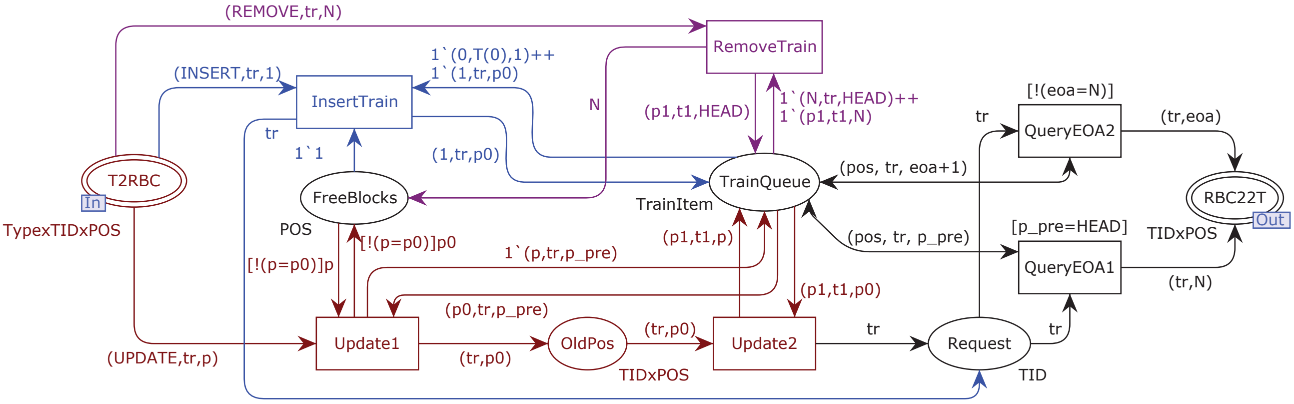


Figure 14 RBC Module of Case Study

REFERENCES

Evans W.A., 1994. Approaches to intelligent information retrieval. Information Processing and Management, 7 (2), 147–168.

van der Aalst, W.M.P., Stahl, C. & Westergaard, M., 2013. Strategies for Modeling Complex Processes Using Colored Petri Nets. In K. Jensen et al., eds. Transactions on Petri Nets and Other Models of Concurrency VII. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 6–55.

Berthelot, G. & Lri-Iie, 1986. Checking properties of nets using transformations. In G. Rozenberg, ed. Advances in Petri Nets 1985. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 19–40.

Chiola, G. et al., 1995. GreatSPN 1.7: GRaphical Editor and Analyzer for Timed and Stochastic Petri Nets. , 24, pp.47–68.

Chiola, G. et al., 1991. On Well-Formed Coloured Nets and Their Symbolic Reachability Graph. In High-level Petri Nets SE - 13. Springer, pp. 373–396.

Couvreur, J.M. & Martínez, J., 1991. Linear invariants in commutative high level nets. In G. Rozenberg, ed. Advances in Petri Nets 1990. Springer Berlin Heidelberg, pp. 146–164.

Diaz, M., 2013. Petri Nets: Fundamental Models, Verification and Applications M. Diaz, ed., London, UK: John Wiley & Sons.

Esparza, J. & Hoffmann, P., 2016. Reduction Rules for Colored Workflow Nets. In Fundamental Approaches to Software Engineering: 19th International Conference (FASE 2016). pp. 342–358.

Haddad, S., 1991. A Reduction Theory for Coloured Nets. In K. Jensen & G. Rozenberg, eds. High-level Petri Nets: Theory and Application. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 399–425.

Janczura, C.W., 1999. Modelling and Analysis of Railway Network Control Logic using Coloured Petri Nets. University of South Australia.

Jansen, L., Meyer zu Hörste, M. & Schnieder, E., 1998. Technical Issues in Modelling the European Train Control System (ETCS) Using Coloured Petri Nets and the Design/CPN Tools. In K. Jensen, ed. Workshop on Practical Use of Coloured Petri Nets and Design. Daimi PB-532, Aarhus, Denmark: Aarhus University, pp. 103–115.

Jensen, K., 1981. Coloured Petri Nets and the Invariant-Method. Theoretical Computer Science, 14(3), pp.317–336.

Jensen, K., Kristensen, L.M. & Wells, L., 2007. Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems. International Journal on Software Tools for Technology Transfer, 9(3–4), pp.213–254.

Kordon, F. & Paviot-Adet, E., 1999. Using CPN-AMI to Validate a Safe Channel Protocol. In Proceedings of the International Conference on Theory and Applications of Petri Nets - Tool presentation part. Williamsburg, USA.

Meyer zu Hörste, M., 1999. Modelling and Simulation of Train Control Systems Using Petri Nets. In FMRail Workshop.

Murata, T., 1989. Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, 77(4), pp.541–580.

Nguyen Huu, V., 2013. Modular Verification of Petri nets. UNIVERSITY OF BORDEAUX 1.

Ratzer, A.V. et al., 2003. CPN Tools for Editing, Simulating, and Analysing Coloured Petri Nets. In Proceedings of the 24th international conference on Applications and theory of Petri nets (ICATPN'03). pp. 450–462.

Vanit-Anunchai, S., 2014. Experience using Coloured Petri Nets to Model Railway Interlocking Tables. In 2nd French Singaporean Workshop on Formal Methods and Applications (FSFMA'2014). Singapore, pp. 17–28.

Vanit-Anunchai, S., 2010. Modelling Railway Interlocking Tables Using Coloured Petri Nets. In D. Clarke & G. Agha, eds. Coordination Models and Languages: 12th International Conference (COORDINATION'2010). Amsterdam, The Netherlands: Springer Berlin Heidelberg, pp. 137–151.

Vanit-Anunchai, S., 2009. Verification of Railway Interlocking Tables using Coloured. In The 10th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools. DAIMI PB 590, Department of Computer Science, University of Aarhus, pp. 139–158.

Xie, Y., Khlif-bouassida, M. & Toguyeni, A., 2016. Modeling Of Automatic Train Operation Control Using Colored Petri Nets. In 11th International Conference on Modeling, Optimization & Simulation (MOSIM 2016). Montréal, Canada.

Žarnay, M., 2004. Use of Petri Net for Modelling of Traffic in Railway Stations. In Proceedings of international conference Infotrans. Pardubice.

ANTI-WIND-UP SOLUTION FOR A TWO DEGREES OF FREEDOM CONTROLLER

Lucrecia Gava[(a)], **Aníbal Zanini**[(b)]

[(a), (b)] Laboratorio de Control - Facultad de Ingeniería - Universidad de Buenos Aires - ITHES - CONICET

[(a)]lucreciagava@hotmail.com, [(b)]azanini@fi.uba.ar

ABSTRACT

This paper proposes an anti-windup solution for a STR controller with two degrees of freedom (2DoF). The main advantage of this kind of controllers is that the dynamic for tracking references can be different than the one specified for disturbance rejection. As a design condition, it is also proposed to eliminate the steady state error so that, the presence of an integral term in the controller is mandatory. Due to the physical limitations and an unstable factor in the controller, an anti-windup strategy is needed. These logics or non-linear schemes are extensively studied for classical controllers but not for 2DoF strategies.

The proposal has been verified both in simulation and on a physical plant.

Keywords: two-degree control, anti-windup, integral action.

1. INTRODUCTION

Controllers with two degrees of freedom (2DoF) are attractive because they can separate the two control problems: the set point (SP) tracking and the disturbance rejection.

PIDs are extensively used in controllers of two degrees of freedom (PID-2DoF). Tripathi R. and Hanamoto T. (2017) show the improvement between 1-DOF and 2-DOF for the optimization of a dc voltage controller. Sahu R., Panda S., Rout U. and Sahoo D. (2015) compare different strategies and expose a new and better technique. Wang D., Liu T., Sun X., Zhong C. (2016) propose a discrete-time 2-DOF design method for integrating and unstable process with time delay.

Even the anti-windup solutions have been largely studied for PID systems, as can be seen in the review of solutions presented by Espina J., Arias A., Balcells J. and Ortega C. (2009), it is not a closed subject. Particular solutions are always being studied and compared for specific systems; such is the case of Perez T. (2009), who presented an anti-windup design for the dynamic positioning of marine vehicles with control allocation. However, there are no previous thorough studies that provide a satisfactory Anti-Wind-Up solution for an STR 2-DoF system as for PIDs.

Although the PID-2DoF is deeply studied and used, the STR has some advantages since any kind of dynamics can be introduced. Some authors (Matijevic M., Sredojevic R. and Stojanovic V., 2011), argue that the little industrial use of the STR is due to its complexity in the implementation. This paper shows a simple way to solve STR implementation problems, in particular the windup effect.

The main advantage of a STR controller is its structure, which is able to carry out improved performances due to the fact that it stores several states of the set point, the process variable and the manipulated variable. What is more, STR method allows the user to divide the two degrees of freedom into individual and understandable elements and to add terms without being restricted to any amount of degrees.

Nevertheless, a set-back of this structure is that the implementation of traditional anti-windup structures does not work. The anti-windup solution is more complex than the used in the classical controllers, where only one previously manipulated variable value is needed.

Moreover, it is well known that controllers with any unstable term (e.g. integral action) working on a process with limited actuators can cause malfunctioning of the closed loop system or even have stability problems. For instance, if there is an integral term and the calculated control action is unfeasible (due to any actuator limitation), the integral calculation will increase indefinitely (windup effect). When the error changes its sign, the manipulated value variation will be delayed because of the windup. This problem is strictly of a physical nature because the model of the controller does not contemplate any kind of restriction.

The main goal of this paper is to provide an anti-windup solution that works for a 2-DoF controller without adding an extra parameter decision or an extra mathematical resolution.

First, we will provide a characterization of anti-windup and of the model used for investigation. Then we will explain the method used to build the STR control model. Next, we will assess the different anti-windup solutions for STR one degree of freedom controllers. And finally, we will present a solution for STR 2-DoF controller.

2. PROCESS MODEL

The process model adopted is the classical discrete linear model expressed in the following equation:

$$A(z^{-1})y_k = B(z^{-1})u_k \tag{1}$$

where *u(k)* is the manipulated variable (MV) and *y(k)* the process variable (PV).

The degree of $B(z^{-1})$ must be less than the degree of $A(z^{-1})$. The polynomials $A(z^{-1})$ and $B(z^{-1})$ have no common factor, and the polynomial $A(z^{-1})$ is monic.

3. CONTROL LAW

Next figure shows the STR structure (Astrom and Wittenmark, 1997) where the block "process" represents the plant to be controlled.

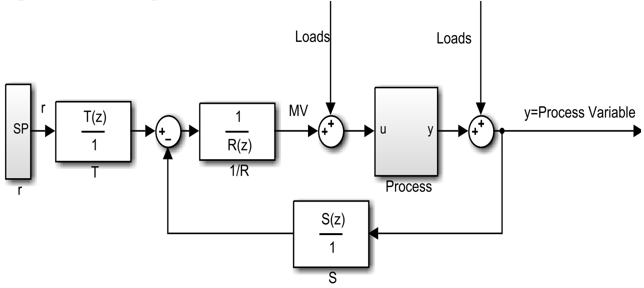


Figure1. STR Block Diagram

The resulting linear controller is:

$$R\left(z^{-1}\right)u_k = T\left(z^{-1}\right)r_k - S\left(z^{-1}\right)y_k \tag{2}$$

The closed loop transfer function becomes

$$\left(A(z^{-1})R(z^{-1}) + B(z^{-1})S(z^{-1})\right)y_k = B(z^{-1})T(z^{-1})sp_k \tag{3}$$

where the characteristic equation

$$\left(A(z^{-1})R(z^{-1}) + B(z^{-1})S(z^{-1})\right) = Alc(z^{-1}) \tag{4}$$

can be solved as a Diophantine equation. There is not a unique solution for the equation and this can be used to solve the 2DoF controller. Defining:

$$\begin{aligned} R\left(z^{-1}\right) &= A_{mr}\left(z^{-1}\right)\bar{R}\left(z^{-1}\right)\Delta\left(z^{-1}\right) \\ S\left(z^{-1}\right) &= A_{mr}\left(z^{-1}\right)\bar{S}\left(z^{-1}\right) \\ T\left(z^{-1}\right) &= \bar{B}_{mr}\left(z^{-1}\right)\bar{A}_{lc}\left(z^{-1}\right) \end{aligned} \tag{5}$$

with

$$\begin{aligned} \Delta\left(z^{-1}\right) &= 1 - z^{-1} \\ \bar{A}_{lc}\left(z^{-1}\right) &= A\left(z^{-1}\right)\bar{R}\left(z^{-1}\right)\Delta\left(z^{-1}\right) + B\left(z^{-1}\right)\bar{S}\left(z^{-1}\right) \\ G_m\left(z^{-1}\right) &= B\left(z^{-1}\right)\bar{B}_m\left(z^{-1}\right) / A_m\left(z^{-1}\right) \end{aligned}$$

In this way, it can obtain: a) the relation between PV and set point (SP) is $G_m\left(z^{-1}\right)$, which can be arbitrarily defined; b) due to $\Delta\left(z^{-1}\right)$, the controller has integral action, and c) the disturbance rejection dynamic is set independently, by the choice of $\bar{A}_{lc}\left(z^{-1}\right)$.

Without any loss of generality, as an example, a process model of first order will be considered as follows:

$$\begin{cases} A = 1 + a_1 z^{-1} \\ B = b_1 z^{-1} \end{cases} \tag{6}$$

The minimum degree of $\bar{R}\left(z^{-1}\right)$ and $\bar{S}\left(z^{-1}\right)$ are:

$$\begin{aligned} \bar{R}\left(z^{-1}\right) &= 1 + r_1 z^{-1} \\ \bar{S}\left(z^{-1}\right) &= s_0 + s_1 z^{-1} \end{aligned} \tag{7}$$

and

$$\bar{A}_{lc}\left(z^{-1}\right) = 1 + a_{lc1}z^{-1} + a_{lc2}z^{-2} + a_{lc3}z^{-3} \tag{8}$$

Hence solving the diophantine equation using the Sylvester matrix:

$$\begin{bmatrix} 1 & b_1 & 0 \\ a_1 - 1 & 0 & b_1 \\ -a_1 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_1 \\ s_0 \\ s_1 \end{bmatrix} = \begin{bmatrix} a_{lc1} - a_1 + 1 \\ a_{lc2} + a_1 \\ a_{lc3} \end{bmatrix} \tag{9}$$

The unknown controller coefficients $\begin{bmatrix} r_1 & s_0 & s_1 \end{bmatrix}$ always can be found if an only if $A\left(z^{-1}\right)$ and $B\left(z^{-1}\right)$ have no common roots.

3.1. Control Law without Anti-Wind-Up

According to (2) and (5), the resulting control law has the following polynomials:

$$\begin{cases} R = 1 + r_1 z^{-1} + r_2 z^{-2} + r_3 z^{-3} + r_4 z^{-4} + r_5 z^{-5} \\ S = s_0 + s_1 z^{-1} + s_2 z^{-2} + s_3 z^{-3} + s_4 z^{-4} \\ T = t_0 + t_1 z^{-1} + t_2 z^{-2} + t_3 z^{-3} \end{cases} \tag{10}$$

4. ANTI-WINDUP STUDY FOR STR MODEL WITH ONE DEGREE OF FREEDOM

There are two common situations that may lead to the need for an anti-windup solution: a) when a set point greater/less than the physically possible achievable is requested, and the manipulated variable saturates without reaching the set point; b) when a great performance is required, and the system is expected to react very fast so, the control actions are too violent, finding the upper or lower physical limitation.

As is proposed in (Huang, Peng, Wang, 2008), the anti-windup solutions can be divided in two groups: a) conditional integration and b) feedback calculation. In the former, the integration value is frozen and forced to be near the actual plant input (MV). In the later, an internal second feedback, which is related to the integrator term and the saturation element, is introduced.

First, the anti-windup solution for the scheme with one degree of freedom will be analyzed, in order to demonstrate that the difference with PIDs is not only related to the two degrees of freedom condition. In this case the closed loop poles to follow references and to reject disturbances are the same.

4.1. Conditional Integration

As a first case of study, the conditional integration solution will be introduced. The control law, in case of saturation, will be forced to be near the actual input variable (MV).

Three controllers are initially simulated, namely:

- Controller 1: with no anti-windup logic.
- Controller 2: the actual MV is saved for futures calculations.
- Controller 3: the actual MV is saved for futures calculations, and a “set point setup” is performed.

The "set point setup" holds the actual set point value to a feasible reachable PV. Then, the system will store the saturated value as the new set point. When the stationary value for that situation is finally reached; the error will be zero because the system is in the new desired point.

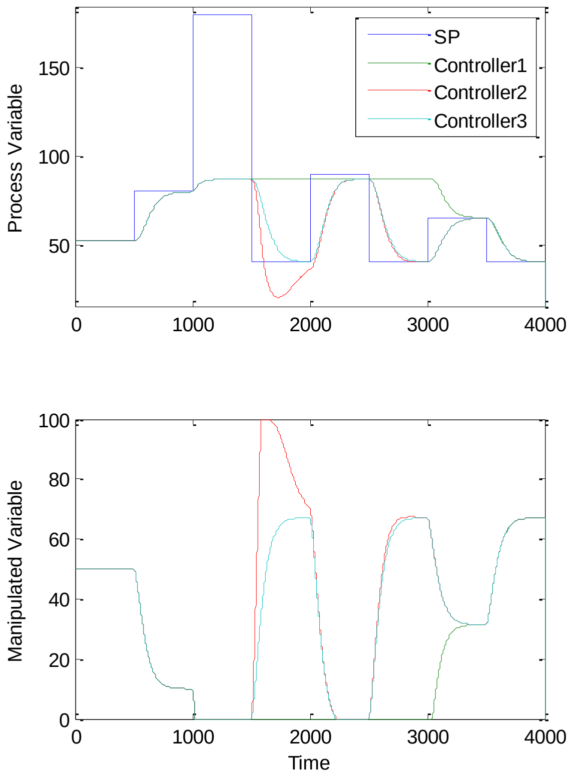


Figure 2: Conditional Integration, $p_{Amr} = 0.98$

Figure 2 shows a simulation of the three controllers with all the closed loop poles equal to $p_{Amr} = 0.98$, and compares how the controllers work when set point is greater than the physically possible achievable for process variable and manipulated:

Controller 1: the system gets stuck due to the difference between set point and process variable, which results in an accumulation of error. The control action starts working again after several further error samples of opposite sign, because there is no anti-windup logic being applied.

Controller 2: when the accumulated error is not correctly adjusted, the system may act automatically after the set point has been changed because the calculation of the control variable allows this. However, it may have an underdamped response, as is the case between times 1500 and 2000 in Figure 2.

Controller 3: It works following the reference model without any delay. As can be seen, the set point error does not influence the performance either (response appears after time 1500 and 2500). Since the system stores previous set point values, these become a key factor in the anti-windup solution.

In order to test Controller 3 against the second typical case where an anti-windup solution is required (that is, when great performance is required), the model poles $p_{Amr} = 0.90$ is used. The simulation result can be seen in Figure 3.

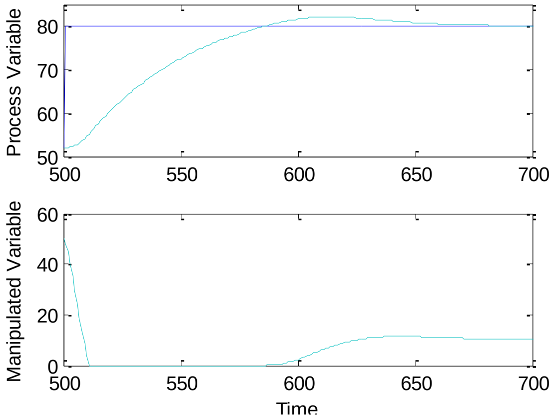


Figure 3: Conditional Integration for $p_{Amr} = 0.90$.

Here, an underdamped behaviour appears. The conclusion is that this technique depends on the required performance. Therefore, the anti-windup logic has not only to limit the MV, but it has also to recalculate its internal value for further states.

4.2. Integrator Feedback

These methods consist of adding a feedback related with the saturation element and the integrator term.

4.2.1. Tracking Feedback

The integral part is separated from the rest of the controller. In Figure 4, it is shown the difference between the *calculated manipulated variable* (CMV or v in the figure) and the actual manipulated variable (MV or *u* in the figure).

Both signals are compared and the difference affected by a gain, is fed to the integrator input.

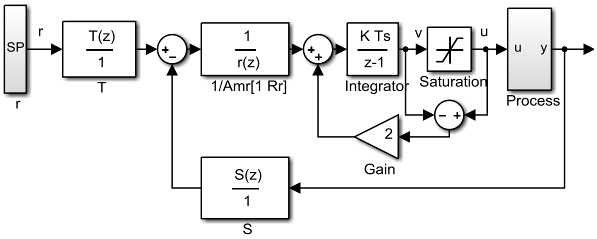


Figure 4: Tracking Feedback

If the MV is not saturated, this feedback has no effect; when the actuator is saturated, the feedback attempts to make the error signal equal to zero changing the integral value.

The feedback gain (2, in the figure) has to be adjusted for having the fastest correction.

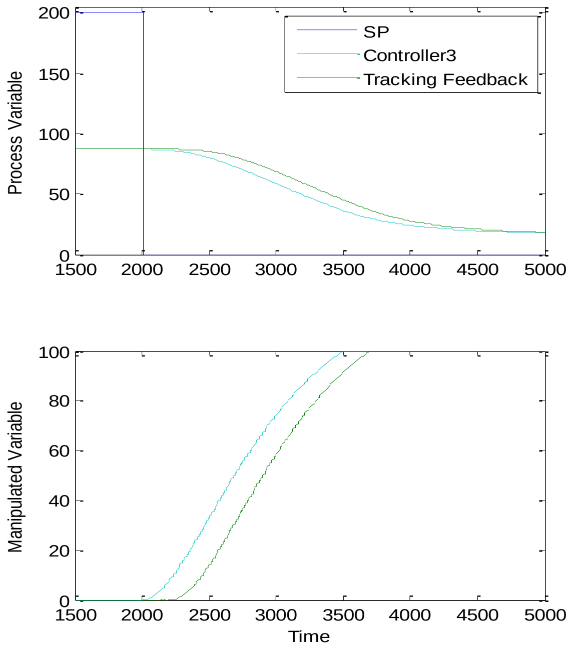


Figure 5: Tracking Feedback vs. Controller3 when the set point is greater than the physically achievable, $p_{Amr} = 0.98$.

This controller uses several (more than a PID) past samples of PV and MV and SP. Therefore, the tracking feedback scheme presents a delay when the SP is not reachable. This effect can be seen in Figure 5.

Thus, the magnitude of the difference between the set point and the process variable will affect the subsequent performance of the system, especially in the response time.

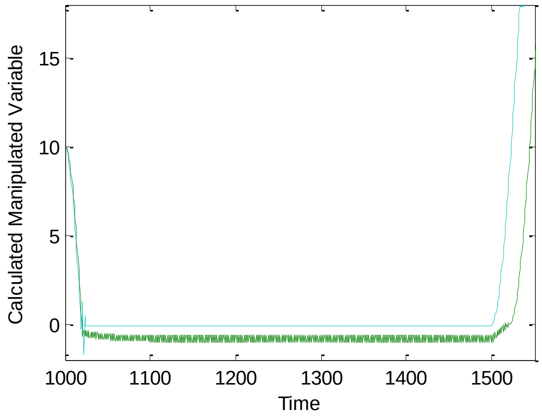


Figure 6: Tracking Feedback (green) vs. Controller3 (light blue) when the set point is greater than the physically achievable. CMV.

The CMV, by both methods in the saturation situation is shown in Figure 6 and 8. In the case of Controller 3, the first points in the saturation present difference between the CMV and the MV. This is due to the fact that the steady state has not been reached and the error is not zero.

However, in the tracking feedback remains in a constant oscillating error that is quickly eliminated after the set point changes, but there is a small delay in the reaction.

Figure 7 compares Tracking Feedback, Controller3 and an ideal situation without saturation, for the case of a great performance requirement. Until the manipulated variable is not saturated, both schemes give the same sequence of values and the same plant response. When the MV becomes saturated, the Controller 3 has an undesired underdamped performance while the tracking feedback reaches the set point with overdamping.

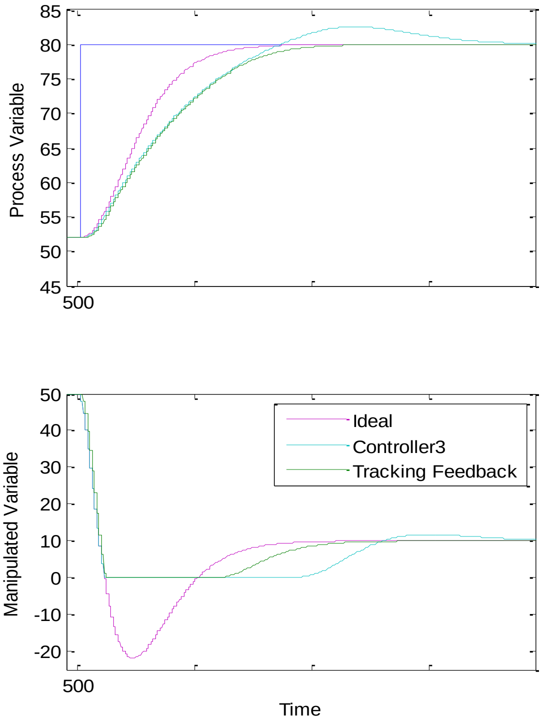


Figure 7: Tracking Feedback, Controller3 and Ideal Situation, for great performance case, $p_{Amr} = 0.90$.

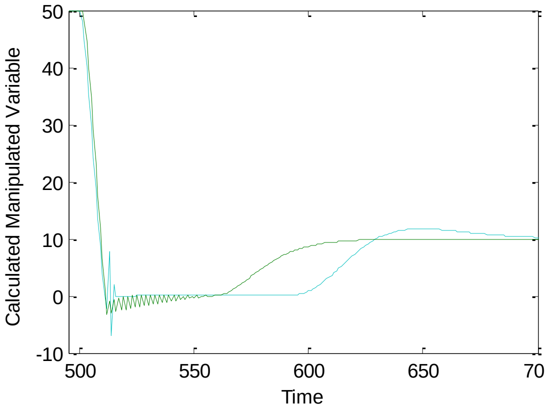


Figure 8: Tracking Feedback (green) vs. Controller3 (light blue) for great performance case, $p_{Amr} = 0.90$. CMV.

It is interesting to see the calculated values of the MV in this situation (Figures 8). The conclusion is that the

Tracking Feedback scheme is better in this situation, with a drawback: the response depends on the feedback gain that is another parameter to be set.
The aim of this work is to present a method for designing a controller with anti-windup without requiring any extra parameter.

4.2.2. Saturated Integration

Another way to obtain an integral with saturation is shown in Figure 9. The advantage of this scheme is that there is not any unstable block.

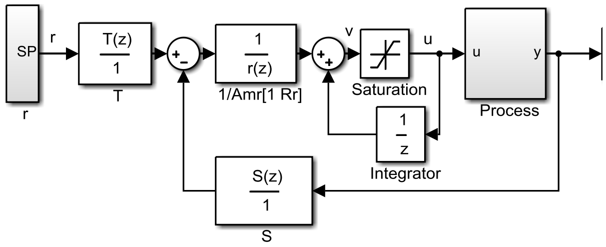


Figure 9: Saturated Integration

Figure 10 shows two overlapped SP changes: a) SP changes from 180 to 40 (blue) and b) SP changes from 90 to 40 (green). It can be seen that in case a) there is a greater delay in the MV (right side of figure 10). In this anti-windup scheme does not have into account the SP value, which should be considered to improve the logic. It can be proved that this delay depends on the position of the reference model poles.

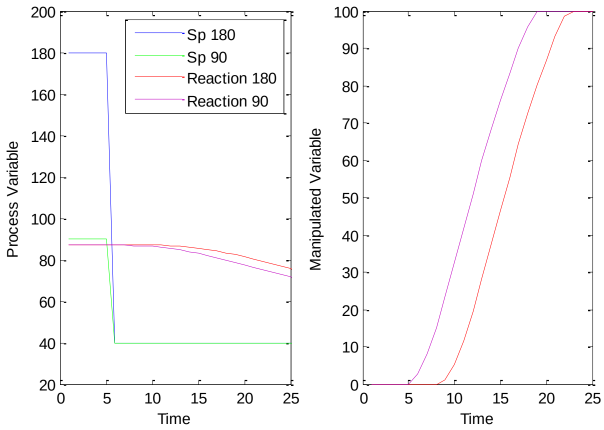


Figure 10: The response in Performance Requirement with model poles of $p_{Amr} = 0.90$.

4.2.3. Polynomial Characteristic Observer

Astrom and Wittenmark (1997) propose a different possible scheme that adds a characteristic polynomial observer $W(z^{-1})$ to the system. Figure 12 presents the block diagram.

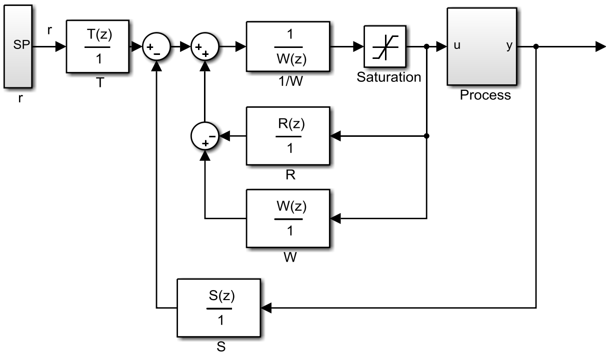


Figure 11: Polynomial Characteristic Observer

The linear equation of control (Equation 2) can be expressed as:

$$W(z^{-1})v_k = T(z^{-1})r_k - S(z^{-1})y_k + \left(W(z^{-1}) - R(z^{-1})\right)u_k \qquad (11)$$

This controller is equivalent to Equation 2 while the system is not in a saturated state; when saturated or equal to one of the saturation limits, $W(z^{-1})$ will determine the dynamics of the saturator. However, the authors don't explain how to define $W(z^{-1})$, which is a key factor.
Considering that $R(z^{-1})$ is derived from Equation 5, $W(z^{-1})$ can be defined as any part of $R(z^{-1})$ or a combination of them.
On the other hand, the stability must be guaranteed in the saturation situation or in the linear behavior. To assure the stability in the saturation situation, the polynomial $T(z^{-1})/W(z^{-1})$ must be stable which is easily seen rearranging Figure 11 as can be shown in Figure 12.

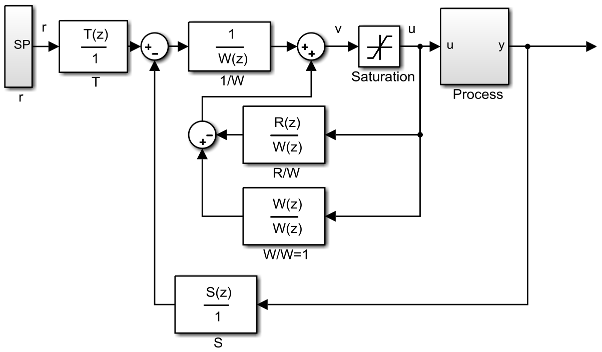


Figure 12: Rearrangement of Astrom's and Wittenmark's Polynomial Characteristic Observer

If, by instance, $W\left(z^{-1}\right)=1+r_1 z^{-1}$, then the system will be unstable in the saturation situation. Figure 13 shows the response in this example.

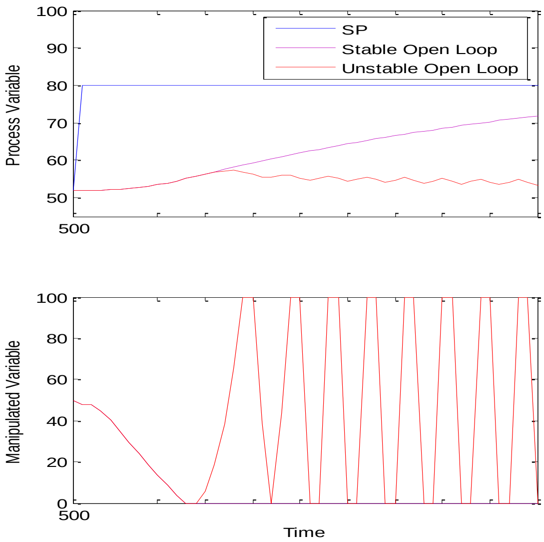


Figure 13: Polynomial Characteristic Unstable Observer $W\left(z^{-1}\right)=1+r_1 z^{-1}$.

If $W\left(z^{-1}\right)=\left(1+r_1 z^{-1}\right)A_{mr}$, the system is equivalent a pure integration feedback, which is not suitable for STR systems.

If $W\left(z^{-1}\right)=A_{mr}$, due to $A_{mr}=A_{lc}$, the dynamic of then $T\left(z^{-1}\right)/W\left(z^{-1}\right)=cte$. Thus, in saturation state, the controller update is made with actual values because they are updated with the manipulated variable that goes into the process (MV) and the process variable (PV). Then, the resulting system has a correct anti-windup solution for all one degree of freedom cases.

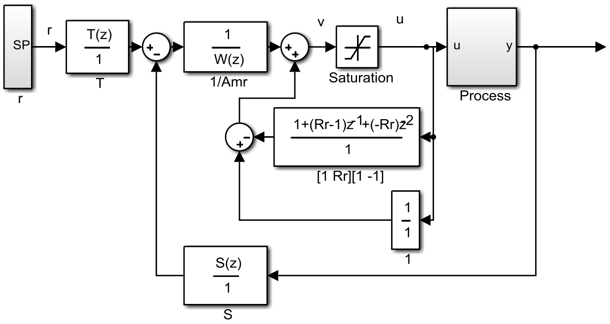


Figure 14: Polynomial Characteristic Observer, $W\left(z^{-1}\right)=A_{mr}$

Figure 14 shows the block logic for the anti-windup solution for the STR 1-DOF. In the next Figure (15), the evaluated simulation for the example, the anti-windup for the same set of set points when the poles are positioned in 0.9 or 0.98, are shown.

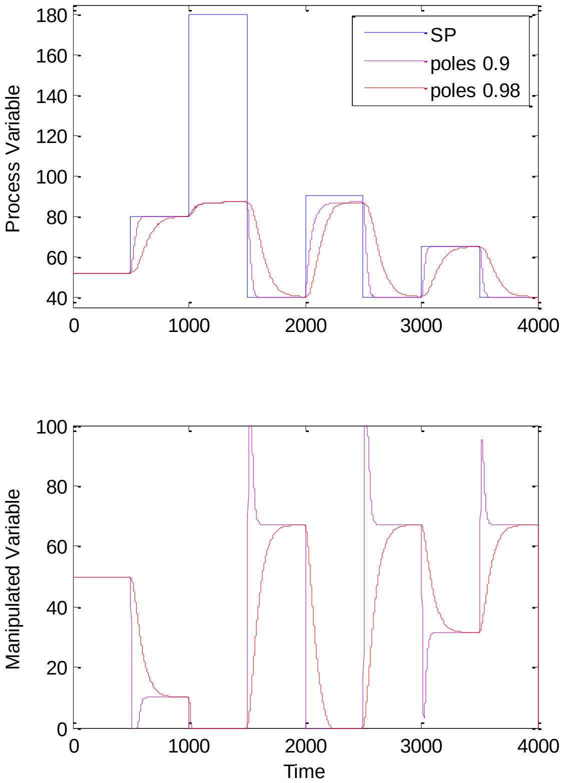


Figure 15: Anti-windup solution for One Degree of Freedom $W\left(z^{-1}\right)=A_{mr}$, poles $p_{Amr}=0.98$ and $p_{Amr}=0.90$.

In conclusion, if $A_{mr}=A_{lc}$ and the Control Law method (see section 4) is used to define all the polynomials, then it will always be possible to apply this anti-windup solution to any processes model.

It is interesting to remark that with this solution one degree is added to the internal feedback of the Pure Integrator. This is clearly related to the fact that the STR model has higher degrees than the PIDs and that conventional system is not enough to guarantee anti-windup for any possible situation.

5. ANTI-WIND-UP STUDY FOR STR MODEL WITH TWO DEGREES OF FREEDOM

With the scheme studied above, the anti-windup solution works exclusively when the poles are equal ($A_{mr}=A_{lc}$) because the term T saves previous states. Then, if $A_{mr}\neq \bar{A}_{lc}$, which is the case in a two-degree of freedom systems, there is no anti-windup solution because the dynamics are not being completely canceled with $W\left(z^{-1}\right)$ (see Figure 16).

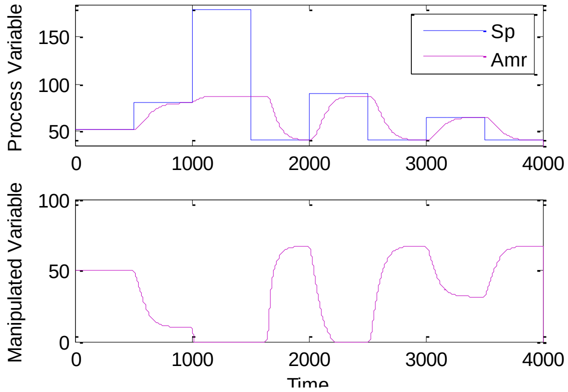


Figure 16: No anti-windup solution for Two Degrees of Freedom $A_{mr} \neq \bar{A}_{lc}$, with $W(z^{-1}) = A_{mr}$, poles $p_{Amr} = 0.98$ and $p_{Amr} = 0.90$.

5.1. The anti-windup solution for 2DoF Controller

Since $T(z^{-1})$ dynamic is not cancelled, this polynomial is introduced in the saturation feedback.

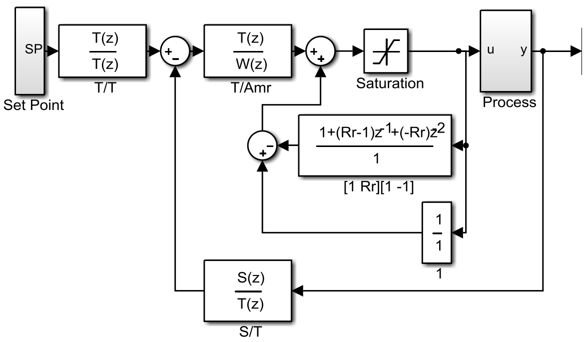


Figure 17: T Rearrangement

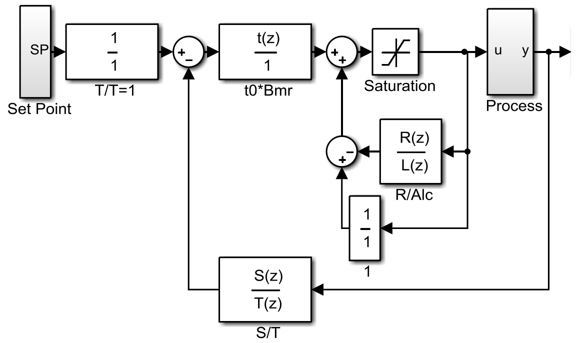


Figure 18: Final anti-windup Block Diagram

Figure 17 shows the rearrangement of Figure 12 and Figure 18 shows the new strategy introducing $\bar{A}_{lc}$ to the saturated intern loop.

Hence, the Block Diagram is rearranged so that all the terms that use stored data are updated with the actual values of u_k and y_k.

As $T(z^{-1}) = \bar{B}_{mr}(z^{-1})\bar{A}_{lc}(z^{-1})$, then it turns out an open direct loop without dynamic can be used if and only if $\bar{A}_{lc}$ is introduced in the loop.

Since the internal saturation feedback has the $R(z^{-1})/A_{lc}(z^{-1})$ term inside, this diagram eliminates all previous information in case of saturation, leaving all the values stuck in the actual process-related values without accumulating any kinds of error. What is more, this method offers the possibility to decrees the order of the A_{mr} polynomial

Figure 19 shows the anti-windup solution where A_{mr} has three different degrees.

Note 1: Figure 19 does not attempt to show better response, the regulation of the poles will deal to a better response or not.

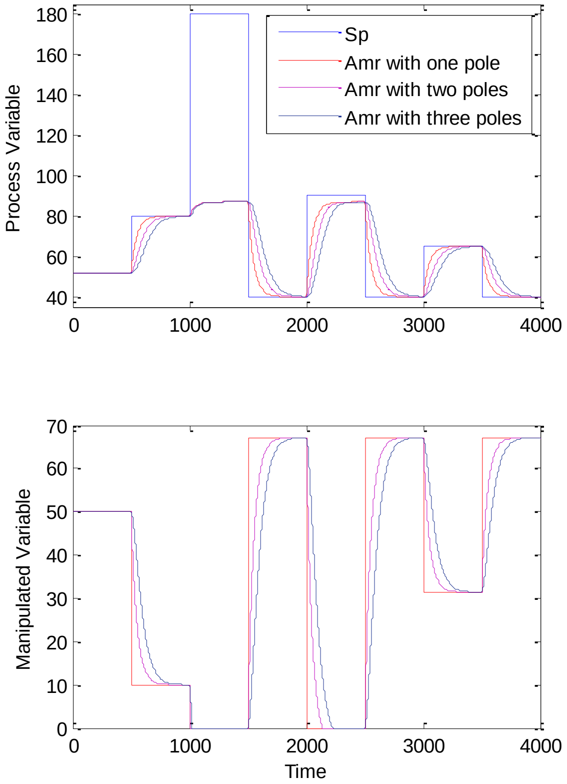


Figure 19: Three different degrees of A_{mr} with Anti-Wind-Up solution, poles $p_{Amr} = 0.98$ and $p_{Amr} = 0.90$.

This gives a solution to the problem of anti-windup that is completely independent of the model and that is not subjected to any extra decision. It can be seen that the resulting performance is overdamped for all the times, regardless of the requirement of the system.

6. EXPERIMENTAL APPLICATION

Finally, the STR anti-windup solution for a real plant is presented. It is implemented in a tank, where the process variable is the water level and the manipulated variable is the valve, with limits $u_{\min} = 0$ and $u_{\min} = 0$. The process to be controlled can be described as:

$$\begin{cases} A = 1 - -0.98585z^{-1} \\ B = -0.00528z^{-1} \end{cases} \quad (12)$$

For modeling the plant, a test by exciting the plant with a pseudo-random binary sequence (PRBS), was performed. The parameters were obtained by least squares identification.

For this model, with $p_{Amr} = 0.98$ and $p_{Amr} = 0.90$, the polynomial controller result as follow:

$$\begin{aligned} R\left(z^{-1}\right) &= 1 - 4.6795z^{-1} + 8.7347z^{-2} - 8.127z^{-3} + \\ &\quad + 3.7677z^{-4} - 0.696z^{-5} \\ S\left(z^{-1}\right) &= -4.795 + 18.7029z^{-1} - 27.3558z^{-2} + \\ &\quad + 17.7826z^{-3} - 4.3347z^{-4} \qquad (13) \\ T\left(z^{-1}\right) &= -0.0015 + 0.0041z^{-1} - 0.0037z^{-2} + \\ &\quad + 0.0011z^{-3} \end{aligned}$$

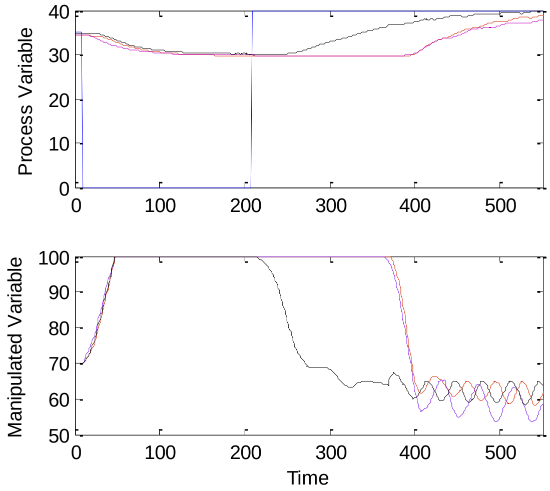


Figure 20: Comparing Pure Integration Feedback, Polynomial Characteristic Observer $W = A_{mr}$ and the proposed anti-windup solution. Poles $p_{Amr} = 0.98$ and $p_{Amr} = 0.90$.

In Figure 20, the response of the physical process is shown for different anti-windup strategies. The setpoint changes in t = 200 seconds and the solution proposed in this paper is the only one with no delay in response.

It should be noted that the manipulated variable has a reaction problem. It is the so-called "stick-slim motion", and it is due to the friction of the actuator that introduces an oscillation in its movement but that is not observed in the process variable.

7. CONCLUSION

This new anti-windup strategy has been successfully analyzed for a STR controller, with either one or two degrees of freedom.

Throughout the analysis of traditional solutions, it was possible to see not only why they are not accurate enough solutions, but also how different terms of the model influence the anti-windup logic, thus gradually leading to the proposed solution.

It was also demonstrated that when the anti-windup is guaranteed to work for Set Point Greater/Less than the Physically Possible Achievable, it does not necessarily guarantee that it also works for great performance requirement (fast reaction), and vice versa. In order to state that the solution works properly, both situations must be verified. The reason for this is that the system is in different physical situations; hence, the process behavior is different and the controller receives other sorts of data.

The solution finally proposed is a simple expression of an internal feedback. In addition to the simulation, it is shown the anti-windup problem and solution for a real process.

This study is relevant to develop different control strategies unlike PIDs. The presented STR method allows the user to effortlessly divide the two degrees of freedom into individual and understandable elements and to add terms without being restricted to any model dimension. More importantly, the saturation is no longer a problem that doesn't allow the use of this kind of algorithms.

REFERENCES

Astrom K. J., Wittenmark B., 1997. Computer-Controlled Systems: Theory and design. New Jersey: Prentice Hall.

Espina J., Arias A., Balcells J., Ortega C., 2009. Speed Anti-Windup PI strategies review for Field Oriented Control of Permanent Magnet Synchronous Machines. 13th European Conference on Power Electronics and Applications, pp. 1-8. September 8-10, Barcelona (Catalonia, Spain)

Huang C., Peng X., Wang J., 2008. Robust Nonlinear PID Controllers for Anti-windup Design of Robot Manipulators with an Uncertain Jacobian Matrix. Acta Automatica Sinica, 34 (9), pp. 1113-1121.

Matijevic M., Sredojevic R., Stojanovic V., 2011. Robust STR Controller Design by Convex Optimization. Electronics, 15 (1), pp. 24-29.

Sahu R., Panda S., Rout U., Sahoo D., 2015. Teaching learning based optimization algorithm for automatic generation control of power system using 2-DOF PID controller. International Journal of Electrical Power & Energy Systems, 77, pp. 287-301.

Tripathi R., Hanamoto T., 2017. Two degrees of freedom dc voltage controller of grid interfaced PV system with optimized gains. International Journal of Electrical Power & Energy Systems, 85, pp. 87-96.

Wang D., Liu T., Sun X., Zhong C., 2016. Discrete-time domain two-degree-of-freedom control design for integration and unstable processes with delay. ISA Transactions, 63, pp. 121-132.

BIOGRAPHY

Lucrecia Gava was born in Buenos Aires, Argentina in July 15th, 1992. She studied Chemical Engineering at the Universidad de Buenos Aires. Since 2016, she belongs to the Control's Lab staff of the same university, where she is doing her graduate thesis; her interest areas are 2-DOF controllers, adaptive control and parameter identification.

Aníbal Zanini was born in Entre Ríos, Argentina in 1954. He studied Electrical Engineering at Universidad Nacional de Rosario and he obtained the Doctor degree at Universidad Politécnica de Madrid. At present, he is professor at the Universidad de Buenos Aires. His areas of interest are Process Control, 2-DOF controllers, adaptive control and parameter identification.

OPTIMUM FUEL CONSUMPTION STRATEGY FOR SERIES-PARALLEL HYBRID ELECTRIC VEHICLES: MODELLING AND APPROACHES

Ivan Miguel Trindade[(a)], Agenor de Toledo Fleury[(b)]

[(a)] AVL Schrick GmbH, Germany
[(b)] Mechanical Engineering Department, Escola Politecnica, University of Sao Paulo, Brazil

[(a)]ivan.trindade@avl.com, [(b)] agenorfleury@usp.br

ABSTRACT

Hybrid Electric Vehicles (HEVs) present a wide range of powertrain configurations, degrees of hybridization and added costs when compared to conventional vehicles. From the point of view of control design, the major challenge is the reduction of fuel consumption and emissions. Methods for determining the energy management strategy that best suits this challenge relies on dynamic optimization techniques that find optimal values for the control variables depending on the system states and on the cost functions.

On the other hand, finding optimal control solutions is useless if the problem is not based on a reliable dynamic model of the system. This paper extends other works by the authors (Trindade, Fleury and Vogelaar, 2014; Trindade and Fleury, 2015) showing the steps for building a very detailed model of a Series-Parallel HEV in MATLAB™ and its identification using data available in the open literature. Afterwards, a Dynamic Programming technique is employed to explore optimal fuel consumption solutions to be compared to the 'non optimal one'. In the first approach, the optimal solution is obtained considering that the Internal Combustion Engine (ICE) follows its best fuel consumption curves and a considerable reduction against the 'non optimal' solution is achieved. Last but not least, in a second approach, the ICE is set not to follow a pre-optimized operating line and this leads to better results when compared to the first optimal approach.

Keywords: hybrid electric vehicle, dynamic programming, optimization

1. INTRODUCTION

While in the powertrain of a conventional vehicle all the driver power demand has to be fulfilled by the engine, in a hybrid vehicle an additional degree of freedom is introduced in the system by the addition of an electric motor. This extra degree of freedom is responsible for enabling hybrid functions, namely electric only propulsion, hybrid generation, hybrid boost and braking energy recuperation. Choosing the appropriate torque demand for electric motor and engine during the cycle is a task of the hybrid control strategy, which has to comply with the necessary performance, emission and fuel consumption requirements. This control strategy is referred in this paper as an Energy Management Strategy with the focus on fuel consumption reduction.

According to the literature, energy management strategies for hybrid powertrains are usually divided into two classes. Desai (2010), Salmasi (2007) and Zhu et al. (2004) classify control strategies into optimal control, which aims to minimizing the fuel consumption by choosing the appropriate control variables, and rule-based control, that controls the powertrain with the use of classic control rules not based on optimization methods. Several authors have shown the use of global optimal control theory applied to a hybrid powertrain, see Sundström (2009), Karbowski et al. (2009), Liu and Peng (2006) and Carignano, Nigro and Junco (2015). Global optimal control is implemented using previous knowledge of the driving route since the cost function aims to optimizing the fuel consumption consumed at the end of the cycle. Therefore, it is not a real-time implementable strategy, but provides the control variables that guarantee that the minimum fuel consumption is achieved.

Different drivetrain configurations for HEVs can accomplish the task of minimizing fuel consumption. The powertrain configuration of this study is a series-parallel electric hybrid which uses a planetary gear set to interconnect the ICE and two electric motors. This paper describes the development of two computational models of the hybrid powertrain: a highly detailed model using heuristic control (Trindade, 2016) and an optimal control model using Dynamic Programming (DP) routine in order to evaluate the full fuel consumption benefit of the system. Two cases of the DP model were simulated: one aiming to optimize only the split of power between battery and fuel; and another were the engine start event and the engine operating points were subject of optimization. The driving cycle FTP75 was used in the simulations since data for this route is available in the literature from dynamometer testing.

Preliminary results of this study (Trindade and Fleury, 2015) have pointed out for the importance of having a small time step size in the dynamic programming routine in order to achieve results comparable to real-world systems. The improved results and the consequent findings will be presented in this paper.

2. DESCRIPTION OF THE POWERTRAIN SYSTEM

The configuration of the powertrain is shown in Figure 1, where a planetary gear set is a transmission component and is used to connect the following components: motor-generator 1 (MG1) to solar gear, ICE to carrier gear and motor-generator 2 (MG2) to ring gear. The ring gear is directly connected to the final gear and to the differential. The ratio of torque amplification from the ICE to the wheels is fixed by the planetary gear ratio from carrier to ring gear.

The planetary gear is known for having two mechanical degrees of freedom. Therefore, different ICE speeds can be realized for a given vehicle speed. This is possible by controlling the MG1 speed on the cost of electrical energy expenditure.

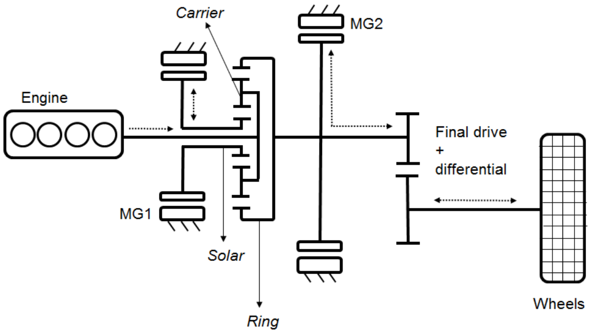


Figure 1: Diagram Of The Power-Split Hybrid Topology With Energy Flow Defined By Arrows.

The 2nd generation of Toyota Prius is the basis from where the main parameters were taken for this study. Data available for this powertrain is available in the literature (Sekimori, 1998; Kamiya, 2006; Abe, 2000). The parameters of the vehicle used in the simulation are shown in Table 1.

Table 1: Characteristics of the vehicle model.

Engine	
Displacement	1.5 dm³
Torque	115 Nm @ 4200 rpm
Power	57 kW @ 5000 rpm
Traction motor (MG2)	
Type	AC Permanent Magnet Motor
Torque	400 Nm
Power	50 kW
Maximum speed	6500 rpm
Generator (MG1)	
Type	AC Permanent Magnet Motor
Power	30 kW
Maximum speed	10000 rpm
Battery	
Type	Ni-MH
Nominal voltage	201.6 V
Rated capacity	6.5 Ah

3. SYSTEM MODELLING

The Series-Parallel configuration shown in Figure 1 allows the system to operate as an electric continuous variable transmission (CVT) since the generator is used to control the ICE speed. The torque relationship in the planetary carrier is fixed by the ratio between the diameters of each gear and a general transmission ratio of the planetary gear set, i_{PGS}, is defined as the ratio between the number of teeth of ring and sun gears. The dynamic equations for the system are shown in Eq.(1) and Eq.(2), where subscripts C, S, R, represent carrier, sun and ring gear parameters. Inertia terms from the planetary are represented by I_S, I_R and I_C, while I_{ICE}/α_{ICE}, I_{MG1}/α_{MG1} and I_{MG2}/α_{MG2} represent inertia and angular acceleration of ICE, MG1 and MG2, respectively. The equivalent rotational inertia of the vehicle mass on the shaft of the ring gear is represented by I_{R-EQ}.

$$T_{MG1} - I_S\alpha_{MG1} = -\left(\frac{1}{i_{PGS}+1}\right)\cdot\left(T_{ICE} - I_C\alpha_{ICE}\right) \tag{1}$$

$$\left(\frac{i}{i_{PGS}+1}\right)\cdot\left(T_e - I_{ICE}\alpha_{ICE}\right) + T_{MG2} - I_R\alpha_{MG2} =$$
$$= \frac{T_{W-R}}{i_{MG2-wheel}} - I_{R-EQ}\alpha_{MG2} \tag{2}$$

The ratio of angular speed between the sun, carrier and ring gear is derived by the equation below, showing the 2 degrees of freedom in the system.

$$(i_{PGS}+1)\cdot\omega_{ICE} = \omega_{MG1} + i_{PGS}\cdot\omega_{MG2} \tag{3}$$

For a given wheel speed, there will be a number of possibilities for engine and generator speeds due to the 2 degrees of freedom. In this way, MG1 will be responsible for controlling the engine speed by the cost of electrical energy.

4. POWERTRAIN PLANT MODEL

This section describes the computational implementation of the different sub-systems of the powertrain plant model.

The software MATLAB/SIMULINK™ was used for the development of the highly detailed powertrain model. This computational model contains the following components:

- Driver model
- Engine
- Battery
- MG1
- MG2
- Transmission

The ICE model does not incorporate thermo-mechanical or combustion phenomena, and therefore, behaviors derived from catalyst and coolant warm-up are neglected. The engine subsystem incorporates a friction model and idle controller and receives an external torque request from the ICE control system. Engine friction is based on the model proposed by Chen and Flynn (1965) where the resistance load is subject to a constant term, and two terms dependent on the rotational speed and its square.

Regarding ICE fuel consumption data, in Duoba, Ng, and Larsen (2000), a torque sensor was added to the engine output shaft and torque measurements were executed in the vehicle at steady state speeds. The results, however, do not cover the whole operation range of the engine, but, instead, only the resulting operating points from the control strategy.

In order to reproduce the efficiency map of the engine, a thermodynamic engine model was created in the GT-Power software and combustion characteristics where calibrated throughout the engine speed and torque in order to result in the Brake Specific Fuel Consumption (BSFC) map shown in Figure 2. The optimum operation line (OOL) for this efficiency map is also shown in the graph.

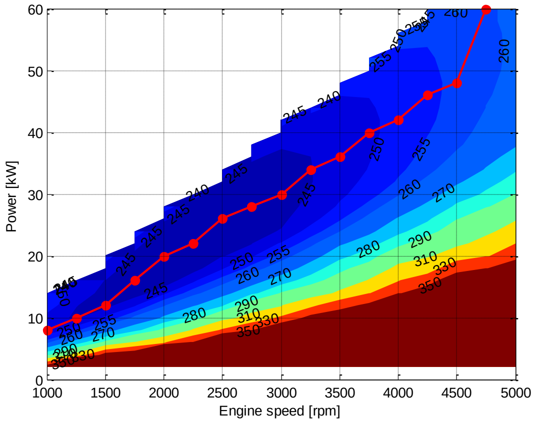


Figure 2: Engine BSFC Map And Optimum Operation Line.

The OOL in Figure 2 produces a fixed correspondence between engine speed and torque and, therefore, it reduces the system from Eq. (3) to only one degree of freedom.

An investigation conducted by Hsu et al. (2005) in order to determine the continuous torque values of the traction motor that produces a limited winding and oil temperature for a certain inlet coolant temperature shows a continuous torque of 167Nm for an inlet coolant temperature of 34.6˚C. Moreover, the peak rated capacity generates a rise in winding temperature of 2.1 °C/s.

The battery model is created using a capacitor as voltage source with internal and parasitic losses. As shown in Ehsani and Emadi (2005), the terminal voltage of such a battery is defined as:

$$V_T = V_{OC} - R_i \cdot I \tag{4}$$

where V_{OC}, R_i and I are the open circuit voltage, internal resistance and terminal current, respectively. The sum of terminal current and leakage current of the battery can be expressed as:

$$I + I_L = -C\frac{dV_{OC}}{dt} \tag{5}$$

where C is the capacitance of the battery. The leakage current is defined by V_{OC} / R_L and, when substituted in Eq. (4) and Eq. (5), it leads to:

$$\frac{dV_{OC}}{dt} = -\left(\frac{V_{OC}}{CR_L} + \frac{I}{C}\right) \tag{6}$$

The resultant battery model in Simulink from this system of equations is shown in Figure 3.

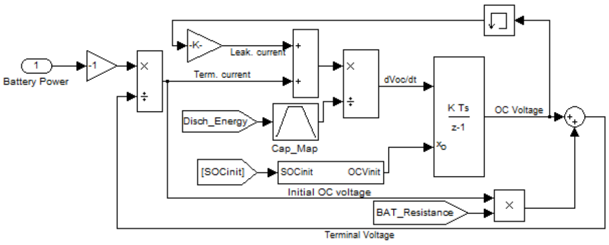


Figure 3: Battery Model In Simulink.

5. DETAILED CONTROL MODEL

This section describes the control system of the detailed powertrain model, which is composed by the following sub-systems:

- State selection
- Torque Demand calculation
- MG1 speed control
- MG2 torque control
- MCI torque control

The model developed in MATLAB™ for the control system contains a high number of sub-systems and block diagrams. It is not possible to provide all the block diagrams in the paper due to lack of available space and, therefore, only some sub-systems were made available in the APPENDIX. For complete details on the full model, please access Trindade (2016) and Trindade, Fleury and Vogelaar (2014).

Since the simulation of the detailed model is explicit, a driver model was developed so that the speed profile of the target driving cycle could be met. A proportional-integral (PI) controller calculates the driver torque demand based on the desired and actual vehicle speeds as shown by the equation below.

$$T_{c,p}(k) = \left(V_{ref}(k) - V(k)\right)\cdot K_{p,c} + K_{i,c}\cdot t_s \sum_{k=0}^{k}\left(V_{ref}(k) - V(k)\right) + T_{ant,p}(k) \qquad (7)$$

Where $T_{c,p}$ is the driver demand torque, V_{ref} is the desired speed, K_p e K_i are the controller gains, t_s is the simulation time step size, k is the instantaneous discrete time and N is the total number of discrete steps. The feed-forward term, T_{ant}, is based on the road resistance forces as shown below. An anti-windup term was incorporated to the driver model in order to prevent torque outputs outside the limits of what the real powertrain can provide.

$$T_{ant,p}(k) = \left(M\cdot r_p{}^2 + I_{TR}\right)\cdot \dot{\omega}_p(k) + \left(0.5\cdot C_d\cdot \rho_{ar}\cdot A\cdot V(k)^2 + c_R\cdot M\cdot g\right)\cdot r_p \qquad (8)$$

Figure 4 shows the difference between target and simulated speed for the NEDC cycle. The absolute difference throughout the cycle is below 0.3 km/h which indicates proper modelling of the sub-system.

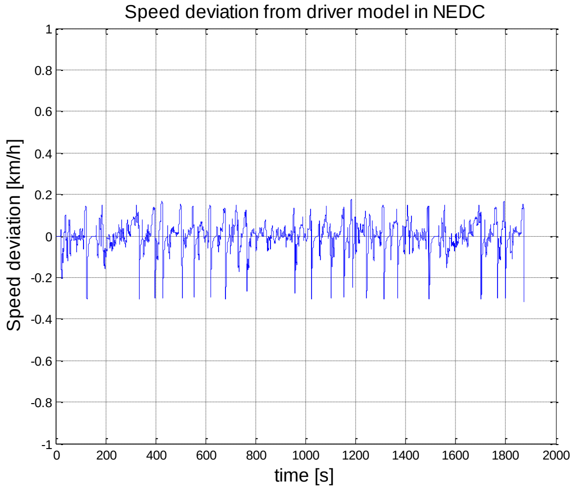


Figure 4: Absolute Speed Error From PID In Driver Model.

The state machine of the controls model defines working state of ICE and MG1. Out of electric only propulsion, the ICE has to be started and, therefore, MG1 has to operate in speed control. In order to achieve a robust control of the system, the control state of MG1 was separated into 3 sub-states: i. Sleep (no speed control); ii. ICE cranking; and iii. ICE optimum speed control. In the state machine, the ICE is activated when the driver power demand is higher than 8 kW or at vehicle speeds higher 60 km/h. This data is originated from Argonne National Laboratory (2013) and is shown in the APPENDIX.

For the speed control of MG1, a certain ICE desired speed, $\omega_{MCI,ref}(k)$ generates a desired MG1 speed, $\omega_{MG1,ref}(k)$, and the controller output will be the desired torque for MG1, $T_{MG1}(k+1)$. A PI controller as shown below was used for the speed control of MG1.

$$T_{MG1}(k+1) = \frac{T_{MCI}(k)}{i_p + 1} + \frac{\left(\omega_{MG1,ref}(k) - \omega_{MG1}(k)\right)}{\tau_{MG1}}\cdot\left(I_{BP} + I_S + I_{MG1}\right) + {} + \left(\omega_{MG1,ref}(k) - \omega_{MG1}(k)\right)\cdot K_{p,MG1} + K_{i,MG1}\cdot\sum_{k=0}^{k}\left(\omega_{MG1,ref}(k) - \omega_{MG1}(k)\right) \qquad (9)$$

Where I_{BP}, I_S e I_{MG1} represent the moment of inertia of carrier gear, sun gear and MG1, respectively, $K_{p,MG1}$ and $K_{i,MG1}$ are the controller gains and τ_{MG1} is a time constant equal to 0.1 s. The first 2 terms on the right of the equation are a feed forward part corresponding to the engine torque being transferred from the carrier to the sun gear and the resistance created by the rotational inertias in the system. The control model also restricts the maximum torque and the maximum desired acceleration of MG1 as shown in the equations below. The model of MG1 control is shown in Figure 13.

$$\frac{\omega_{MG1,ref} - \omega_{MG1}}{\tau_{MG1}} \in [-1000,1000]\,\mathrm{rad/s}^2 \qquad (10)$$

$$T_{MG1} \in [-45,45]\,\mathrm{Nm} \qquad (11)$$

In the ICE control system, there is a sub-module for calculation of the desired optimum engine speed generated by the desired engine torque. The final torque request to the engine is generated by the actual engine speed correlated to the OOL in a sub-module of the control system. The control system of the ICE is shown in Figure 14 of the APPENDIX.

Figure 15 of the APPENDIX shows the control system for MG2. The desired torque for MG2 has to fulfill the difference between driver torque demand and ICE delivered torque. Therefore, during electric only propulsion, when the ICE torque demand is zero, MG2 corresponds to the driver torque demand, while in hybrid mode, the torque request for MG2 corresponds to the extra power necessary for propulsion

6. DETAILED MODEL VS MEASUREMENTS

Data from tests performed with Toyota Prius Gen 3 are available from Argonne National Laboratory (2013). This powertrain configuration has small differences in comparison with the one presented here – it includes a reduction gear between e-motor and differential - the data provides a good base for analyzing the system behavior. These data correspond to chassis dynamometer testing of the vehicle operating under the urban cycle FTP75.

As mentioned before, the simulation runs in explicit mode with the traction torque demand originated set by the driver model. Figure 5 shows vehicle and ICE speed for the simulation and test data results under the FTP75 cycle. No emission strategy is considered in this simulation, which makes the ICE operates in a start stop profile at the beginning of the cycle. Besides this difference against the test data, it is noted a high correlation of the measured and simulated engine speed. The same can be said when one analyzes the battery current signal. Battery current is analyzed in this section as it is directly related to the power-split ration between engine and e-motor.

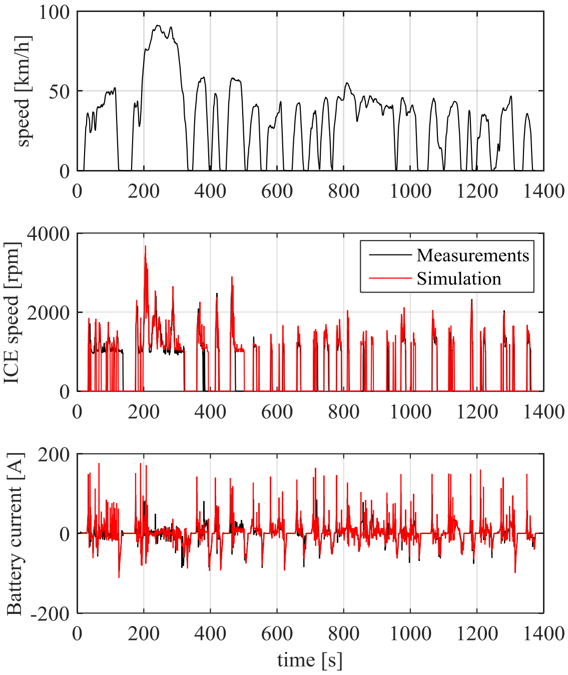


Figure 5: Comparison between detailed model and measurements: vehicle speed, ICE speed and battery current.

7. GLOBAL OPTIMAL CONTROL

Dynamic programming is an optimization algorithm which aims to finding the solution that generates the global minimum result for a given cost function. This means that for a certain driving cycle, the optimized solution will be a vector of control values against time. A time continuous function represents the current system which can be synthetized by:

$$\dot{x}(t) = f(x(t), u(t), t) \tag{12}$$

where u(t) is the control variable, in this case the power-split (PS), and x(t) is the vector of state variables of the system, in this case the battery SOC. The cost function for this system is:

$$J(u(t)) = G(x(t_f)) + \int H(x(t), u(t), t)dt \tag{13}$$

where G(x(t_f)) is the final cost and the second term represents a penalty to ensure that a dynamic constraint should be satisfied, in this case that the SOC at the beginning and at the end are the same. The following cost function represents the fuel consumption in the vehicle over the driving cycle:

$$J(u(t)) = \int \dot{m}_{fuel}(u(t), t)dt + (SOC_{end} - SOC_{ini})p \tag{14}$$

The constraints for the optimization have to be set in order to prevent that the system drift out of its boundaries:

$$T_{\min} < T_{req} < T_{\max} \tag{15}$$

$$SOC_{\min} < SOC < SOC_{\max} \tag{16}$$

$$SOC_{end,\min} < SOC_{end} < SOC_{end,\max} \tag{17}$$

$$u_{\min}(SOC, t) < u(SOC, t) < u(SOC, t)_{\max} \tag{18}$$

where T_{req} are the torque requests in the system for ICE, MG1 and MG2 and SOC_{end} is the SOC value at the end of the cycle.

The DP routine developed by Sundström and Guzzella (2009) was used in this analysis. The range of the PS control variable was divided in 0.1 intervals from [-1, 1], where 1 means pure electric driving, values between 0 and 1 mean electric assist drive and negative values mean hybrid generation.

Two simulation cases were generated using the dynamic programming routine. In the first case (DP 1), the only control variable to be optimized was the power-split. The engine start behavior was set according to Figure 16 and the engine follows the Optimum Operation Line (OOL) as in Figure 2. In the second case (DP 2), the engine start behavior and the engine operating points were also subject of optimization together with the power-split ratio. The intention with DP 2 was to evaluate the interconnection between the different variables and to evaluate if the best results in terms of overall powertrain efficiency are really achieved by having the engine following the OOL.

The SOC possibilities were divided in 61 steps between the 50% and 70%, which represents the usable SOC of the battery. The SOC was allowed to have a variation of -1.6%, which corresponds to the net variation in the test results. This was done in order to have a comparable behavior with the measurements which also improves the comparison of fuel consumption results.

7. RESULTS

The different simulations and the test measurement have different values for the SOC at the end of the cycle and, therefore, it is necessary to correct the final fuel consumption value in order to account for the cost of battery energy. The factor used for the corrections was 340 g of fuel per kWh of battery energy. This value was obtained by observing the average values of engine, transmission, electric motors and battery efficiency during the cycle.

Figure 6 shows the results for the fuel consumption and state variable trajectories during the cycle. The resultant SOC trajectory for the detailed model does not present a high correlation with the measurement data, which shows the limitation of a rule based control strategy in order to replicate a real and complex system. On the other hand, the SOC trajectory of the DP 1 is curiously similar to the test measurements, except for a deviation in the first 350 s of the cycle, which is most probably due to warm-up strategies that have the engine running continuously rather than following the control strategy in the tests.

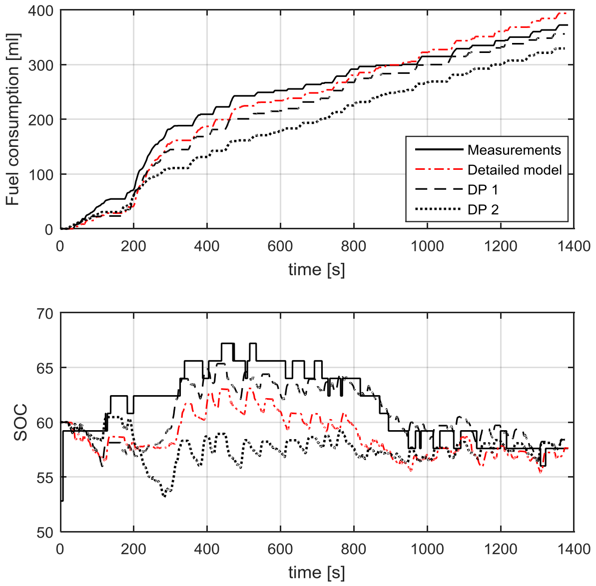


Figure 6 – Fuel consumption (top) and SOC (bottom) trajectories for measurements, detailed model, DP 1 and DP 2.

Figure 7 shows the final fuel consumption for all the simulations and for the test measurement. The detailed model had a final fuel consumption 6.75% above the measurements. This is an expected result as a heuristic control strategy is known for not providing results close to the system minimum. Additionally, from the SOC trajectory in Figure 6, it is strongly believed that the real vehicle has an optimized control strategy since the measurement result was very close to DP 1.

The result for DP 1 was 4.5% below the measured data, which is an acceptable difference since most of the simulation parameters were taken from the literature and the model results could only be compared to measurements on the system level rather than on the component level. The fuel consumption result for DP 2 was 7.5% lower than DP 1, indicating a high efficiency gain due to the extra optimized variables. The reason for the differences will be discussed further in this document.

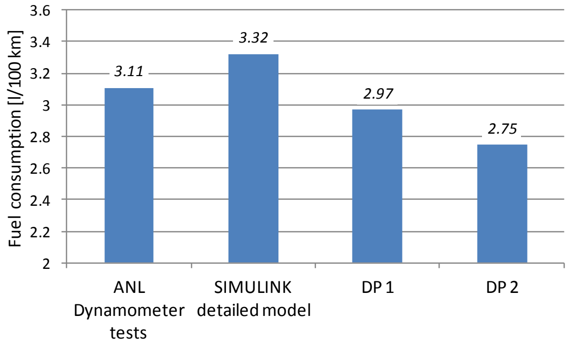


Figure 7 – Fuel consumption results for measurements, detailed model, DP 1 and DP 2.

Figure 8 shows the energy on the different hybrid modes for the different simulations. The "Engine" energy corresponds to the mechanical energy of the ICE, while the energy in EV (pure electric propulsion), Regeneration, Hybrid Boost (or assist) and Hybrid Generation corresponds to the electrical energy at the battery terminal.

From Figure 8, it is noticed that DP 1 prioritises EV driving in comparison with the Detailed model and uses less engine energy. Moreover, DP 1 provides more energy to the battery via Hybrid Generation in order to extend EV driving. Regarding DP 2, there is a trend to use even less "Engine" energy by operating more in "Hybrid Boost" and less in "EV" mode.

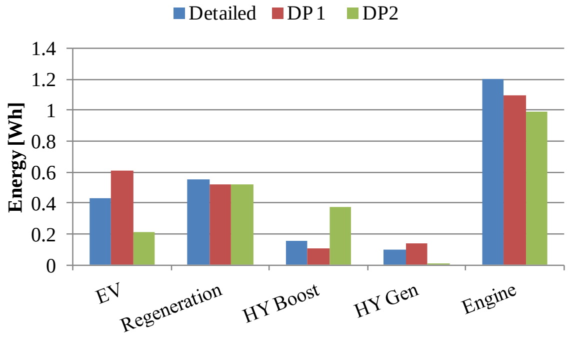


Figure 8 – Result of energy consumption (absolute values) from simulations for the different hybrid modes.

Figure 9 shows the engine start behavior for DP 2. The power demand at which the engine starts is now around 4 kW, instead of 8 kW as seen from the test results and implemented in the detailed mode and in DP 1. By doing this, there are less EV driving events, which explain the low EV energy.

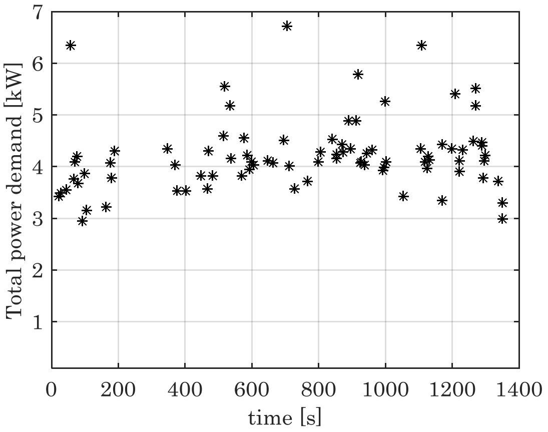


Figure 9 – Engine start behavior from DP 2.

The early engine start from DP 2 was not expected at first since low engine power demands would not directly provide high engine efficiency. The same is valid for the increased energy spent in hybrid boost mode as the hybrid controls would give priority to increase the load on the engine rather than to decrease it. However, this behavior is understood first by looking at Figure 10, which shows the BSFC for the operating points on the OOL. The minimum BSFC (242.5 g/kWh) in the graph is only around 1% better than the BSFC at low 10 kW. This indicates that increasing the load on the engine (hybrid generation) only provides a small increase in engine efficiency and that the increase in overall powertrain efficiency should be lower than zero due to the losses on the electrical path.

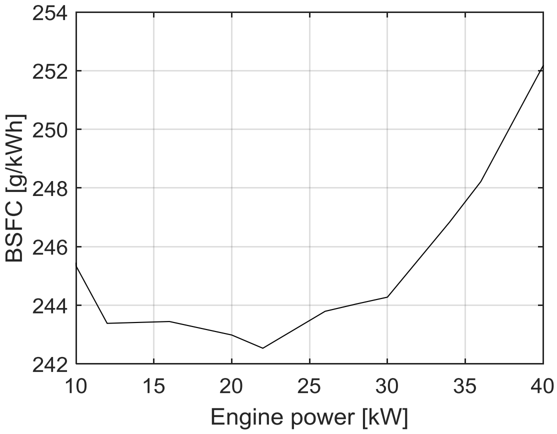


Figure 10 – BSFC of the engine at the Optimum Operation Line (OOL).

Nonetheless, perhaps the greatest advantage from DP 2 is from decreasing the “energy recirculation” in the powertrain, which occurs when electric power is generated at the MG1 to be directly consumed by MG2, or vice-versa. This is a known drawback of series-parallel hybrid with such arrangement of planetary gear. The sum of “EV”, “Hybrid boost” and “Hybrid generation” energy in DP2 is lower than in DP 1, which in turn produces less electric losses. These losses do not have to be overcome by the higher “Engine” energy, increasing the overall powertrain efficiency.

Figure 11 and Figure 12 show the traction torque at the differential input for each hybrid mode. In Figure 11, the hybrid strategy results in many operating points in hybrid generation mode, which seems to be located in on the transition between EV and hybrid boost mode. This is partly due to the fact that the ICE has to follow the OOL and partly due to the high ICE start threshold, which creates the need for hybrid generation as the recuperation energy is not enough to assure charge sustaining mode. Hybrid boost mode around 50 km/h and 50 Nm were identified where the battery SOC was 3% above the target, which explains the choice of spending electrical energy even at low power demands.

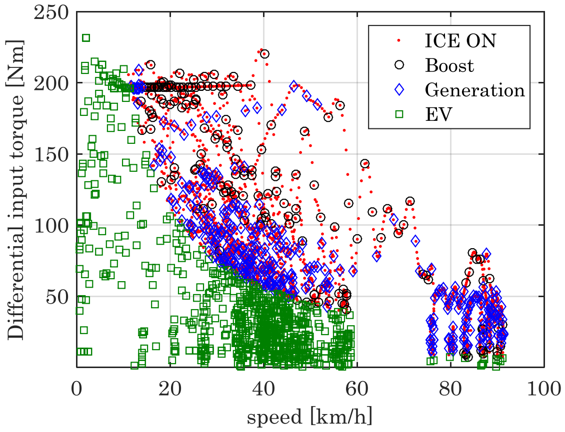


Figure 11 - Torque at the differential input shaft during traction events for different hybrid modes in DP 2.

The lower ICE ON power threshold (transition from EV to ICE ON) from DP 2 is seen in Figure 12, which also shows a much lower amount of hybrid generation points than DP 1 as indicated in Figure 8. The control strategy in this case prioritizes starting the engine earlier and use more electrical energy for propulsion in “Hybrid boost” rather than in “EV” mode. The reason for this may be to use fuel energy, since the engine efficiency would already be high enough, and also use electrical energy, since it is widely available from recuperation events.

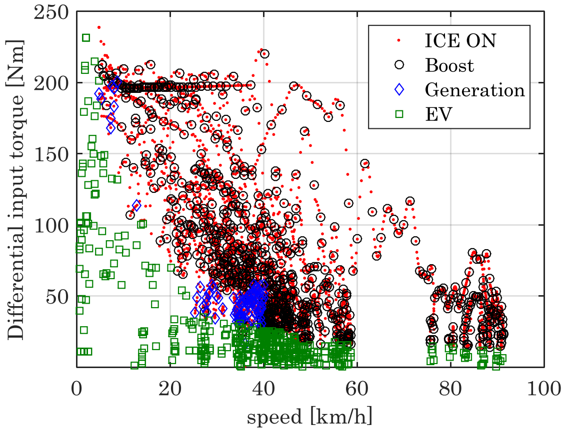


Figure 12 – Torque at the differential input shaft during traction events for different hybrid modes in DP 1.

CONCLUSION

This paper showed the development of two different powertrain models for a Series-Parallel hybrid electric vehicle: 1. A detailed powertrain model using heuristic control laws and 2. A model using global optimum control implemented via a dynamic programming (DP) routine. The DP model simulation comprised two different cases aiming to further explore the total fuel economy potential of the system by finding optimal values for additional control variables. Chassis dynamometer data from the baseline vehicle was also used to assist the development of the models.

From the results of the first simulation case of the DP routine (DP 1), it was identified in Figure 6 that the real vehicle has a very similar response to a globally optimized system. This points out that the fuel consumption deviation of 6.75% between the detailed model against the real vehicle is acceptable, since the control in the simulation uses a rule based approach.

From the measurement data, it was identified that the real vehicle controls the engine by having it operating on its optimum operating line (OOL) in order to maximize engine efficiency. This strategy was implemented in the detailed model and in DP 1, however, differently from DP 1, in the second case of the DP routine (DP 2), the engine operation was set not to follow the optimum operating line but was free to be optimized. Besides that, the engine start threshold was also subject of optimization. At the end, the fuel consumption result from DP 2 was 7.5% lower than DP 1, showing that optimizing the system for overall powertrain efficiency provides an extra fuel consumption benefit against an optimized system focusing on engine efficiency. This improved result was achieved by actively reducing the losses on the electrical path, consequently having less load cycles in the battery.

APPENDIX

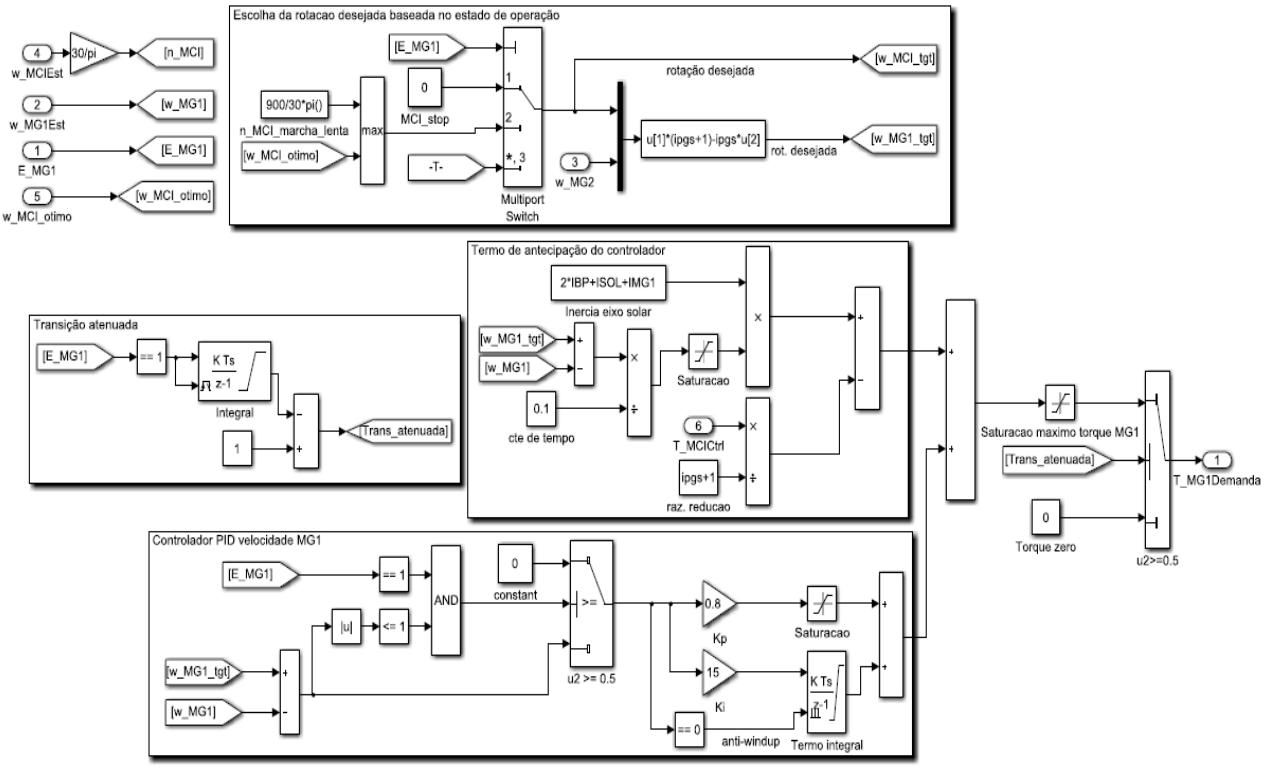


Figure 13 – MG1 control system.

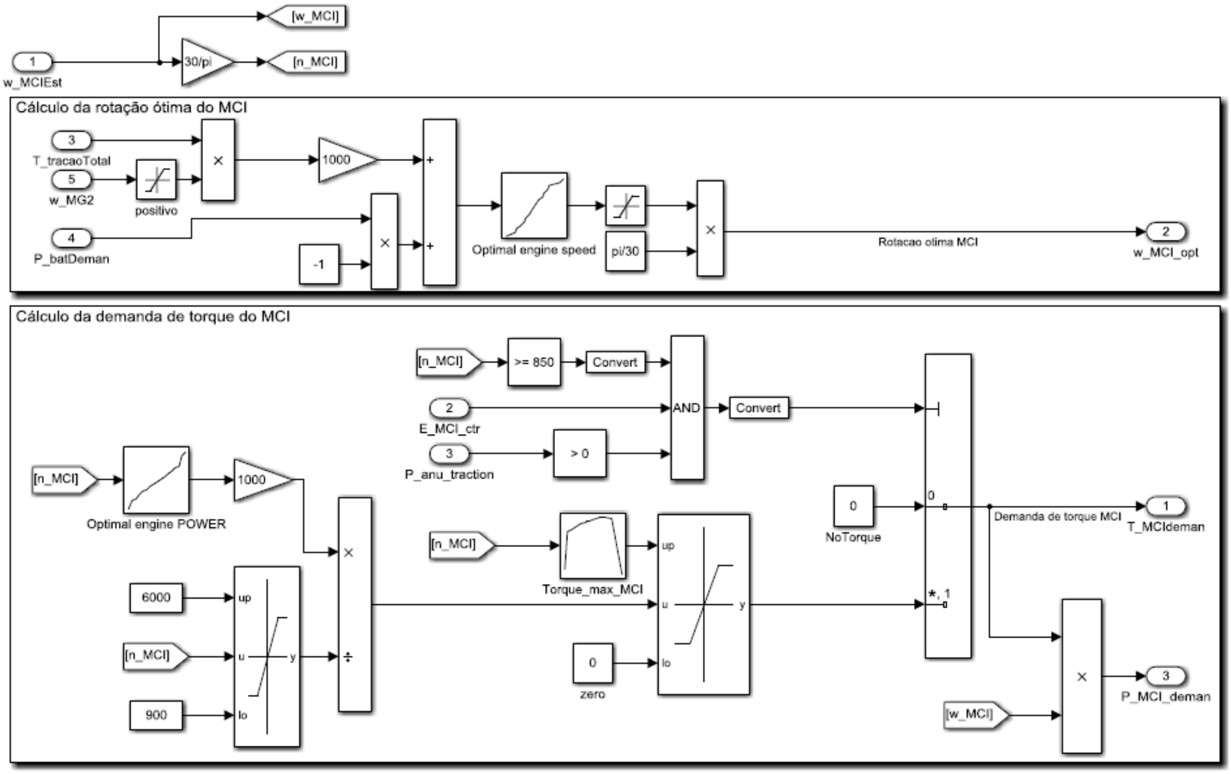


Figure 14 – Control system of the ICE.

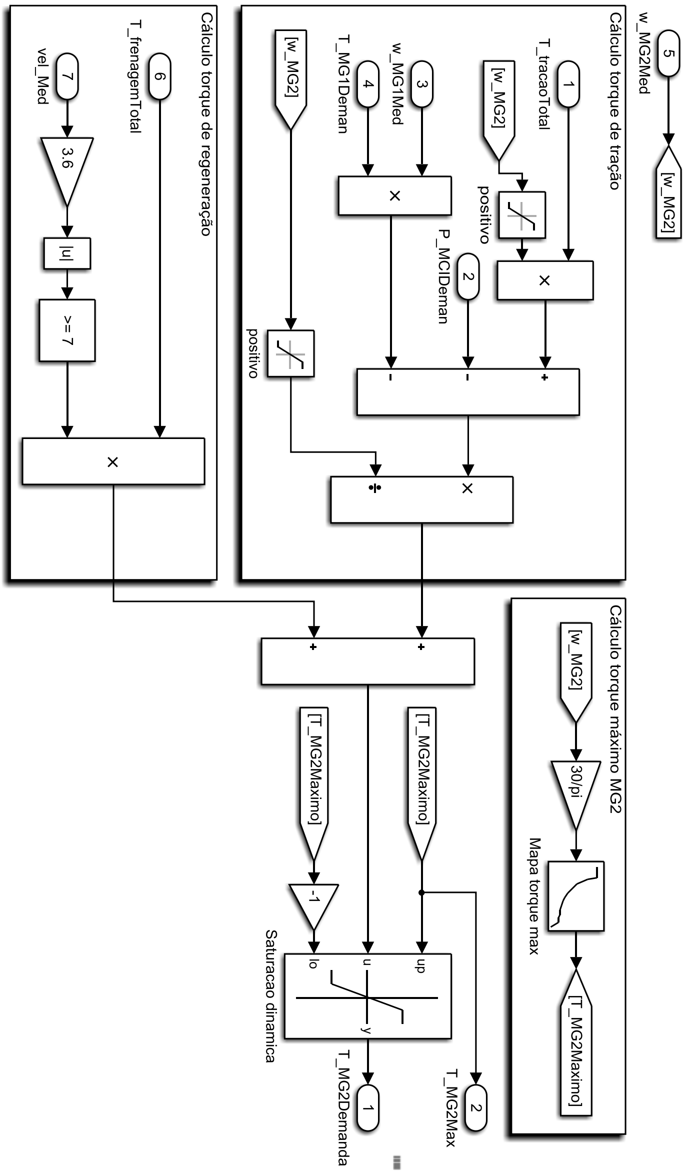


Figure 15 – MG2 control system.

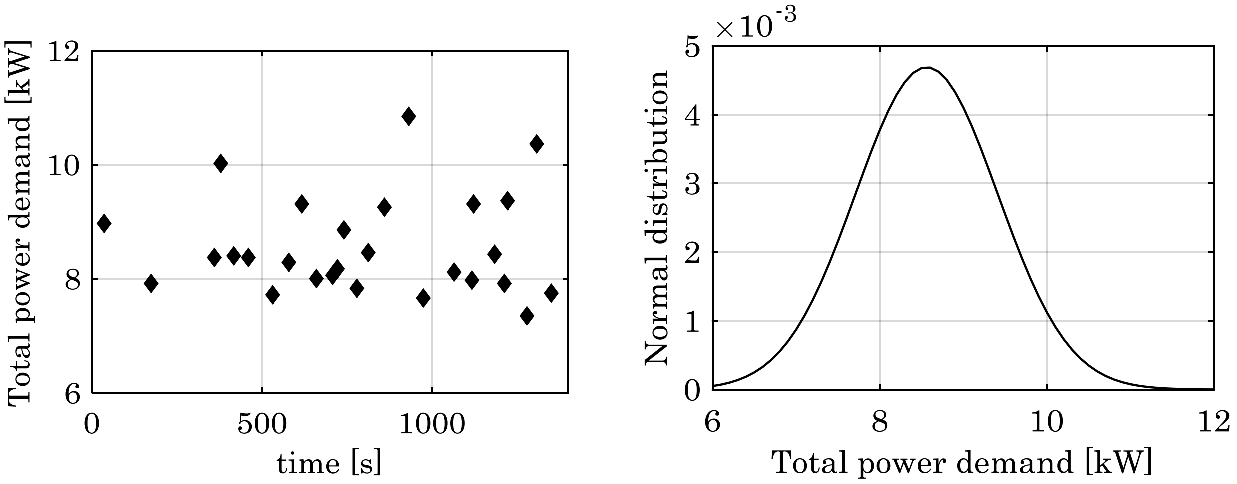


Figure 16 - Test results for engine start behavior (Argonne National Laboratory, 2015).

REFERENCES

Abe S., 2000. Development of the Hybrid Vehicle and its Future Expectation. SAE Technical Paper 2000-01-C042.

Argonne National Laboratory, 2013. All Data- 2010 Toyota Prius, Available from: http://www.anl.gov/energy-systems/group/downloadable-dynamometer-database/hybrid-electric-vehicles/2010-toyota-prius, [Accessed 17 February 2015].

Ayers C., Hsu J., Marlino L., Miller C., Ott Jr.G., Oland C., 2004. Evaluation of 2004 Toyota Prius Hybrid Electric Drive System Interim Report. ORNL/TM-2004/247.

CarignanoI. M. G., Nigro, N. M., Junco, S., 2015. Hybridization Effect On Fuel Consumption and Optimal Sizing of Components For HEV. In The 8th. International Conference on Integrated Modeling and Analysis in Applied Control and Automation (IMAACA 2015). Begerggi, Italy, pp. 48–54.

Chen S. and Flynn P., 1965. Development of a Single Cylinder Compression Ignition Research Engine. SAE Technical Paper 650733.

Delprat S., Lauber J., Guerra T., Rimaux J., 2004. Control of a parallel hybrid powertrain: Optimal control. IEEE Transactions on Vehicular Technology, 53(3), 872–881.

Duoba M., Ng H., Larsen R., 2000. In-Situ Mapping and Analysis of the Toyota Prius HEV Engine. SAE Technical Paper 2000-01-3096.

Duoba M., Ng, Henry, Larse R., 2001. Characterization and Comparison of Two Hybrid Electric Vehicles (HEVs) – Honda Insight and Toyota Prius. SAE Techinical Paper 2001-01-1335.

Ehsani M., Gao Y., Emadi A., 2005. Modern Electric, Hybrid Electric, and Fuel Cell Vehicles, Fundamentals, Theory, and Design. 2nd ed. Florida: CRC Press.

Gray T., Shirk M., 2013. Toyota Prius VIN 0462 Hybrid Electric Vehicle Battery Test Results. Idaho National Laboratory INL/EXT-13-28025.

Hsu J.; Nelson S., Jallouk P., Ayers C., Campbell S., Coomer C., Lowe K., Burress T., 2005. Report on Toyota Prius motor thermal management. Oak Ridge National Laboratory, ORNL/TM-2005/33.

Kim N., Rousseau A., Rask, E., 2012. Autonomie Model Validation with Test Data for 2010 Toyota Prius. SAE Technical Paper 2012-01-1040.

Lin C., Peng H., Grizzle J., Kang J., 2003. Power management strategy for a parallel hybrid electric truck. IEEE Trans. Control Systems Technol., 11(6), 839–849.

M. Kamiya, 2006. Development of traction drive motors for the toyota hybrid system. IEEE Transactions on Industry Applications, 126(4), 473-479.

Muta K., Yamazaki M., Tokieda J., 2004. Development of New-Generation Hybrid System THS II - Drastic Improvement of Power Performance and Fuel Economy. SAE Technical Paper 2004-01-0064.

Rask E., Duoba M., Busch H., Bocci D., 2010. Model Year 2010 (Gen 3) Toyota Prius Level-1 Testing Report. Argonne National Laboratory. Report ANL/ES/RP-67317.

Sekimori T., 1998. Development of Toyota's Electric and Hybrid Vehicle. SAE Technical Paper 98C053, 355-361.

Sundström O., Guzzella L., 2009. A Generic Dynamic Programming Matlab Function. Proceedings of the 18th IEEE International Conference on Control Applications, 1625–1630, Saint Petersburg, Russia.

Trindade I, 2016. Modelagem, Controle e Otimização do Consumo de Combustível Para Um Veículo Híbrido Elétrico do Tipo Serie-Paralelo. Master’s Thesis. Polytechnic School, University of Sao Paulo. Available from: www.teses.usp.br.

Trindade I., Fleury, A., 2015. Modeling, control and application of dynamic programming to a series-parallel hybrid electric vehicle. In The 8th. International Conference on Integrated Modeling and Analysis in Applied Control and Automation (IMAACA 2015). Begerggi, Italy, pp. 71–78.

Trindade I., Fleury, A., Vogelaar, G.-J., 2014. Modeling, Simulation and Analysis of Operation Modes in a Series-Parallel Hybrid Electric Powertrain With Torque-Split Device. SAE Technical Paper 2014-36-0351, Sao Paulo.

AUTHORS BIOGRAPHY

Ivan Miguel Trindade has a Master’s degree from Polytechnic School of University of São Paulo and a degree in mechanical engineering (2008) from the same university. He has worked on various projects related to internal combustion engines and hybrid powertrain development. He currently works for AVL Schrick GmbH covering a variety of tasks involving hybridization and electrification of automotive powertrain.

Agenor de Toledo Fleury has a degree in mechanical engineering from ITA - Technological Institute of Aeronautics (1973), a MSc (1978) and a PhD degree (1985) in Mechanical Engineering from the University of São Paulo. He is currently an assistant professor at Polytechnic School, University of Sao Paulo. He has previously served the Brazilian Institute for Space Research (INPE), the Brazilian Aeronautic Enterprise (EMBRAER), the Sao Paulo State Institute for Technological Research (IPT) and FEI University, leading various projects with emphasis on Dynamics and Control Systems. His most recent projects address modeling and control of nonlinear systems, optimal control and estimation, in applications of Biomechanics, Robotics and Automotive Engineering.

PHM ORIENTED BEHAVIOR MODELING FOR PEM FUEL CELL SYSTEMS VIA DELAYED FEEDBACK RESERVOIR COMPUTING MODEL

Zhongliang Li, Rachid Outbib

LSIS Laboratory, UMR CNRS 6168, Aix-Marseille University

zhongliang.li@lsis.org, rachid.outbib@lsis.org

ABSTRACT

In this paper, a framework named *model space* based PHM design is proposed for PEM fuel cell systems. The objective is to address the PHM problem in consideration of the various operating conditions and system dynamics. In this frame, modeling of fuel cell systems is realized via a type of black box model named Delayed Feedback Reservoir Computing Model (DFRCM). The modeling and the PHM performance is validated using experimental data.

I. INTRODUCTION

Knowing that reliability and durability are two key criteria for commercialization of Polymer Electrolyte Membrane Fuel Cell (PEMFC) technologies, efforts have been taken not only to improve the fuel cell (FC) design and assembly, but also to optimize system operations. This necessitates the availability of automatic detection and isolation of the faults and reconfiguration of the control system accordingly. Recently, topics on Prognostics and Health Management (PHM) of PEMFC systems have attracted increasing attention of both academic and industrial communities.

Fault diagnosis and prognosis are two main elements in the PHM cycle. Fault diagnosis is deserved to detecting and isolating the faults that occur at different parts of the system. While the main goal of prognosis is to estimate Remaining Useful Life (RUL) and associate a confidence interval [1]. A main branch of theories on diagnosis and prognosis methods are built with the assumption that the system process model is available. The fault diagnosis can be realized by comparing the measured physical variables with the ones that calculated with the model and a part of measured data. While, prognosis can be realized by evaluating the variation of some crucial parameters of the system model.

PEMFC systems are undoubtedly complex and nonlinear ones which involve the phenomenons of electro-chemistry, fluid mechanics, thermodynamics. Modeling an arbitrary PEMFC system with first principle is always considered as a tough task. As well, various parameters, structures and control laws exist for different FC designs and applications. Some well-established models, such as the one proposed in [2], may not be well fitted for other systems with different system parameters, constructions and control laws. The adaptation of a model form for an alternative FC system is not a trivial task.

Meeting the above mentioned difficulties on first principle modeling, the researchers have been trying to analyze some crucial variables by adopting advanced signal processing and machine learning techniques to solve the PHM problems, meanwhile avoid the modeling process. For instance, concerning fault diagnosis aspect, Zheng et al. [3] propose to realize the fault diagnosis via analyzing Electrochemical Impedance Spectroscopy (EIS) with some machine learning tools. Benouioua et al. [4] achieve fault diagnosis by analyzing voltage signal via wavelet transformation combined with multifractal formalism. Concerning prognosis aspect, method named Echo State Network is used to simulate and predict the trajectory of FC voltage degradation [5].

The common points of the above work is that the FC stack is assumed to operate at a constant or quasi constant operating conditions. In such conditions, the factors which affect the variation of the FC behaviors are limited to the faults and ageing effect. However, the assumption seems to be hard to arrive in practical cases. First, the FC systems are usually operated in a varied operating condition with dynamic and transitional processes. The variation of FC behaviors can be affected by changing operating condition other than faults or ageing effect. Second, it has been found that the degradation mechanism is highly dependent on operating conditions. The operations such as dynamic load cycling, startup/shutdown significantly accelerate the degradation rate of FCs. Hence, PHM strategy should be designed in consideration of the different operating conditions and dynamic processes.

To conquer the above mentioned problem, a framework of PHM is proposed in this paper. In the framework, fault diagnosis and prognosis are not investigated in the data space but in the model space. The inherent idea is that, the data sampled in normal state with dynamic process in wide operating range can be described by a behavior model. The parameters of the model is constant if there are no faults and ageing effect. On contrary, some faults or degradations related to ageing effect are detected if the parameters of the model are changed to some degree.

Considering that a process model can probably well describe the dynamic process in a wide operating range, and a sufficiently accuracy, generalized, but not sophistical first-principle model is hard or even impossible to obtain for PEMFC systems, the model selected to describe the FC stack is a recently proposed model. The model, inherently belongs to black-box model class, is named Delayed Feedback Reservoir Computing

Model (DFRCM) in this study. The main advantages of this modeling technique are that the model fitting process can be efficiently performed. Meanwhile, the high modeling precision can be maintained.

In the rest of the paper, the frame work of *model space* based PHM strategy is introduced firstly in Section II. Then, the main effort is taken to presenting the technique DFRCM in Section III and its application for PHM oriented PEMFC system modeling in Section IV. Finally, the study is concluded in Section V.

II. FRAMEWORK OF MODEL SPACE BASED PHM STRATEGY

Traditionally, PHM is realized in the data space. In this paper, the framework in which the PHM is realized in the model space instead of data space is proposed. As shown in Fig. 1, the model space based PHM strategy is divided into learning phase and implementing phase. In the learning phace, the historical data including input and output ones are firstly segmented into a series of data segments. Then, using these data segments, a series of models with the same form can be fitted. Therefore, the model parameters corresponding to the data segments can be obtained. In the implementing phase, fault diagnosis and prognosis can be implemented based on the obtained system parameters. For instance, the fault detection problem can be considered as a outlier detection problem [6]. While prognosis can be considered as a curve fitting or regression problem.

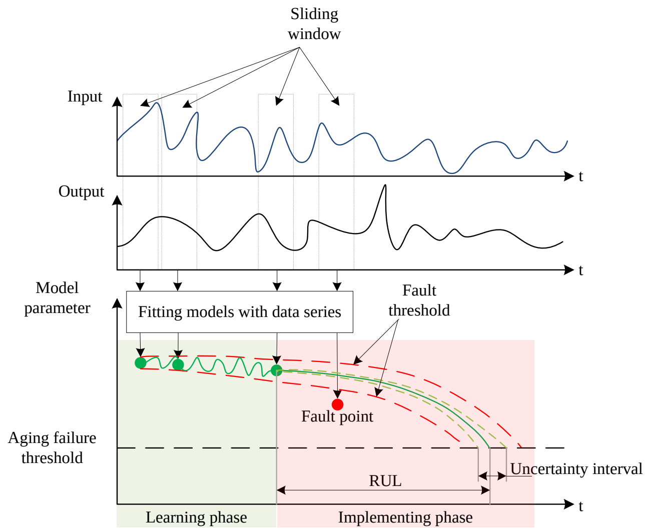


Fig. 1. Schematic of model space based PHM

The benefits of this framework can be summarized as follows:

- In a real FC system, a variety of dynamic processes exist at the moments of starting up, shutting down, increasing and decreasing the load. The data sampled in dynamic processes are not time-independent. Dynamic processes are considered in this framework though fitting the models with time series.
- The models are fitted using both input and output data. Therefore, the impacts of input variables other than faults and ageing effect are taken into account.

III. DELAYED FEEDBACK RESERVOIR COMPUTING MODEL

Recurrent neural network (RNN) has been considered as a powerful tool to model a nonlinear and dynamic system, as it can exhibit a nonlinear dynamical temporal behavior. At the early of the last decade, a new paradigm, called Reservoir Computing (RC) emerges which solves the training problems of RNN effectively from a new perspective [7].

More recently, a simple structure consisting a single non linear node and a delay line, i.e. DFRCM in this paper, is proposed to implement RC experimentally via optoelectronic tools [8]. The results proposed in [8] show that the very simple structure has high-level information-processing capabilities in both dynamic system modeling and pattern classification aspects. Taking into account the characteristics of simple structure and high performance in modeling dynamical systems, DFRCM is selected to project the data into model space.

A. Principle of DFRCM

The general scheme of DFRCM is shown in Fig. 2. In the structure, a nonlinear node with delayed feedback is used. A reservoir is obtained by dividing the delay loop into N intervals and using time multiplexing. The input states are sampled and held for a duration τ_D, where τ_D is the delay in the feedback loop. For any time, the input state is multiplied with a mask, resulting in a temporal input stream $J(t)$ that is added to the delayed state of the reservoir $x(t-\tau_D)$ and then fed into the nonlinear node. The output is calculated as the weighted sum of the state variables [8].

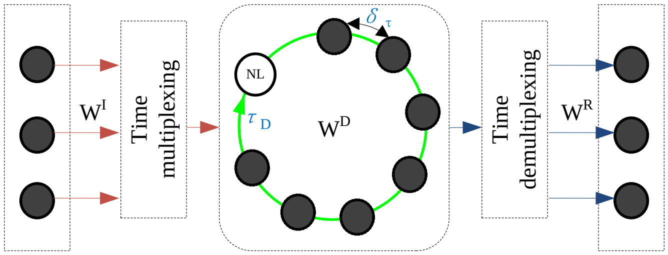


Fig. 2. Illustration of basic RC structure [8]

B. Steps to realize a general DFRCM

Step 1: Time multiplexing

The input stream $u(k)$ undergoes a sample-and-hold operation, resulting in a stream $I(t)$ that is constant in time delay τ_D, as

$$I(t) = u(k) \in \mathcal{R}^{M\times 1} \quad \text{for } \tau_D k \leq t < \tau_D(k+1) \tag{1}$$

Within one delay of τ_D, N_{node} virtual nodes are defined. τ_D is separated into N_{node} sub time interval, denoted as $\delta\tau_D = \tau_D/N_{node}$.

Defining periodic mask function, as:

$$M(t) = W_{I,i} \in \mathcal{R}^{1\times M},\ i = 1, \ldots, N_{node} \quad (2)$$

for $(i-1)\delta\tau_D \leq t < i\delta\tau_D$ and $M(t+\tau_D) = M(t)$. Generally, the values of $W_{I,i}$ are chosen randomly from some probability distribution.

Then, the value to be injected into the reservoir is given by

$$J(t) = M(t) \cdot I(t) \quad (3)$$

Step 2: States calculation

In the reservoir, the nonlinear evolution equation is implemented. In this paper, referring the proposal of [9], the nonlinear equation of the following form is considered:

$$\tau\frac{dx(t)}{dt} + x(t) = \beta sin^2[\alpha x(t-\tau_D) + \gamma J(t) + \Phi] \quad (4)$$

where τ is the internal characteristic time scale of the nonlinear dynamic; $x(t)$ is the state variable of the reservoir; β, α, γ, Φ are parameters to be initialized.

The equation can be realized using the ronge-kutta method (RK4). The integration time step, denoted as h, is a fraction of response time τ.

For ith virtual node of the kth discrete reservoir, the state is given by

$$x_i(k) = x\left(k\tau_D - (N-i)\delta\tau_D\right) \quad (5)$$

the states of all the nodes in kth discrete reservoir is collectively expressed as $\boldsymbol{x}(k) = [x_1(k), \ldots, x_{N_{node}}(k)]^T$.

Step 3: Output calculation

To

$$y(k) = W_O\boldsymbol{x}(k) = \sum_{i=1}^{N_{node}} w_i \cdot x_i(k) \quad (6)$$

where $w_i \in \mathcal{R}^{L\times 1}$ is the element of *output weight matrix* $W_O = [w_1, \ldots, w_{N_{node}}] \in \mathcal{R}^{L\times N_{node}}$.

So-called *ridge regression* is used to pursue the weight matrix in the training process. W_O is obtained through

$$W_O = (X_{train}^T X_{train} + \lambda I)^{-1} X_{train}^T Y_{train} \quad (7)$$

where I is N_{node} order unit matrix,

$$X_{train} = \begin{bmatrix} \boldsymbol{x}(1)^T \\ \vdots \\ \boldsymbol{x}(N_{train})^T \end{bmatrix}$$

$$Y_{train} = \begin{bmatrix} \boldsymbol{y}(1)^T \\ \vdots \\ \boldsymbol{y}(N_{train})^T \end{bmatrix}$$

DFRCM can be summarized as Algorithm 1.

Algorithm 1 DFRCM

```
Training:
1: Collect $N_{train}$ input and output data, i.e. $\boldsymbol{u}(1)$, $\boldsymbol{u}(2)$, ..., $\boldsymbol{u}(N_{train})$, and $\boldsymbol{y}(1)$, $\boldsymbol{y}(2)$, ..., $\boldsymbol{y}(N_{train})$.
2: Initialize parameters $\beta$, $\alpha$, $\gamma$, $\Phi$, $\tau_D$, $N_{node}$, $h$, $\tau$, $\lambda$. Among them, $\tau$ is set as the base value, $h$ and $\tau_D$ are set with respect to $\tau$.
3: Define mask function $M(t)$ according to (2).
4: Calculate $J(t)$ according to (3).
5: Initial $x(t)$, $0 \leq t < \tau_D$.
6: Calculate $X_{train}$ according to (4) by using RK4 method.

7: Calculate $W_O$ according to (7).
Performing:
1: For a new input series $\boldsymbol{u}(1)$, ..., $\boldsymbol{u}(N_p)$.
2: Repeat step 2 to step 6 in training procedure. $X_{N_p}$ is obtained.
3: Calculate $\boldsymbol{y}(1)$, ..., $\boldsymbol{y}(N_p)$ using $X_{N_p}$ and $W_O$ according to (6).
```

C. Remarks

Algorithm 1 presents the training and performing procedures to achieve a general modeling. In this study, DFRCM is adopted not for a traditional modeling goal but for PHM. To be specific, the parameters of output matrix W_O obtained in training procedure are considered as the variables for diagnosis and prognosis goals.

IV. APPLICATION OF DFRCM FOR PEMFC SYSTEM

In this section, the DFRCM is applied for real PEMFC systems in two aspects. In the first aspect, DFRCM is used for modeling the dynamic behaviors of a commercial PEMFC system. In the second aspect, DFRCM is used to extract the PHM oriented features.

A. Example 1: Modeling dynamic profiles for a commercial PEMFC system

The concerned PEMFC system is a commercial air-cooled 1.2-kW Ballard NEXA system. This stack is supplied by compressed air and hydrogen and is cooled with air fans. The detailed technique parameters can be found in [10]. From the test bench, the stack voltage, stack temperature, load current, reactant air flow rate, cooling air flow rate, and environment temperature can be measured. A DC electronic load is used to simulate an arbitrary current profile with abundant dynamic processes.

In the experiment, a dynamic current cycle during 522 s was produced using the electronic load. The current form is shown in Fig. 1. To construct the DFRCM for this cycle, the environment temperature, current, cooling air flow, and stack temperature are considered as the input variables. While the reactant air flow and stack voltage are seen as the output variables. Actually, the reactant air flow is regulated with respect to the current value. The stack voltage is normally considered as the output variable of the whole system.

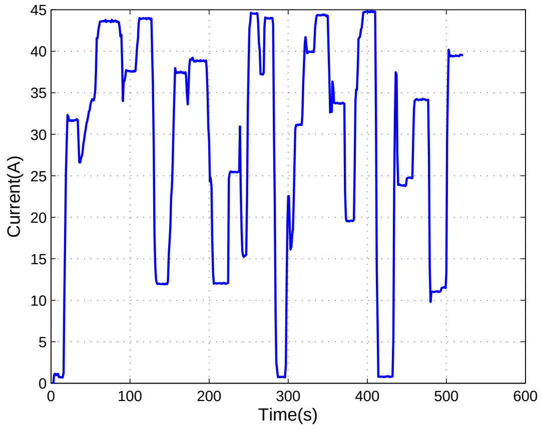


Fig. 3. Load current profile

The first 312 samples are used for training, while the rest 211 samples are used to test the trained model. The parameters used for the modeling procedure are summarized in Table I.

TABLE I
PARAMETERS USED FOR MODELING NEXA PEMFC SYSTEM

β	α	γ	Φ	τ_D	N_{node}	h	λ
0.85	1	0.5	0.76	160	400	0.02	5×10^{-6}

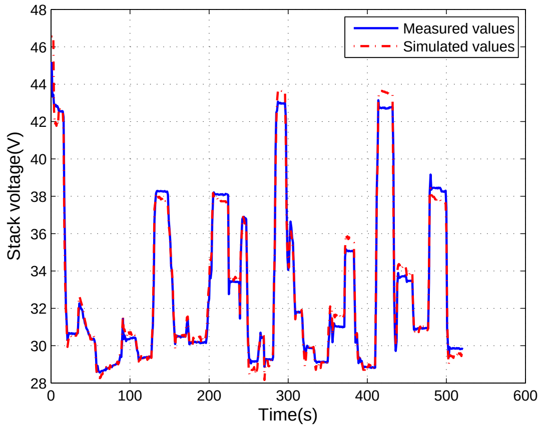


Fig. 4. Comparison between measured and simulated stack voltages

The real stack voltage response and the model results are illustrated in Fig. 3. The proposed fuel cell stack model demonstrates great accuracy with regard to the experimental measurements. For the entire stack operating range, the maximum relative error in voltage is less than 5%. The voltage dynamic behavior is well reproduced by the model.

B. Example 2: model based PHM

1) Experimental setting: The concerned fuel cell stack is designed with the structure of open cathode and dead-end anode. Some crucial parameters are listed in Table II.

TABLE II
PARAMENTERS OF THE INVESTIGATED FUEL CELL STACK

Fuel cell type	Open cathode/Dead-end anode
Nominal pressure at hydrogen inlet	0.35 bar
Number of cells	15
Nominal output current	8 A
Nominal output power	84 W
Maximum temperature	75 °C
Maximum current	13.45 A
Lowest permitted stack voltage	7.5 V

A current profile obtained from the real motive application is simulated thanks to the programmable DC load. A length of current shape is shown in Fig. 5. It can be seen that the current varies between 0 and 8 A.

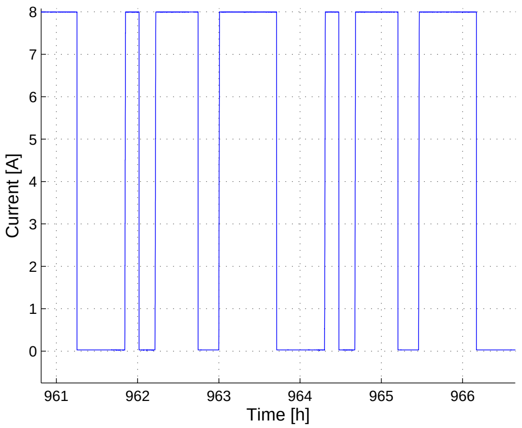


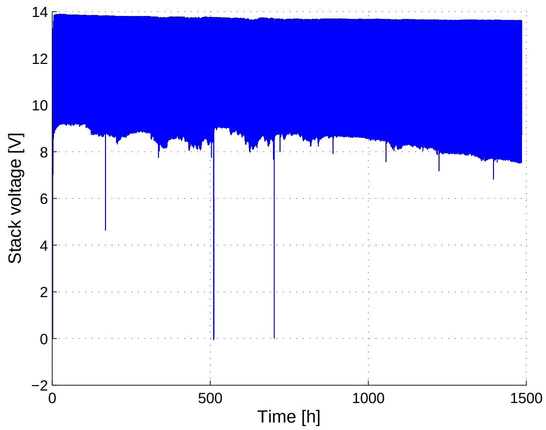
Fig. 5. Current profile

Stack voltage, individual cell voltages (cell voltages 1 to 15), temperatures (temperatures of the positions cell 1, cell 8, and cell 15), current. Sample frequency is 5 Hz.

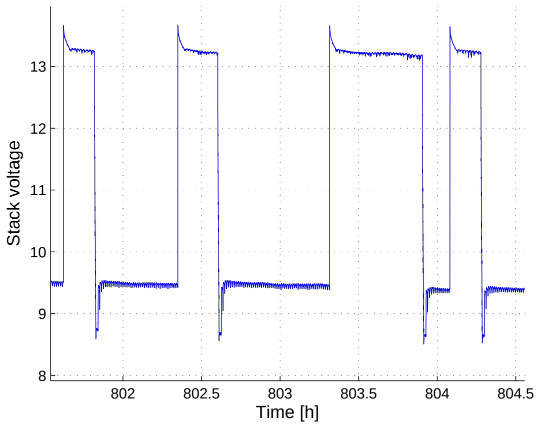
The data during the 1500 h test were recorded. The overview of the stack voltage evolution is shown in Fig. 6(a). Some details can be seen in Fig. 6(b). These stops are considered as the environment disturbances in this study.

2) DFRCM model identification: The whole data are divided into 1000 equal segments. Thus, the duration of each segment is 1.5 h. For each data segment, a DFRCM model identification is implemented. The model input is defined as the current values of present sample and last two samples ($I(k)$, $I(k-1)$ and $I(k-2)$). After model identification procedure, the output is defined as the current stack voltage ($V(k)$). The model identification result, i.e. the model output and real measurement, in a segment is shown in Fig. 7. It is seen that the model fits the system behavior well.

3) Prognosis implemented in model space: With the identified model, the prognosis can be implemented in this model space. Here, by setting the input as constant, the corresponding steady state output voltage can be reconstructed. The output in nominal condition can thus be pursued and considered as the prognosis oriented feature. Based on the experimental data,



(a) Stack voltage evolution during the whole test



(b) Details of the stack voltage evolution

Fig. 6. Stack voltage evolution

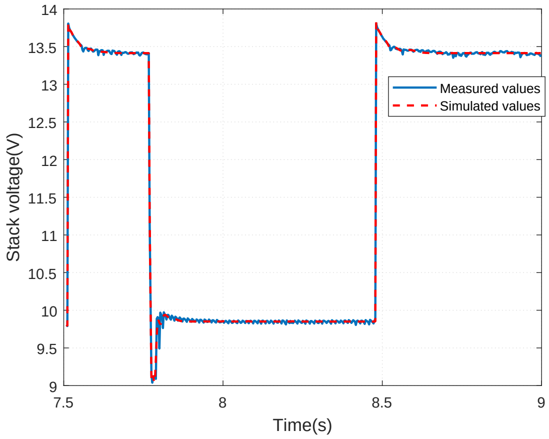


Fig. 7.

the calculated nominal stack voltage versus time is shown as the blue line in Fig. 8 and Fig. 9.

With the extracted nominal stack voltage evolution, the prognosis can be implemented by creating a time series model and using the model for prediction. Here, the time series model is a 2nd order state-space system with noise input. The implementation details can be found in [11].

Fig. 8 and Fig. 9 show the mean prognosis results. The prediction horizon in Fig. 8 is 750 h, while the one in Fig. 9 is 300 h. The bounders which correspond to the 99.7 % confidence interval. It can be seen that the prediction values well fit the real signal evaluation in the two cases.

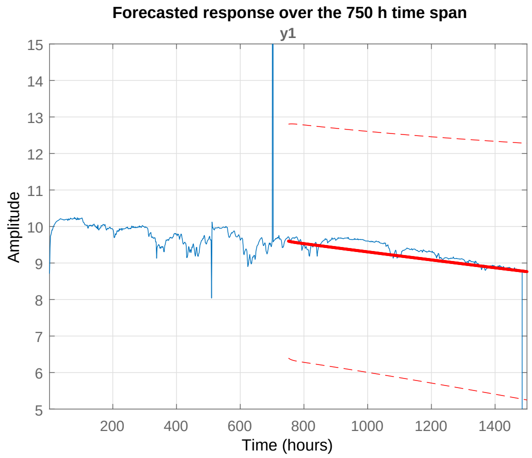


Fig. 8. Model identification result

C. Discussion

1) In order to obtain a precise model, the data used for model identification should contain sufficient information on the concerned system dynamics. This requires the data segment should be large enough.
2) The DFRCM is used to model the concerned fuel cell systems. It should be noted that this specific model

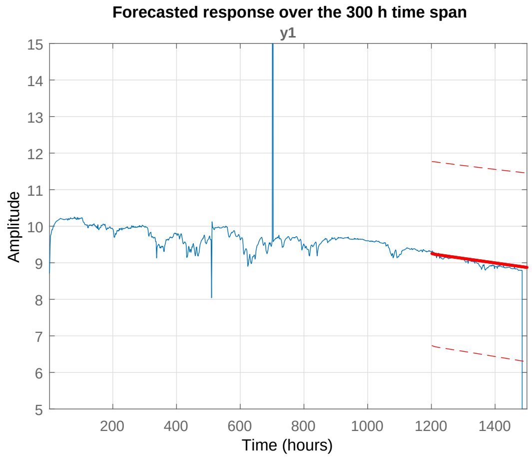


Fig. 9.

can possibly be replace by other models to obtain a comparable performance.

V. CONCLUSION

In this paper, model space based framework is proposed to design PHM for FC systems. DFRCM, a black-box model, is used as the modeling tool to demonstrate the proposed framework. By exploring the evaluation of the model parameters, PHM in dynamic operating conditions can be realized without knowing the first principle system model in prior. DFRCM is firstly used for modeling the NEXA fuel cell system to validate its modeling capability. Then, this modeling tool is used to solve the prognosis problem for another fuel cell system. The results show that DFRCM is suitable for modeling the dynamics of fuel cell systems. Meanwhile, the model space generated by DFRCM identification can be used for PHM and acquire satisfying performance.

REFERENCES

[1] M. Thurston, "An open standard for web-based condition-based maintenance systems," in *AUTOTESTCON Proceedings, 2001. IEEE Systems Readiness Technology Conference*, 2001, pp. 401–415.

[2] J. T. Pukrushpan, A. G. Stefanopoulou, and H. Peng, "Modeling and control for pem fuel cell stack system," in *American Control Conference, 2002. Proceedings of the 2002*, vol. 4. IEEE, 2002, pp. 3117–3122.

[3] Z. Zheng, M.-C. Pra, D. Hissel, M. Becherif, K.-S. Agbli, and Y. Li, "A double-fuzzy diagnostic methodology dedicated to online fault diagnosis of proton exchange membrane fuel cell stacks," *Journal of Power Sources*, vol. 271, no. 0, pp. 570 – 581, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378775314012117

[4] D. Benouioua, D. Candusso, F. Harel, and L. Oukhellou, "Fuel cell diagnosis method based on multifractal analysis of stack voltage signal," *International Journal of Hydrogen Energy*, vol. 39, no. 5, pp. 2236 – 2245, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0360319913027912

[5] S. Morando, S. Jemei, R. Gouriveau, N. Zerhouni, and D. Hissel, "Fuel cells prognostics using echo state network," in *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*. IEEE, 2013, pp. 1632–1637.

[6] H. Chen, P. Tino, A. Rodan, and X. Yao, "Learning in the model space for cognitive fault diagnosis," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 25, no. 1, pp. 124–136, Jan 2014.

[7] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Computer Science Review*, vol. 3, no. 3, pp. 127–149, aug 2009. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1574013709000173

[8] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer, "Information processing using a single dynamical node as complex system." *Nature communications*, vol. 2, p. 468, 2011. [Online]. Available: http://dx.doi.org/10.1038/ncomms1476

[9] L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutiérrez, L. Pesquera, C. R. Mirasso, and I. Fischer, "Photonic information processing beyond turing: an optoelectronic implementation of reservoir computing," *Optics express*, vol. 20, no. 3, pp. 3241–3249, 2012.

[10] G. Sevjidsuren, E. Uyanga, B. Bumaa, E. Temujin, P. Altantsog, and D. Sangaa, "Exergy analysis of 1.2 kw nexatm fuel cell module," in *Clean Energy for Better Environment*. InTech, 2012.

[11] Mathworks, "Time series prediction and forecasting for prognosis," https://fr.mathworks.com/help/ident/ug/time-series-prediction-and-forecasting-for-prognosis.html.

A NOVEL PORT-HAMILTONIAN BASED DESIGN OF STABILIZING CONTROLLER FOR DC-DC BUCK CONVERTERS

Juan Tomassini[(a)], Alejandro Donaire[(b)], Sergio Junco[(c)]

[(a) ,(c)] LAC, Laboratorio de Automatización y Control, Departamento de Control, Escuela de Ingeniería Electrónica, Facultad de Ciencias Exactas e Ingeniería, Universidad Nacional de Rosario, Argentina.
[(b)] Institute for Future Environments, School of Electrical Engineering and Computer Science Science, Queensland University of Technology, Australia.

[(a)]tomajuan@fceia.unr.edu.ar, [(b)]alejandro.donaire@qut.edu.au, [(c)]sjunco@fceia.unr.edu.ar

ABSTRACT

In this work we present a port-Hamiltonian supported control system design aimed at stabilizing a DC-DC Buck converter driving a nonlinear dissipative load. A desired closed-loop dynamics in the form of a port-Hamiltonian system is proposed, whose parameterization enforces the asymptotic stability of the desired equilibrium point. Moreover, the closed loop incorporates a first order dynamic extension allowing to reject constant disturbances on the load side. We prove that the closed loop is ISS respect to unmatched disturbances. To extend this property to disturbances acting on the supply side we add a standard PI output regulator to the previous closed loop. The performance of the closed loop is verified via simulation.

Keywords: DC-DC Buck converter, averaged models, passivity-based control, port-Hamiltonian systems.

1. INTRODUCTION

Due to their versatility, high efficiency, controllable behaviour, fast dynamics and wide-range of power management, Power Electronic Converters (PEC) are ubiquitous and pervade most of the cutting-edge engineering application areas. Indeed, they can be found in electrical drives, switched-mode power supplies, battery chargers, uninterrupted power supplies, all type of mobile devices, distributed generation and renewable energy conversion systems, embedded in electric/hybrid vehicles (cars, trains and airplanes), etc. (F. Dong Tan 2013).

Closed-loop control of PEC is mandatory when their mission is the conditioning of the processed or the output power subject to hard application specifications and under the effect of significant disturbances. Model-based control system synthesis methods are required for high-performance behaviour. From a Modelling point of view, PEC are hybrid, non linear systems composed of continuous elements like inductors, capacitors, resistors, sources, etc., and switching devices allowing for the control actions, like transistors, diodes, etc. As opposed to hybrids associating continuous-variables (with continuous- or discrete-time) and discrete-event models, the vast majority of techniques employed to perform dynamic analysis and control system synthesis are developed on averaged continuous models of PEC. A further division concerns the direct use of nonlinear averaged models or their linearizations around a desired equilibrium point. Linear controllers are tuned for specific operating points and, unless complemented with adaptation mechanisms –what adds complexity to the controller–, the closed-loop performance degrades when the operating point changes. Nonlinear controllers with a unique parameterization valid for the whole operating range are thus preferable, see for instance (Bacha, Munteanu and Bratcu 2014). Exact feedback-linearization, passivity-based control and Lyapunov-like stabilization count among the continuous-time control techniques derived on nonlinear averaged continuous converter models (Sira-Ramírez and Silva-Ortigoza 2006).

Through its application to the control of a Buck converter, this paper presents a method to address these kinds of problems in the modelling framework known as pH systems (PHS), see (Ortega, van der Schaft, Maschke and Escobar, 2002). The rationale of our technique is to find a control law that renders the closed-loop dynamics as a desired PHS, which incorporates a first order extension of the original dynamics. The associated storage function qualifies as a Lyapunov function, therefore guaranteeing the stability of the closed-loop. The resultant feedback law is a controller robust in face of parameter uncertainty and load-side varying bounded disturbances. As it is not robust respect to supply-side disturbances, a second robustifying PI-output regulator is added, which rejects piece-wise constant disturbances.

The remainder of the paper is organized as follows: Section 2 presents the averaged model of the Buck converter and the control system objectives. Section 3 deals with the design of the control system and the derivation of the control law. The behaviour of the overall control law is demonstrated in Section 4 with the help of simulation results in different scenarios, including state-dependent and external disturbances. Finally, Section 5 presents the conclusions.

2. PROBLEM FORMULATION

This section first introduces the topology of the switched converter as well as its averaged nonlinear model and then specifies the control problem to be solved.

2.1. System Model

The idealized equivalent switched circuit of the DC-DC Buck converter is first introduced followed by the average state-equation model employed for the controller design. Only the continuous-conduction mode (CCM) of the inductor is considered.

Figure 1 shows the Buck converter fed by a (possibly non-constant) dc-voltage power supply (on the left) and connected to a load (on the right). With abuse of notation all the variables and functions given in this topological representation will be used in the state-equation model even though their time evolutions in both representations would differ, as the circuit topology contains an idealized switch and the state-equations assume a smooth variation of the supply voltage on the terminals of the converter.

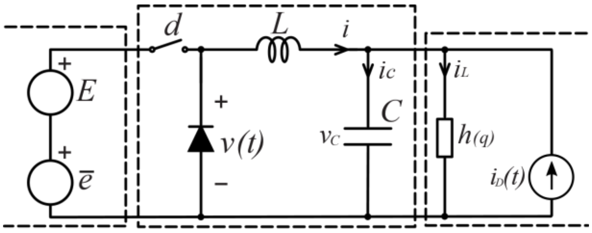


Figure 1: Buck converter with disturbances

The load side is modelled as the parallel connection of a generic dissipative nonlinear load and a current source. The volt-ampère law $i_L = g(v_C) = g\left(\frac{q}{C}\right) = h(q)$ (with q the capacitor charge) of the static dipole is assumed known, and the technical assumption of $h(q)$ a monotone non-decreasing nonlinear function is made. The current source models an unknown independent disturbance $i_D(t) = \bar{i_D} + \tilde{i_D}(t)$. Here, as well as everywhere else in the paper, a bar and a tilde over a variable indicate, respectively, a constant component and a bounded variation of its value.

Taking the flux linkage in the inductance and the capacitor charge as state variables, the following average state-equation model can be derived, where v(t) is the average voltage across the diode, calculated as $\mathrm{v(t) = d(t) \cdot\ [E + \bar{e}]}$, with d(t) the duty-cycle of the binary signal commanding the ideal switch.

$$\begin{aligned} \dot{\psi} &= -\frac{q}{C} + v(t) \\ \dot{q} &= \frac{\psi}{L} - h(q) + i_D(t) \end{aligned} \tag{1}$$

Remark 1: In the sequel the first step in designing the controller will be considering the signal $\mathrm{v(t) = d(t)\big(E + e(t)\big)}$ as its output. The true control variable, i.e., the duty-cycle signal $d(t)$ will be designed in a second step. Notice that this latter signal is constrained to the interval $(0, 1)$. As only the CCM is considered and $d(t) \in (0,1)$, the following holds:

$$\begin{aligned} & i_L = \frac{\psi}{L} > 0 \\ & E > v_C = \frac{q}{C} \geq 0 \end{aligned} \tag{2}$$

2.2. Specifications of closed-loop behaviour

The following problem of output regulation with disturbance rejection has to be solved:

1. Global asymptotic stabilization of the desired equilibrium point (EP) under the solely presence of the constant disturbances $\bar{e}, \bar{i_D}$, where the EP is characterized as follows:

$\overline{v_C} = v_e$, specified constant $\Rightarrow \bar{q} = q_e = C \cdot v_e$
$\bar{\psi} = L \cdot \bar{\imath} = L \cdot [h(q_e) - \bar{i_D}]$

2. Ultimately bounded stability of $(\bar{\psi}, \bar{q})$ under the additional presence of the bounded variable disturbance signal $\tilde{i_D}(t)$, i.e. ISS with respect to this disturbance.

3. CONTROLLER DESIGN

3.1. Desired closed loop dynamics

The problem is formulated in the state-space proposing a first order dynamic extension (state x_3 in the closed loop model) and a feedback control law $v(\psi, q, x_3)$ such that the desired equilibrium point is Globally Asymptotically Stable (GAS) (up to the restrictions (2)). *Disregarding the varying* part of the load-side disturbance, i. e., considering only the constant part of it, this desired closed-loop dynamics (CLD) is a-priori proposed as the following PHS:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \end{bmatrix} = \underbrace{\begin{bmatrix} S_{11} & S_{12} & S_{13} \\ -S_{12} & S_{22}(q) & S_{23} \\ -S_{13} & -S_{23} & S_{33} \end{bmatrix}}_{\boldsymbol{S}(x_2)} \cdot \begin{bmatrix} \frac{\partial H(x)}{\partial x_1} \\ \frac{\partial H(x)}{\partial x_2} \\ \frac{\partial H(x)}{\partial x_3} \end{bmatrix} \tag{3}$$

with $S_{ii} < 0,\ i = \{1,2,3\}\ ;\ \forall\ x_2$

$$H(x_1, x_2, x_3) = \frac{1}{2} \cdot \left(\frac{x_1^2}{L_1} + \frac{x_2^2}{C_2} + \frac{(x_3 - \alpha)^2}{K_I} \right) \tag{4}$$

with the constants $L_1, C_2, K_I > 0$

And the satisfaction of the following conditions:

$$\alpha = -\frac{K_I S_{13} i_D}{S_{13} S_{23} - S_{33} S_{12}} \tag{5}$$

$$S_{23} = -\frac{S_{12}}{S_{13}} \left[\frac{C_2}{L_1 g_2} (S_{13}^2 + S_{11} S_{33}) - S_{33} \right] \tag{6}$$

$$S_{22}(q) = -g_2 - C_2 \frac{h(q) - h(q_e)}{q - q_e}\ ; \tag{7}$$

with the constant $g_2 > 0$

It can be seen that a positive definite storage function $H(x_1, x_2, x_3)$ has been chosen, with its minimum located at $(x_1, x_2, x_3) = (0,0,\alpha)$.

Remark 2: Equations (3), (4) and (5) define a PHS whose equilibrium point $(x_1, x_2, x_3) = (0,0,\alpha)$ is asymptotically stable. Indeed, referring to the standard notation $\dot{x} = (\boldsymbol{J} - \boldsymbol{R})\frac{\partial H(x)}{\partial x}$, the decomposition $\boldsymbol{S}(q) = \boldsymbol{J} - \boldsymbol{R}$ explicitly shows the antisymmetric interconnection matrix $\boldsymbol{J}$ and the positive definite matrix $\boldsymbol{R}$, defined as follows:

$$\boldsymbol{J} = \begin{bmatrix} 0 & S_{12} & S_{13} \\ -S_{12} & 0 & S_{23} \\ -S_{13} & -S_{23} & 0 \end{bmatrix}; \text{ and } \boldsymbol{R} = \operatorname{diag}\{-S_{ii}\}$$

By virtue of the properties of $\boldsymbol{J}$ and $\boldsymbol{R}$, the orbital derivative of $H(x_1, x_2, x_3)$ is a negative definite function of $(x_1, x_2, x_3 - \alpha)$, meaning that the closed loop trajectories converge to $(0,0,\alpha)$ as time goes to infinite. This is shown by the following calculation demonstrating that $H(x_1, x_2, x_3)$ is a Lyapunov function for the equilibrium point:

$$\frac{dH(x)}{dt} = -\left(\frac{\partial H(x)}{\partial x}\right)^T \boldsymbol{R}\,\frac{\partial H(x)}{\partial x} < 0$$

3.2. Controller Design

3.2.1. Definition of the closed loop states

To obtain the control law, firstly we need to define the closed loop states, starting with the output regulation signal:

$$x_2 = q - q_e \tag{8}$$

Since q_e is considered constant, then:

$$\dot{x}_2 = \dot{q} \tag{9}$$

Computing this identity taking $\dot{x}_2$ from (3) and $\dot{q}$ from (1), the following change of variables for x_1 is obtained:

$$x_1 = \frac{L_1}{S_{12}} \cdot \left[S_{22}(q)\frac{q - q_e}{C_2} + S_{23}\frac{x_3 - \alpha}{K_I} - \overline{\iota_D} - \left(\frac{\psi}{L} - h(q)\right)\right] \tag{10}$$

Equations (8) and (10) are now used to replace x_1 and x_2 in the dynamics of x_3 given in (3), that is:

$$\dot{x}_3 = \frac{-S_{13}}{S_{12}} \cdot \left[S_{22}(q)\frac{q - q_e}{C_2} + S_{23}\frac{x_3 - \alpha}{K_I} - \overline{\iota_D} - \left(\frac{\psi}{L} - h(q)\right)\right] - S_{23}\frac{q - q_e}{C_2} + S_{33}\frac{x_3 - \alpha}{K_I} \tag{11}$$

Substituting the value of α from (5) in (11) and using (7), the state equation of x_3 may be written as:

$$\dot{x}_3 = \frac{-S_{13}}{S_{12}} \cdot \left[-g_2\frac{q - q_e}{C_2} - \left(h(q) - h(q_e)\right) + S_{23}\frac{x_3}{K_I} - \left(\frac{\psi}{L} - h(q)\right)\right] - S_{23}\frac{q - q_e}{C_2} + S_{33}\frac{x_3}{K_I} \tag{12}$$

In this way, x_3 can be computed without using the information of the unknown constant disturbance $\overline{\iota_D}$.

Remark 3: The convergence $x_2 \to 0$ amounts to satisfying the regulation requirement for the output (capacitor) voltage $v_c \to \frac{q_e}{C} = v_e$, see Eq. (8).

Remark 4: The dynamic extension has been introduced in order to provide the integral action necessary to asymptotically reject load-side constant disturbances. It is well known that controller integrator outputs tend to constant values which depend on the value of the constant disturbance. In this case this is the convergence $x_3 \to \alpha$, where α depends on the disturbance magnitude, see Eq.(5). Notice that although the change of coordinates and the closed-loop PHS are written using the unknown disturbance, it is only in order to analyze the stability of the control system. Indeed, as it will be seen later, the controller finally implemented *does not* require the information of the disturbance.

Proposition 1: The asymptotic stability of the EP $(\overline{x_1}, \overline{x_2}, \overline{x_3}) = (0,0,\alpha)$ of system (3) implies the asymptotic stability of the EP $(\bar{\psi}, \bar{q})$ of the original system (1) under the action of a constant disturbance $i_D(t) = \overline{\iota_D}$.

Proof: The convergence of the output to its desired equilibrium value has been already established in Remark 3. It remains to show the convergence of the inductance flux to its equilibrium value $\bar{\psi}$. Recalling the equilibrium values of the x variables $(\overline{x_1}, \overline{x_2}, \overline{x_3}) = (0,0,\alpha)$ and using Eq. (10):

$$\overline{x_1} = 0 \Leftrightarrow \frac{L_1}{S_{12}} \cdot \left[S_{22}(x_2)\frac{\bar{q} - q_e}{C_2} + S_{23}\frac{\overline{x_3} - \alpha}{K_I} - \overline{\iota_D} - \left(\frac{\bar{\psi}}{L} - h(\bar{q})\right)\right] = 0 \Leftrightarrow \bar{\psi} = L\left(h(\bar{q}) - \overline{\iota_D}\right) \tag{13}$$

3.2.2. Computing the control law

The control law is calculated matching the expressions for $\dot{x}_1$ given in (3) and the one obtained via time differentiation of (10). The following auxiliary calculations lead to the desired matching equation (ME). First, from (3) we obtain the left side of it as:

$$\dot{x}_1 = S_{11}\frac{x_1}{L_1} + S_{12}\frac{x_2}{C_2} + S_{13}\frac{x_3 - \alpha}{K_I}$$

$$= \frac{S_{11}}{S_{12}}\left[S_{22}(q)\frac{q - q_e}{C_2} + S_{23}\frac{x_3 - \alpha}{K_I} - \overline{\iota_D} - \left(\frac{\psi}{L} - h(q)\right)\right] + S_{12}\frac{q - q_e}{C_2} + S_{23}\frac{x_3 - \alpha}{K_I}$$

Then, differentiating (10), and using (1) to replace $\dot{q}$ and $\dot{\psi}$ we obtain the right side for the ME:

$$\dot{x_1} = \frac{L_1}{S_{12}}\left(\left[\frac{\partial S_{22}(q)}{\partial q} \cdot \frac{q-q_e}{C_2} + \frac{S_{22}(q)}{C_2}\right] \dot{q} + S_{23}\frac{\dot{x_3}}{K_I} - \frac{\dot{\psi}}{L} + \frac{\partial h(q)}{\partial q}\dot{q}\right)$$

$$= \frac{L_1}{S_{12}}\left(\left[\frac{\partial S_{22}(q)}{\partial q} \cdot \frac{q-q_e}{C_2} + \frac{S_{22}(q)}{C_2} + \frac{\partial h(q)}{\partial q}\right]\left(\frac{\psi}{L} - h(q) + \overline{\iota_D}\right) + S_{23}\frac{\dot{x_3}}{K_I} - \frac{1}{L}\left(v(t) - \frac{q}{C}\right)\right)$$

Now, the control input $v(t)$ is explicitly shown. If (7) holds, then:

$$\frac{\partial S_{22}(q)}{\partial q} \cdot \frac{q-q_e}{C_2} + \frac{S_{22}(q)}{C_2} + \frac{\partial h(q)}{\partial q} = -\frac{g_2}{C_2}$$

Now the ME can be written as follows:

$$\frac{S_{11}}{S_{12}}\left[S_{22}(q)\frac{q-q_e}{C_2} + S_{23}\frac{x_3-\alpha}{K_I} - \overline{\iota_D} - \left(\frac{\psi}{L} - h(q)\right)\right] + S_{12}\frac{q-q_e}{C_2} + S_{23}\frac{x_3-\alpha}{K_I} = \frac{L_1}{S_{12}}\left(\frac{-g_2}{C_2}\left(\frac{\psi}{L} - h(q) + \overline{\iota_D}\right) + S_{23}\frac{\dot{x_3}}{K_I} - \frac{1}{L}\left(v(t) - \frac{q}{C}\right)\right) \quad (14)$$

Recall x_3 is independent of α and $\overline{\iota_D}$. So, if (5) and (6) hold, the ME (14) together with Eq. (12) leads us to the following control law:

$$v(\psi, q, x_3) = \left[\left(\frac{\psi}{L} - h(q)\right)\left(\frac{L_1 S_{23} S_{13}}{S_{12}^2 K_I} - \frac{L_1 g_2}{S_{12} C_2} + \frac{S_{11}}{S_{12}}\right) + \frac{x_3}{K_I}\left(\frac{L_1 S_{23}}{S_{12} K_I}\left(S_{33} - \frac{S_{13}}{S_{12}}S_{23}\right) - \left(S_{13} + \frac{S_{11}}{S12}S_{23}\right)\right) + \left(\frac{q-q_e}{C_2}g_2 + h(q) - h(q_e)\right) \cdot \left(\frac{L_1 S_{23} S_{13}}{S_{12}^2 K_I} + \frac{S_{11}}{S_{12}}\right) - \frac{q-q_e}{C_2}\left(S_{12} + \frac{L_1 S_{23}^2}{K_I S_{12}}\right) + \frac{L_1}{LS_{12}}\frac{q}{C}\right]\frac{LS_{12}}{L_1} \quad (15)$$

Equation (15), together with the dynamics of x_3 given by (12), provides the feedback control law that produces the desired CLD of the open loop system (1).

The fulfillment of (5), (6) and (7) together with $S_{11}, S_{33} < 0$ ensures that the control law and the controller state dynamics are independent of $\overline{\iota_D}$ and α, and $S_{22}(q) < 0$, $\forall\, q$.

Remark 5: Notice that, despite its seeming complexity, the control law (15) is a simple linear expression up-to the need to reconstruct the load current in the controller through the nonlinear function $h(q)$. The same holds for the dynamics of x_3. In the implementation of the control law, instead of ψ and q, the inductor current i and the capacitor voltage v_c are to be used.

Remark 6: Recalling that the real control input is not $v(\psi, q, x_3)$ but the duty cycle signal $d(t)$ calculated as follows:

$$d(t) = \frac{1}{E} \cdot\ v(\psi, q, x_3) \quad (16)$$

it is recognized that for this controller to assure asymptotic stability the condition $\bar{e} = 0$ must be satisfied, i.e., the supply-side disturbance must be zero. This shortcoming will be removed supplementing this controller with a PI-regulator acting on the output error (see subsection 3.4).

3.3. Rejection of load-side time-varying disturbances

When considering the time varying disturbance $\widetilde{\iota_D}(t)$ acting on the load side, i.e., the whole disturbance $i_D(t) = \overline{\iota_D} + \widetilde{\iota_D}(t)$, the PHS (3) no longer the closed-loop dynamics, as it is driven by the disturbance as specified in (17):

$$\dot{x} = \boldsymbol{S}(x_2) \cdot \frac{\partial H(x)}{\partial x} + \begin{bmatrix} d_{x1} \\ d_{x2} \\ 0 \end{bmatrix} \quad (17)$$

where the driving inputs $d_{x1,2}$ depend on the disturbance $\widetilde{\iota_D}(t)$ acting on the open loop system (1) as follows:

$$\begin{bmatrix} d_{x1} \\ d_{x2} \\ 0 \end{bmatrix} = \begin{bmatrix} -\frac{L_1 g_2}{S_{12} C_2} \cdot \widetilde{\iota_D}(t) \\ \widetilde{\iota_D}(t) \\ 0 \end{bmatrix} \quad (18)$$

Proposition 2: System (1) in closed loop with the controller given by Eq. (15) and Eq. (12) is Input-to-State-Stable (ISS) (Khalil, 2002) with respect to the bounded disturbance $\widetilde{\iota_D}(t)$.

Proof: First we compute the time derivative of $H(x)$, considering the disturbances $d_{x1,2}$, i.e., the dynamics (17):

$$\dot{H}(x) = \frac{S_{11}x_1^2}{L_1^2} + \frac{S_{22}(x_2)x_2^2}{C_2^2} + \frac{S_{33}(x_3-\alpha)^2}{K_I^2} + \frac{x_1}{L_1}d_{x1} + \frac{x_2}{C_2}d_{x2} \quad (19)$$

Recalling that $S_{11} < 0$, $S_{33} < 0$ and $S_{22}(x_2) \leq -g_2$, replacing $d_{x1,2}$, and using the following auxiliary fact (written for some generic variables γ, ω)

$$-a\gamma^2 + b\gamma\omega \leq -\frac{a}{2}\gamma^2 + \frac{2\,b^2}{a}\omega^2\ ; \quad \text{with } a, b > 0$$

the following inequalities can be obtained:

$$\frac{S_{11}x_1^2}{L_1^2} + \frac{x_1}{L_1}d_{x1} \leq \frac{-|S_{11}|}{2L_1^2}x_1^2 + \frac{2}{|S_{11}|}\left(\frac{-L_1 g_2}{S_{12}C_2}\right)^2 \widetilde{\iota_D}(t)^2$$

$$\frac{S_{22}(x_2)x_2^2}{C_2^2} + \frac{x_2}{C_2}d_{x2} \leq -\frac{g_2}{2C_2^2}x_2^2 + \frac{2}{g_2}\widetilde{\iota_D}(t)^2$$

Thus, the following inequality for $\dot{H}(x)$ can be written:

$$\dot{H}(x) \leq \frac{-|S_{11}|}{2L_1^2} x_1^2 + \frac{2}{|S_{11}|}\left(\frac{-L_1 g_2}{S_{12}C_2}\right)^2 \widetilde{i_D}(t)^2 - \frac{g_2}{2C_2^2} x_2^2 + \frac{2}{g_2}\widetilde{i_D}(t)^2 + S_{33}\frac{(x_3-\alpha)^2}{K_I^2} \tag{20}$$

Defining:

$$\lambda_1 = \text{mín}\left\{\frac{|S_{11}|}{2L_1^2} ; \frac{g_2}{2C_2^2} ; \frac{|S_{33}|}{K_I^2}\right\},\ \lambda_1 > 0$$
$$\lambda_2 = \frac{2}{|S_{11}|}\left(\frac{L_1 g_2}{S_{12}C_2}\right)^2 + \frac{2}{g_2},\ \lambda_2 > 0 \tag{21}$$

$$\chi = \begin{bmatrix} x_1 \\ x_2 \\ x_3 - \alpha \end{bmatrix}$$

$$\dot{H}(\chi) \leq -\lambda_1 |\chi|^2 + \lambda_2 \widetilde{i_D}(t)^2 \tag{22}$$

Eq. (22) completes the proof.

3.4. Rejection of supply-side disturbances

As seen in Figure 1, the supply voltage assumes a constant known value E (rated voltage of the source) plus an unknown (possibly piece-wise) constant disturbance value $\bar{e}$.

There are many applications where the mean value $\bar{e}$ is very small or directly zero, but there are others where it is of paramount importance, for instance, the case of solar PV arrays providing energy to a load through the converter system. This shows the importance of having a controller able to reject both disturbance inputs, $i_D(t) = \bar{i_D} + \widetilde{i_D}(t)$ and also $\bar{e}$, thus assuring the ISS stability of the EP. This property can be achieved enhancing the control law with an additional PI action processing the output error. This yields the following expression for the duty-cycle (recall that $\frac{x_2}{C} = \frac{q-q_e}{C} = v_c - \bar{v}$):

$$d(t) = \frac{1}{E}\cdot\left(v(\psi, q, x_3) + K_{pi}\frac{x_2}{C} + K_{ii}\int\frac{x_2}{C}dt\right) \tag{23}$$

where E is the constant voltage of the source, which is not measured, but programmed as its rated value for the calculation of $d(t)$.

Remark 7: To maintain the properties of the control law (15) without the addition of the PI, the duty cycle had to be calculated not as shown in (16) but as $d(t) = \frac{1}{E+\bar{e}}\cdot v(\psi, q, x_3)$. This is not convenient (or possible, under certain circumstances) because it would imply equipping the system with one more sensor to measure the supply voltage.

4. VALIDATION THROUGH SIMULATION

The performance of the controller (15), (12) is tested via simulation. First, the controller is tested under design conditions and next in presence of constant and bounded time-varying disturbances and parameter dispersion in some key electrical components (load model and the value of the voltage source). The model of the load, which is graphically given in Figure 2, is:

$$h(q) = \left(\frac{v_C}{20}\right)^5 - \left(\frac{v_C}{20}\right)^3 + \left(\frac{v_C}{87}\right) + \text{atan}\left(\frac{v_C}{1.5}\right) \tag{24}$$

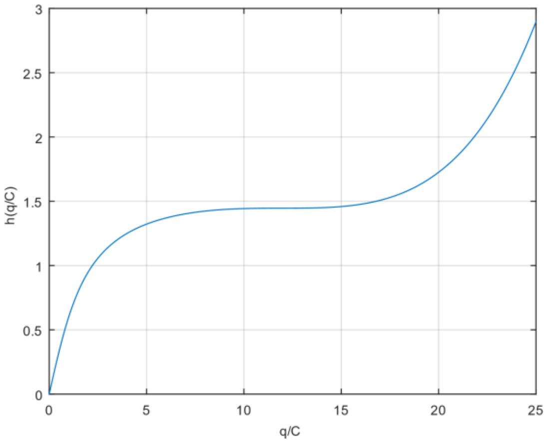


Figure 2: Nonlinear dissipative volt-ampère law

Thus, $h(q)$ is a nonlinear dissipative load (NLD) fulfilling the non-decreasing assumption. The parameters used in simulation of the Buck converter are taken from (Kwasinski and Krein 2007): $L = 500\,uH$, $C = 1000\,uF$ and $E = 22{,}2\,V$. The set of parameters for the controller are $S_{11} = -0{,}5$, $S_{12} = -1$, $S_{13} = -1$, $S_{23} = -1{,}195$, $g_2 = 3$, $S_{33} = -0{,}7$, $L_1 = 2L$, $C_2 = 1{,}1C$ and $K_I = 0{,}04$.

4.1. Controller analysis under design conditions

In this subsection the controller is tested under the design conditions. First, considering perfect knowledge of the model (1); next, introducing a constant disturbance $\bar{i_D}$; and, at last, a time-varying disturbance $\widetilde{i_D}(t)$ to show the ISS property.

Experiment 1: The system starts with zero initial conditions. The capacitor voltage reference is set to $\bar{v} = 12V$. At $t = 70ms$ the voltage reference changes to $\bar{v} = 17V$. At time $t = 140ms$ a $50mA$ constant current load is connected in parallel with the NLD. The time response is shown in Figure 3.

Experiment 2: Same simulation scenario as Experiment 1. At $t = 25ms$ a bounded disturbance is injected: $\widetilde{i_D}(t) = 0.5\frac{\bar{\psi}}{L}sin(2\pi 50t)$ is injected. In presence of $\widetilde{i_D}(t)$, the capacitor voltage reference is changed and the constant current load is connected. The time response is shown on Figure 4.

The controller asymptotically stabilizes the desired equilibrium point, even under the presence of a *constant* load-side disturbance. This feature is provided by the dynamic extension, conceived to reject that type of disturbances. The ISS property, i.e., the bounded response under the action of a bounded disturbance, can be observed in the last set of simulations.

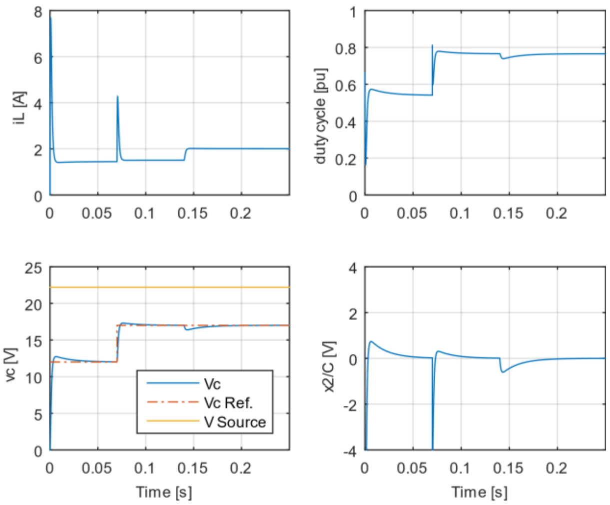


Figure 3: Time responses of Experiment 1.

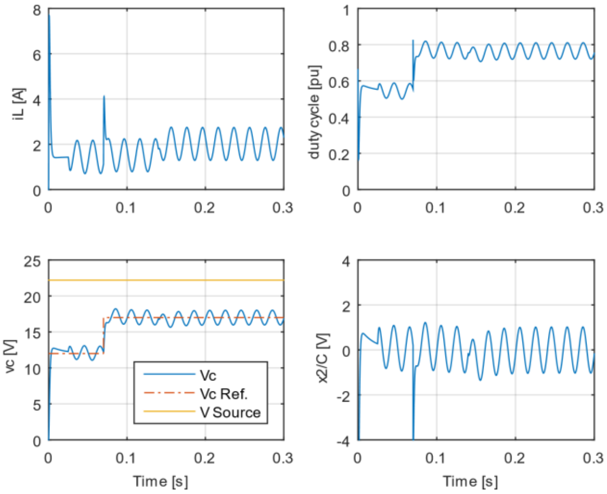


Figure 4: Time responses of Experiment 2.

4.2. Disturbances on supply-side

First, the non-robustness of the control law configured by {(15), (12),(16)} respect to supply-side disturbances is shown. Later, the rejection by the outer PI-loop of piece-wise constant disturbances acting on this side is demonstrated.

4.2.1. Performance degradation of controller {(16), (12),(17)}

This controller, designed to reject load side disturbances, is tested now under the presence of a supply-side disturbance.

Experiment 3: For this experiment $i_D(t) = 0$. The voltage source value is $19{,}98\,V$, (90% of $22{,}2\,V$). Recall that the controller uses $E = 22{,}2\,V$ to calculate the duty cycle.

It is obvious from Figure 5 that the presence of a disturbance $e(t) = \bar{e}$ changes the EP of the closed loop to a different one, with $\overline{x_1} \neq 0$ and $\overline{x_2} \neq 0$. This is because the duty cycle is miscalculated under the effect of the unknown value $\bar{e}$.

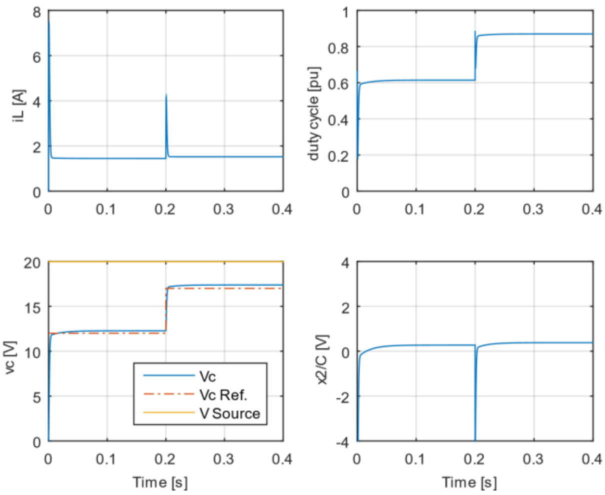


Figure 5: Time responses of Experiment 3

4.2.2. Supply-side disturbance rejection

An additional integral action is performed in order to reject $\bar{e}$, see Eq. (23).

Experiment 4: The simulation scenario is the same as Experiment 3*:* , but now the duty cycle is calculated using Eq. (23). The parameters of the additional PI controller are: $K_{pi} = 0{,}75$ and $K_{ii} = 1/0{,}03$.

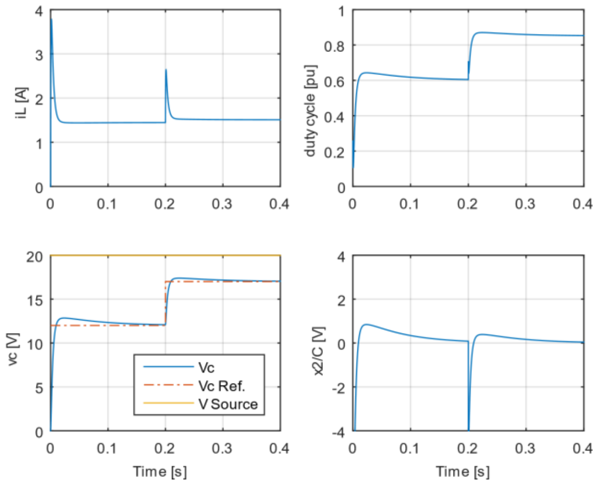


Figure 6: Times responses of Experiment 4.

Using Eq. (23), $\bar{e}$ is rejected, but the response of the whole system becomes slower. This is because the additional PI is tuned to be slower than the closed loop system (3).

4.3. Disturbances on load- and supply-sides

The objective of the experiment below is to show that the controller (15), (12) enhanced with an additional PI controller (Eq. (23)) can reject constant disturbances on both sides and preserves the ISS property with respect to load side time varying bounded disturbances.

We also introduce now a state dependent disturbance: changing the model of the load connected to the output of the Buck converter by the following one, see also Figure 7:

$$h(q) = \left(\frac{v_C}{20}\right)^5 - \left(\frac{v_C}{17}\right)^3 + \left(\frac{v_C}{87}\right) + \text{atan}\left(\frac{v_C}{1.5}\right) \qquad (25)$$

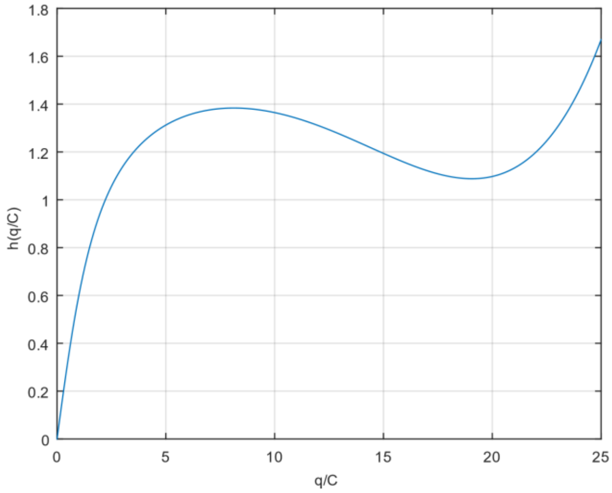


Figure 7: Volt-ampère law of nonlinear load for Experiment 5.

The controller remains programmed with the load model given by Eq. (24). Notice that this load does not fulfill the non-decreasing assumption, which can be relaxed as long as $S_{22}(q) < 0, \forall\, q$. This implies that g_2 must be bigger enough to ensure it.

Experiment 5: The voltage reference, initially set to $12\,V$, changes to $16\,V$ at $t = 0{,}15s$ (notice that both reference values are in the decreasing volt-ampère zone of the load model). The supply-side disturbance is the same as in the previous experiment. At time $t = 0{,}3s$ a $50\,mA$ constant current load is connected in parallel with the load, and at time $t = 0{,}5s$ a bounded load side disturbance $\widetilde{\iota_D}(t)$ is injected (same as Experiment 2).

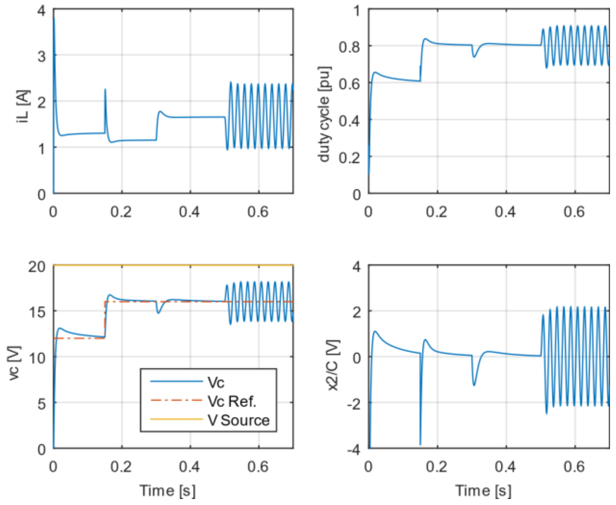


Figure 8: Time responses of Experiment 5.

The desired equilibrium point is stabilized, under the presence of disturbances acting on both sides, and the ISS property is conserved.

5. CONCLUSIONS

A dynamic controller assuring global asymptotic stability of the desired equilibrium point of a DC-DC Buck converter has been designed. On the basis of a nonlinear averaged state-equation model, the control system design was achieved proposing a closed-loop PHS target model having a positive definite energy function and dissipation in all its states, making it a Lyapunov function for any desired equilibrium point. It has been shown that this design guarantees ISS stability regarding (possible varying) bounded disturbances on the load-side. An outer PI-loop was added in order to reject also piece-wise constant disturbances on the supply-side. Simulation experiments confirm the correct performance of the overall controller.

Further work includes increasing by one the order of the dynamic extension embodied in the closed-loop PHS in order to be able to reject supply-side disturbances without adding a classical PI outer loop. Also of interest is extending the results to other kind, possibly dynamic and/or non passive loads, as well as applying the methodology to other topologies of power electronic DC-DC converters. In view of the controller practical implementation, testing and tuning the closed-loop performance on a hybrid model where the control input is provided by a switch driven by a PWM-modulated continuous duty cycle is also planned, as well as performing experimental validation tests.

ACKNOWLEDGMENTS

The authors wish to thanks SeCyT-UNR (the Secretary for Science and Technology of the National University of Rosario) and ANPCyT for their financial support through projects PID-UNR 1ING502 and FONCyT PICT 2012 Nr. 2471, respectively.

REFERENCES

Bacha S., Munteanu I., Bratcu A., 2014. Power Electronic Converters Modelling and Control: with Case Studies. London: Springer

Dong Tan F., 2013. Address from the Editor-In-Chief of the IEEE J. Emerging & Selected Topics in Power Electronics. Available from: http://www.ieee-pels.org/publications/jstpe [accessed 15 May 2016].

Khalil, H., 2002. Nonlinear Systems Second Edition. New Jersey: Prentice-Hall.

Kwasinski, A., Krein, P. T., 2007. Passivity-based control of Buck converters with constant-power loads. Power Electronics Specialists Conference, PESC 2007. IEEE, pp. 259-265, 17-21 June 2007, Orlando, FL.

Ortega R., van der Schaft A., Maschke B., Escobar G., 2002. Interconnection and damping assignment passivity-based control of port-controlled Hamiltonian systems. Automatica, Volume 38, Issue 4: pp. 585–596.

Sira-Ramírez H., Silva-Ortigoza R., 2006. Control Design Techniques in Power Electronics Devices. London: Springer-Verlag

Tan S. C., Lai Y. M, Tse, C. K., Cheung, M. K. H., 2006. Adaptive feedforward and feedback control schemes for sliding mode controlled power converters. IEEE Trans. Power Electron., vol. 21: no. 1, pp. 182–192.

AUTHORS BIOGRAPHY

Sergio Junco received the Electrical Engineer degree from the *Universidad Nacional de Rosario* (UNR) in 1976. In 1982, after 3 years in the steel industry and a 2-year academic stage at the University of Hannover, Germany, he joined the academic staff of UNR, where he currently is a Full-time Professor of System Dynamics and Control and Head of the Automation and Control Systems Laboratory. His current research interests are in Modelling, simulation, control and diagnosis of dynamic systems, with applications in the fields of motion control systems with electrical drives, power electronics, mechatronics, vehicle dynamics and smart grids. He has developed, and currently teaches, several courses at both undergraduate and graduate level on System Dynamics, Bond Graph Modelling and Simulation, Advanced Nonlinear Dynamics and Control of Electrical Drives, as well as Linear and Nonlinear Control with Geometric Tools.

Juan Tomassini was born in Rosario, Argentina. He received his degree in Electrical Engineering from the Universidad Nacional de Rosario (UNR), Argentina, in 2013. He worked as an electrical generation programmer in the administrator company of the wholesale electricity market (CAMMESA). Since September 2014 he has been a PhD student in Electrical Engineering and Control at the Faculty of Engineering (FCEIA) of UNR. His work is supported by the Nacional Agency for Scientific and Technological Promotion (ANPCyT). His main research interests are on IDA-PBC control, renewable energy and smart grids.

Alejandro G. Donaire was born in Rosario, Argentina. He received the Electronic Engineering and PhD degrees in 2003 and 2009, respectively, both from the Universidad Nacional de Rosario (UNR), Argentina. His work was supported by the Argentine National Council of Scientific and Technical Research, CONICET. In 2009, he joined the Centre for Complex Dynamic Systems and Control, The University of Newcastle, Australia. From 2015 to March 2017 he was with the PRISMA Lab at the Università degli Studi di Napoli Federico II. He recently joined the Institute for Future Environments, School of Electrical Engineering and Computer Science, Queensland University of Technology, Australia. His research interests include nonlinear and energy-based control theory with application to electrical drives, multi-agent systems, robotics, smart micro-grids networks, marine and aerospace mechatronics, and power systems.

OPTIMAL SIZING OF A HYBRID COOLING SYSTEM

Farah Kojok[a], Rachid Outbib[b], Oussama Ibrahim[c], Rafic Younes[d], Farouk Fardoun[e]

[a],[b] LSIS- Aix-Marseille University - Domaine Universitaire de Saint-Jérôme, Av. Escadrille Normandie-Niemen, 13397 Marseille Cedex 20, France
[c], [d] Department of Mechanical Engineering, Lebanese University, Beirut, Lebanon
[d] LISV Laboratory, Versailles Saint Quentin University, Versailles, France
[e] University Institute of Technology, Department GIM, Lebanese University Campus, P.O. Box 813, Saida, Lebanon

[a] farahkojok2579@hotmail.com, [b] rachid.outbib@lsis.org , [c] oibrahimul@hotmail.com ,
[d] ryounes@ul.edu.lb, [e] fardoun2@yahoo.com

Abstract

The aim of this work is to propose an optimal sizing method for a hybrid cooling system with minimum energy consumption and maximum share of renewable resources. For this purpose, a hybrid cooling system that benefits from renewable energy resources is designed and modeled using Trnsys 17. Then, an optimal sizing method where the problematic is formalized as a mathematical problem under constraints is presented. Finally, the results are obtained and discussed for the two case studies selected, namely Marseilles-France and Beirut-Lebanon.

Keywords: hybrid cooling, optimal sizing, energy management, renewable energy, grid electricity.

1. INTRODUCTION

The demand for cooling is dramatically increasing due to the climate change and the global temperature rise. However, cooling systems are a matter of debate, not only due to the high-energy consumption, but also due to the excessive CO2 emissions as well as the use of synthetic refrigerants that produce greenhouse effect and cause global warming. Therefore, energy efficient cooling systems are key contributors to the reduction of the electrical energy consumption. Solar cooling is one of the energy saving cooling technologies used all over the world. Moreover, and due to the variability of these local energy sources, hybridization in sources and storage systems is to be considered to get new solutions for energy production and hence meet the needs (Poulet and Outbib 2015). Actually, hybrid cooling technology could lead to a great energy saving and COP improvement by more than 50% (Kairouani and Nehdi 2006). However, the improvement achieved by a hybrid cooling system vary widely. It depends on the system design and the way the different types of energy are used. Further, it is found that in some cases, individual cooling systems are more effective than hybrid systems. For instance, in very hot and arid climate, the removal of latent load separately by desiccant system becomes unnecessary, and thus, the comfort level could be reached using an electric or absorption chiller standing alone (Kojok, Fardoun, Younes and Outbib 2016)).

In this work, an optimal sizing method that defines, in a specific region, a hybrid cooling energy system, economically feasible with maximum renewable energy share is presented. The method tends to minimize the dependency of the cooling system on grid electricity and to reduce the nonrenewable energy consumption. Based on the above, a hybrid cooling system, used for a small residential house, is designed and presented in the first section. After that, an optimal sizing method is proposed in section two. The main purpose is to determine, for a given region of the world, a cooling system with minimum energy consumption and maximum share of renewable resources. More precisely, once a hybrid technology is adopted, the main purpose is to define an optimal size taking into consideration a certain number of inherent constraints that could be the energy potential, the financial costs of the system designed and the area available in a building to install the said system. For this purpose, a residential hybrid cooling system that benefits from renewable energy resources is designed and modeled using Trnsys 17. The system, schematized in figure 1, mainly consists of a solar absorption chiller (AC) , a vapor compression chiller (VCC), a solid desiccant system (DSC), two evacuated tube solar thermal collectors (ET) , two hot water storage tanks (HW) and a chilled water storage tank (CHW), a wind turbine (WT) , a Photovoltaic cell (PV) and batteries with inverter for electric storage the batteries that satisfy the maximum energy output of both WT and PV.

This designed cooling system is used to ensure the cooling need of a standard small house. The total cooling load is separated into its two components ;the sensible load that is related to the dry bulb temperature and the latent load that is related to the wet bulb temperature or the relative humidity. Each load is treated separately by using separate components of the hybrid system. The desiccant dehumidifier component is used to meet the latent load, while chiller components are utilized to meet the sensible load and excess latent load, if found.This strategy of separating both kinds of loads has been proved to be an efficient energy-saving method since it raises the evaporator temperature in the sensible cooling machine (Ling, Kuwabara, Hwang and Radermacher 2013) and (Ling, Kuwabara, Hwang and Radermacher 2011).

In addition, a solid type desiccant is used because it is preferred to a liquid type one, especially for residential applications, due to the simplicity of its structure and its common regeneration by a thermal solar energy source (Ge,Dai and Wang 2014). Concerning the control method of the system, If the solar absorption chiller is not able to meet the chilled water need, then the electric vapor compression chiller starts operating, where its electricity need is secured from the electrical energy renewable energy systems, PV cells, wind turbines and batteries, if stuffiest. Else, grid electricity is utilized. As for the desiccant system, it starts operation when the relative humidity in the house is greater than the relative humidity set point RH_h^{sp} and the desiccant material is regenerated by solar thermal energy stored in the hot water tank water HW2. After being dehumidified in the desiccant wheel, the process air is cooled by the chilled water through a water-to-air heat exchanger (HX) before passing into the house.

2. OPTIMAL SIZING METHOD

This section presents the optimal sizing method applicable to the hybrid cooling system defined in the previous section to define the best configuration from energetic and economic standpoints. At the end of the optimization problem, the number of renewable energy systems and cooling machines for the best configuration hybrid system is defined. Hence, for this hybrid system, the energetic objective aims to minimize the total grid-electrical energy consumption and consequently maximize the renewable energy share. The economic objective aims to obtain a positive net present value (NPV) for the proposed system, which is also greater than that of conventional electric vapor compression chiller over a pre-specified period.

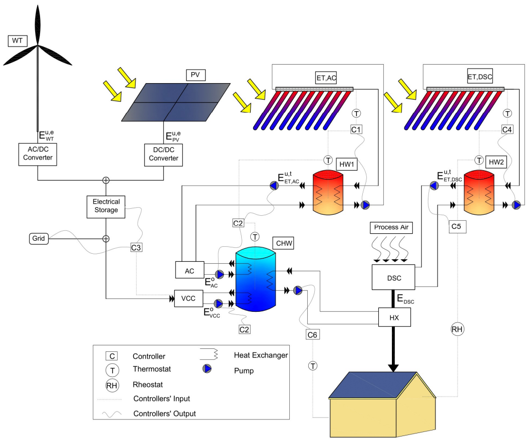


Figure 1: Schematic representation of the proposed residential hybrid cooling system

The optimal sizing method is based on the concept of simulating the proposed hybrid cooling system using small base-units for the renewable energy components, solar and wind systems: PV cell, wind turbine, evacuated tube solar collector connected to hot water storage tank for both absorption chiller and desiccant system. Note that, the evacuated tube solar collector connected to hot water storage tank is considered as one unit. These base units are supposed to produce their maximum possible energy during the studied period of time. Thus, a multiple of a base unit would multiply its maximum energy output for the same climatic and boundary conditions. Results of the sizing problem should then find the appropriate number of individual renewable base units where the number is limited to a maximum value. This method has been first proposed in (Ibrahim, Fardoun, Younes and Louahlia-Gualous 2014). However, in this study, another application is investigated and another formulation and problem solution are presented.

2.1. Objective function

The objective function is determined according to the following demonstration.

The following equations represent the relations between input and output energies (E^i and E^o) of each of the absorption chiller, the vapor compression chiller and the desiccant wheel are presented as follow, as well as the energy produced by renwable energy systems ($E^{u,t}_{ET,AC}$, $E^{u,t}_{ET,DSC}$, $E^{u,e}_{PV}$, $E^{u,e}_{WT}$) and electrical energy from grid E_{grid}.

$$E^{o}_{AC} = COP_{AC}\, E^{i}_{AC} = COP_{AC}\, N^{AC}_{ET}\, E^{u,t}_{ET,AC} \qquad (1)$$

$$E^{o}_{VCC} = COP_{VCC}\, E^{i}_{VCC} = COP_{VCC}\left(N_{PV}E^{u,e}_{PV} + N_{WT}E^{u,e}_{WT} + E_{grid}\right) \qquad (2)$$

$$E^{o}_{DSC} = COP_{DSC}\ E^{i}_{DSC} = COP_{DSC}\ N^{DSC}_{ET}\ E^{u,t}_{ET,DSC} \qquad (3)$$

Where N and COP are the number and the coefficient of performance of each unit.

The hybrid system should meet the total cooling load (E_L) of the house. Hence, this constraint is presented in equation (4).

$$E^{o}_{AC} + E^{o}_{VCC} + E^{o}_{DSC} \geq E_L \qquad (4)$$

Using equations (1) to (4), the following equation is obtained:

$$COP_{VCC}\left(N_{PV}E^{u,e}_{PV} + N_{WT}E^{u,e}_{WT} + E_{grid}\right) + COP_{AC}\ N^{AC}_{ET}\ E^{u,t}_{ET,AC} + COP_{DSC}\ N^{DSC}_{ET}\ E^{u,t}_{ET,DSC} \geq E_L \qquad (5)$$

Hence, equation (5) could be reformulated to give equation (6):

$$E_{grid} \geq \frac{E_L}{COP_{VCC}} - \left(\frac{COP_{AC}}{COP_{VCC}}\ N^{AC}_{ET}\ E^{u,t}_{ET,AC} + \frac{COP_{DSC}}{COP_{VCC}}\ N^{DSC}_{ET}\ E^{u,t}_{ET,DSC} + N_{PV}E^{u,e}_{PV} + N_{WT}E^{u,e}_{WT}\right) \qquad (6)$$

The second term of the right part of equation (6) represents the total renewable energy share of the hybrid system and maximizing this term will minimize the grid electricity consumption as it is obvious from this equation. Therefore, the objective function stated in equation (7).

$$\max f = \frac{COP_{AC}}{COP_{VCC}}\ N^{AC}_{ET}\ E^{u,t}_{ET,AC} + \frac{COP_{DSC}}{COP_{VCC}}\ N^{DSC}_{ET}\ E^{u,t}_{ET,DSC} + N_{PV}E^{u,e}_{PV} + N_{WT}E^{u,e}_{WT} = \langle \widehat{N}, \beta \rangle \qquad (7)$$

With

$$\widehat{N} = \begin{pmatrix} N_{PV} \\ N_{WT} \\ N^{AC}_{ET} \\ N^{DSC}_{ET} \end{pmatrix} \quad \text{and} \quad \beta = \begin{pmatrix} E^{u,e}_{PV} \\ E^{u,e}_{WT} \\ \frac{COP_{AC}}{COP_{VCC}}\ E^{u,t}_{ET,AC} \\ \frac{COP_{DSC}}{COP_{VCC}}\ E^{u,t}_{ET,DSC} \end{pmatrix}$$

2.2. Constraints

2.2.1. Energy balance constraints

The maximum cooling energy delivered by the desiccant wheel should not exceed the total latent cooling load as formulated in the following equation:

$$\langle \widehat{N}\ , \rho \rangle \leq E_l \qquad (8)$$

Where $\rho = \begin{pmatrix} 0 \\ 0 \\ 0 \\ COP_{DSC}\ E^{u,t}_{ET,DSC} \end{pmatrix}$

2.2.2. Area constraints

Furthermore, given that the hybrid cooling system is applied to a residential with limited roof surface area, it is necessary to impose area constraints related to the number of renewable energy systems installed on the roof. Equation (9) represents this constraint as follows:

$$\langle \widehat{N}\ , \gamma \rangle \leq A_T \qquad (9)$$

Where $\gamma = \begin{pmatrix} A_{PV} \\ A_{WT} \\ A^{AC}_{ET} \\ A^{DSC}_{ET} \end{pmatrix}$

where A_X is the surface required for renewable energy system and A_T is the useful total surface area of the house roof.

2.2.3. Economic constraints

From an economic point of view, when comparing two systems, the one with greater Net Present Value (NPV) is the preferred. Thus, in this case, the proposed hybrid cooling system economically compared to a conventional electric vapor compression chiller, this constraint could be translated into the following equation:

$$NPV_{VCC}^{cv} - \mathrm{NPV_{hyb}} \leq 0 \tag{10}$$

Where $\mathrm{NPV_{VCC}^{cv}}$ and $\mathrm{NPV_{hyb}}$ are the net present values of the conventional vapor compression chiller and the hybrid system, respectively.

Since there is no energy savings considered for the conventional system, then its net present value is simply its capital cost during the studied period. Hence, equation (10) could be written as follows:

$$\begin{aligned}\left[\left(\sum_{X\in H_1} \mathrm{C}_{\mathrm{X}}^{\mathrm{hyb}} + \sum_{X\in H_2} \mathrm{C}_{\mathrm{X}}^{\mathrm{hyb}}\right) - \mathrm{C_{VCC}^{cv}}\right] \\ - \left[\alpha(\mathrm{N_y})\mathrm{C^e}\left(\mathrm{E_{grid}^{Cv}}\right.\right. \\ \left.\left. - \left(\frac{\mathrm{E_L}}{\mathrm{COP_{VCC}}} - \langle\widehat{\mathrm{N}}, \beta\rangle\right)\right)\right] \\ \leq 0\end{aligned} \tag{11}$$

Where

- $H_1 \ \epsilon \{PV;\ WT;\ ET,AC;ET,DSC;\ st^e;\ st^{t1};st^{t2}\}$
- $H_2 \ \epsilon \{AC;\ VCC;DSC\}$
- $\alpha(N_y) = N_y\, DR^{-1}(1-(1+DR)^{-N_y})$, with N_y being the number of operation years and DR being the discount rate of the considered case study.
- C_X^{hyb} is the cost of each component X in the hybrid system , and C_e is the cost of electrical energy from grid.

The cost of each component $X\epsilon H_1$ and $X\epsilon H_2$ of the hybrid system is calculated using equations (12) and (13), respectively.

$$C_{X\epsilon H_1} = \left(\left[\frac{N_y-1}{N_X^{lt}}\right] + 1\right) C_X^{hyb} = \varphi_{N_y}(N_X^{lt})\, C_X^{hyb} \tag{12}$$

Where $\mathrm{N_X^{lt}}$ is the life time of the component "X" and $\left[\frac{\mathrm{N_y}-1}{\mathrm{N_X^{lt}}}\right]$ denotes the integer part of $\frac{\mathrm{N_y}-1}{\mathrm{N_X^{lt}}}$. In addition, $N_{st^e} = 0$ if $(N_{PV} + N_{WT}) = 0$; $N_{st^{t1}} = 0$ if $N_{ET}^{AC} = 0$ and $N_{st^{t2}} = 0$ if $N_{ET}^{DSC} = 0$, because storage electrical and thermal systems are used jointly with the corresponding renewable energy system.

$$C_{X\epsilon H_2} = \sigma\,(Z)\varphi_{N_y}(N_X^{lt})\, C_X^{hyb} \tag{13}$$

Where $\sigma(Z) = \begin{cases} 0 \ if \ \ Z \leq 0 \\ 1 \ if \ \ Z > 0 \end{cases}$

with Z being defined as follows:

$$\mathrm{Z} = \begin{cases} \mathrm{E_L} - \left(\mathrm{COP_{AC}}\ \mathrm{N_{ET}^{AC}}\ \mathrm{E_{ET,AC}^{u,t}} + \mathrm{COP_{DSC}}\ \mathrm{N_{ET}^{DSC}}\ \mathrm{E_{ET,DSC}^{u,t}}\right) & \text{if } \mathrm{X} \equiv \mathrm{VCC} \\ \mathrm{N_{ET}^{AC}} & \text{if } \mathrm{X} \equiv \mathrm{AC} \\ \mathrm{N_{ET}^{DSC}} & \text{if } \mathrm{X} \equiv \mathrm{DSC} \end{cases}$$

Table 1 illustarets the lietime of each component found in the literature.

Table 1: Lifetime of hybrid cooling system components

Component	Lifetime (years)
Photovoltaic Panel	25
Wind Turbine	20
Evacuated Tube Solar Collector	40
Vapor compression chiller	15
Absorption Chiller	23
Desiccant System	10
Electric Storage	5
Thermal Storage	10

2.3. Optimization problem

Finally, considering the objective function and constraints, the optimal sizing problem can be stated as follows:

the objective function : $Max\ \langle \widehat{N}, \beta \rangle$

$$\begin{cases} \langle \widehat{N}\ , \gamma \rangle\ \leq\ A_T \\ \langle \widehat{N}\ , \rho \rangle\ \leq\ L \\ NPV_{VCC}^{cv} - \mathrm{NPV}_{\mathrm{hyb}}\ \leq\ 0 \end{cases}$$

Where NPV is predifiened in the previous section.

The problem is stated as a discrete problem with a limited number of cases, namely number of renewable energy systems. Thus, it can be solved by using a straight forward algorithm using Matlab. The optimal solution was found by searching the maximum of the objective function, of all combinations of the vector *N*, in such condition that all constraints are respected.

3. RESULTS AND DISCUSSIONS: CASE STUDIES

The introduced optimal sizing method applied for the proposed hybrid cooling system is investigated in two case studies: Beirut-Lebanon and Marseille- France. The input data concerning the climate of a typical meteorological year and the corresponding cooling load in both regions are obtained using Trnsys. Figures 2 and 3 illustrates the data necessary for energy balance equations and the objective function namely, cooling loads (latent, sensible and total load), thermal energy outputs of evacuated tube solar collectors, electric energy output of photovoltaic panels and the electric energy output of the wind turbine.

Additionally, one constraint of the optimal sizing problem is economic, which needs the price and life-time of each of the various system components as inputs in the way for obtaining the solution. Accordingly, the market prices as well as the life time of hybrid cooling system components in Beirut and Marseille are considered.

The simulations results obtained using Trnsys, the prices from the local market in each city and the lifetime of each system are introduced in the optimization problem. After that, Matlab software is utilized to model the optimal sizing method, where two time periods, 7 and 10 years, are considered for each case. In fact, 7 years represents the minimum number of operating years where the hybrid cooling system become feasible in case of Beirut; and 10 years in case of Marseille. Simulation results yield the optimum number of each energy unit composing the hybrid system. Table 2 illustrates these results for the each considered case all along with the capital cost of the system and the annual operating-cost savings compared with a conventional electric vapor compression chiller. It is shown that for the case of Beirut, the same components-configuration of the hybrid system is obtained for both studied operation periods (7 and 10 years). As for Marseille, the electric vapor compression chiller is preferred to any other hybrid configuration over the 7-years operation period. This is mainly attributed to lower electricity costs in the country compared with that in Beirut and thus, annual operating-cost savings would not be able to meet the economic criterion. On the other hand, for 10-years operation period, the same hybrid system configuration obtained for the case of Beirut is found. Moreover, comparing with the annual savings in both cities, it is clear that they are much higher in Beirut due to higher cooling loads and higher electricity tariff.

Furthermore, although the capital costs of the obtained hybrid system are high in both cities; however, annual cost-savings are also considerable. Hence, if good financial incentives and loans exist, there would be a large spread of such high efficient energy systems.

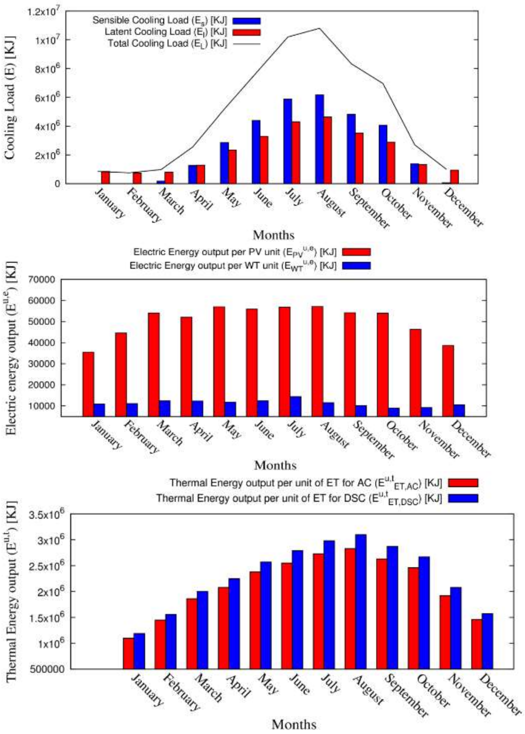


Figure 2: Input data concerning the climate of a typical meteorological year in Beirut-Lebanon

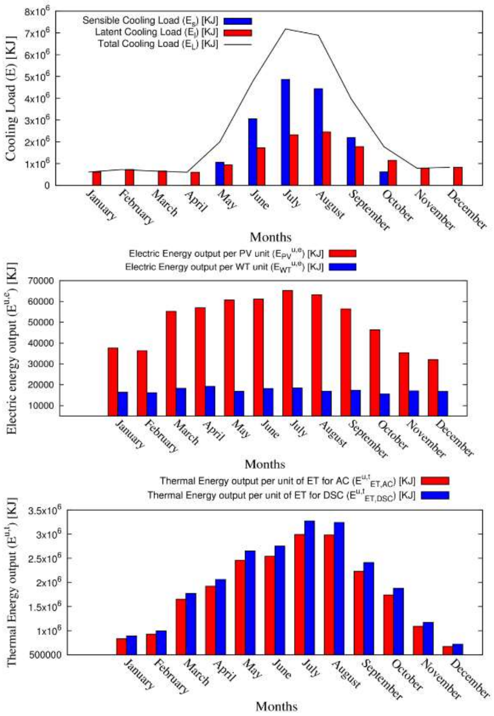


Figure 3: Input data concerning the climate of a typical meteorological year in Marseille-France

4. CONCLUSION

In this work, the optimal sizing of a hybrid cooling system is studied. For this purpose, a hybrid cooling system based on renewable energy system was defined. Then, an optimal sizing method was presented in order to obtain the best configuration of the hybrid cooling system in term of maximum renewable energy share and minimum grid contribution. The optimization problem was applied for two cases study namely, Beirut-Lebanon and Marseille-France. It was shown that the grid electricity cost has a significant role in determining the optimal size of a hybrid cooling system. For instance, at lower electricity cost in Marseille-France, a conventional cooling system could be more feasible than hybrid systems especially at short operation period. On the other hand, at high electricity cost, a hybrid cooling system could be more economical than conventional one, even if the prices of renewable energy systems are high.

Finally, this method need to be improved, and an optimal strategy for energy management and control - according to the cooling demand, energy input, costs, etc. -should be implemented. Hence, it allows the components of the system to operate at its maximal performance.

Table 2: Results for the optimal number for each system component as well as the capital cost and annual saving for both case studies.

Case Study	Beirut, Lebanon		Marseille, France	
Number of operation years (N_y) [years]	7	10	7	10
Number of PV [unit]	10	10	0	10
Number of WT [unit]	2	2	0	2
Number of ET, AC [unit]	2	2	0	2
Number of ET,DSC [unit]	1	1	0	1
Number of AC [unit]	1	1	0	1
Number of VCC [unit]	1	1	1	1
Number of DSC [unit]	1	1	0	1
Total Capital Cost [USD $]	32130	32130	1087	22242
Annual Saving [USD $/year]	795	795	0	253

NOMENCLATURE

Variables

A	*Area [m2]*
C	*Cost [$]*
COP	*Coefficient of Performance [-]*
N	*Number of Units/years [Units/years]*
RH	*Relative Humidity [%]*
T	*Temperature [°]*

Abbreviations

AC	*Absorption Chiller*
CHW	*Chilled Water tank*
COP	*Coefficient of performance*
DR	*Discount Rate*

DSC	*Desiccant System*
ET,AC	*Evacuated tube solar collector for Absorption chiller*
ET,DSC	*Evacuated tube solar collector for Desiccant system*
HW	*Hot Water tank*
Hyb	*Hybrid*
NPV	*Net Present Value*
PV	*Photovoltaic*
VCC	*Vapor Compression Chiller*
WT	*Wind Turbine*

Subscripts

chw	*Chilled water*
cv	*Conventional*
e	*Electric*
gen	*Generator*
i	*Input*
l	*Latent*
l	*load*
lt	*Lifetime*
max	*Maximum*
min	*Minimum*
o	*Output*
s	*Sensible*
sp	*Set point*
st	*Storage*
t	*Thermal*
u	*Unit*
y	*Years*

List of References

Kairouani L., Nehdi E., 2006. Cooling performance and energy saving of a compression–absorption refrigeration system assisted by geothermal energy. Applied Thermal Engineering, 26 (2-3), 288–294.

Poulet P., Outbib R., 2015. Energy production for dwellings by using hybrid systems based on heat pump variable input power. Applied Energy, 147 (1), 413–429.

Ling J., Kuwabara O., Hwang Y., Radermacher R., 2013. Enhancement options for separate sensible and latent cooling air-conditioning systems. International Journal of Refrigeration , 36 (1), 45–57.

Ling J., Kuwabara O., Hwang Y., Radermacher R., 2011. Experimental evaluation and performance enhancement prediction of desiccant assisted separate sensible and latent cooling air-conditioning system. International Journal of Refrigeration, 34 (4), 946–957.

Ibrahim O., Fardoun F., Younes R., Louahlia-Gualous H., 2014. Optimal management proposal for hybrid water heating system. Energy and Buildings, 75, 342–35

Ge T.S., Dai Y.J., Wang R.Z., 2014. Review on solar powered rotary desiccant wheel cooling system. Renewable and Sustainable Energy Reviews, 39, 476–497.

BOND GRAPH MODEL CONDITIONING FOR ANALYSIS, SIMULATION AND CONTROL SYSTEM DESIGN: *APPLICATION TO A PLANAR MOBILE ROBOTIC MANIPULATOR*

Matías Nacusse[a,b], Martín Crespo [a,b], Sergio Junco [a], Vitalram Rayankula[c], Pushparaj Mani Pathak[c]

[a] Laboratorio de Automatización y Control (LAC), Departamento de Control, FCEIA, UNR. Rosario, Argentina
[b] CONICET: Consejo Nacional de Investigaciones Científicas y Técnicas. Argentina
[c] Department of Mechanical and Industrial Engineering, Indian Institute of Technology Roorkee, India

[a]{nacusse, crespom, sjunco}@fceia.unr.edu.ar
[c]rvitalram@gmail.com, pushpfme@iitr.ac.in

ABSTRACT

The appearance of algebraic constraints among energy variables in models of physical systems leads to sets of (possibly nonlinear) implicit state equations, which usually complicate the treatment of the problems to be solved on the model. Building up on the Bond Graph model of a Planar Mobile Robotic Manipulator, this paper discusses some techniques to handle this kind of situations, determined here by the coupling of rigid bodies. Two alternatives to break the constraints are presented, consisting in the insertion between the coupled elements of: a) parasitic components –mostly spring-dampers, which is standard practice– or b) residual sinks –which is equivalent to the practice of adding constraint forces. Modifying the Bond Graph through the introduction of storage fields is the third method presented. Further, the extraction of constraint-free Euler-Lagrange and Hamiltonian descriptions from the Bond Graph are addressed. Finally, the suitability of all of these five alternatives for the purposes of simulation, analysis and control system design are discussed, and illustrated with simulation results.

Keywords: Planar mobile manipulator, Bond Graphs, Euler-Lagrange equations, Port-Hamiltonian systems, Simulation, Nonlinear control.

1. INTRODUCTION

The multidomain nature of modern engineering systems has renewed the interest in energy-based modeling formalisms. This is above all true in Mechatronics in general and Robotics in particular. Euler-Lagrange (EL) modeling is the classical approach in Robotics (Siciliano et al., 2009). Since relatively recent times the Hamiltonian formalism under the new, extended Port-Hamiltonian System (PHS) version has also been considered for the purposes of nonlinear control system synthesis, mainly in the framework of Passivity-Based Control (PBC) (Ortega et al., 2002). Besides these two modeling approaches rooted in Classical Physics, the Bond Graph (BG) technique, an engineering graphical modeling method, has gained importance (Karnopp, Margolis and Rosenberg, 2006), (Merzouki et al., 2012). It uses a reduced and unified set of symbols –which describe basic energy phenomena and interconnection structure in a physical system– able to represent not only the mechanical parts of a multidomain system, but also the electromechanical actuators and their associated power electronic converters, pneumatic or hydraulic actuators, etc.

One of the advantages of BGs is their modular or object-oriented modeling nature which allows to construct the whole system model by coupling the models of its subsystems. However, this advantage comes with a drawback in certain cases: when the system order is lower than the number of energy variables, there appear algebraic constraints among them, fact that leads to sets of implicit state equations (Karnopp, Margolis and Rosenberg, 2006). Graphically, this translates into storages being in *derivative causality* and the presence of algebraic loops or zero order causal paths (ZCP), see (van Dijk and Breedveld, 1991) for a classification. This is a recurrent problem when modeling mechanical systems, especially in systems with kinematic constraints. The existence of ZCP always implies that the set of state equations is an implicit set of Differential-Algebraic Equations (DAEs) with the consequent problems to the numerical solvers, see for example (van Dijk and Breedveld, 1991) or (Cacho, Felez and Vera, 2000) for the numerical issues in solving these systems. Modern modeling and simulation (M&S) software is equipped with tools capable to handle implicit systems, but they frequently fail when the algebraic dependence is complex. In these cases, while still useful for direct analysis of some system properties, the BG models are not immediately useful for simulation or control system design. They must be suitably adapted for those purposes. To deal with ZCPs the authors in (Karnopp and Margolis, 1979) add some parasitic components to break the causal loops. While this classic engineering approach is simple to implement, it is not always evident how to parameterize the parasitic components, and results in highly stiff models of higher order which increase the total simulation time and, occasionally, the numerical errors too. An alternative technique to break the ZCPs without modifying the dynamics consists in adding residual sinks (rS) in adequate positions of the causal paths (Gawthrop and Smith, 1992) (Borutzky and Cellier, 1996). This resource has been incorporated in certain simulation tools via programming commands, called "*constraint*" in some of them (20sim®, for instance, see (Controllab Products, n.d.)). This method solves the problem calculating efforts that enforce the geometric or kinematic constraints producing the ZCP. As it proceeds numerically at each simulation step, it could increase the total simulation time and accumulate errors beyond admissible limits. This problem can be circumvented when the constraint can be explicitly solved and embodied in the residual sink (Nacusse and Junco, 2017), as it is done in the example case treated in this paper. A radical different approach is to collect the dependent storage phenomena in a so called *storage field* (Karnopp, 1992). This approach, while abandoning the modular feature of the BGs, yields a model without ZCPs between energy storage elements. Another solution, which is directly in line with the construction of the storage field, consists in choosing sets of independent coordinates in the BG model and to derive EL- or PHS-models from it (Karnopp, 1977) (Donaire and Junco, 2009) (van der Schaft and Jeltsema, 2014).

The main goal of this paper is to present the derivation and discuss the application of the full panoply of the previously mentioned alternative models starting from a BG of a planar mobile robotic manipulator (PMRM) featuring derivative causality. Even if issues related to analysis and control synthesis and design are considered, the main stress is put on simulation matters.

The rest of the paper is organized as follows. Section 2 briefly reviews the EL and PHS formalisms and describes the PMRM. The construction of the BGs is done in Section 3. First the base BG in derivative causality is built and then it is shown how to get the models with rS or parasitic elements to break the ZCPs. Second, following the procedure detailed in (Karnopp, 1992), a BG featuring an *IC-Field* is derived as a means to avoid the derivative causality. Section 4 compactly discusses the derivation of EL and PHS models from the BGs previously introduced. Section 5 presents some simulation results with the PMRM in closed-loop to compare the differents approaches in terms of simulation time and error. Finally, some conclusions are given in Section 6.

2. MODELING FORMALISMS AND SYSTEM DESCRIPTION

2.1. Modeling formalisms

The EL equations are probably the most classical approach to modeling in the field of robot dynamics. These equations are obtained performing the operations indicated in (1) on the Lagrangian function $\mathcal{L}(q,\dot{q}) = \mathcal{T}^*(\dot{q}) - \mathcal{V}(q)$, where $\mathcal{T}^*(\dot{q})$ is the kinetic co-energy, $\mathcal{V}(q)$ is the potential energy, q and $\dot{q}$ are the vector of generalized coordinates and velocities respectively, and E is the vector of generalized non-conservative forces.

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial\mathcal{L}}{\partial\dot{\mathrm{q}}}\right) - \frac{\partial\mathcal{L}}{\partial \mathrm{q}} = \mathrm{E} \tag{1}$$

Since more recently, a Hamiltonian formulation of the system dynamics called PHS has gained importance in robotics because of the benefits it offers through the application of the Passivity-Based Control (PBC) techniques to robot control problems (Ortega et al., 2001). Classically, the Hamilton equations of motion are derived from (1) via the definition of the generalized momenta $p := \partial\mathcal{L}/\partial\dot{q}$ and the Legendre transformation $\mathfrak{S}\{\mathcal{L}(q,\dot{q})\} := p^T\dot{q} - \mathcal{L}(q,\dot{q})$. This yields the Hamiltonian state function as $\mathcal{H}(q,p) = \mathfrak{S}\{\mathcal{L}(q,\dot{q})\} = T(p,q) + V(q)$, where $T(p,q)$ is the kinetic energy in the system. Furthermore, operating on (1), and assuming a system with n degrees of freedom, the following Hamiltonian model is obtained:

$$\underbrace{\begin{bmatrix}\dot{\mathrm{q}}\\ \dot{\mathrm{p}}\end{bmatrix}}_{\dot{\mathrm{x}}} = \underbrace{\begin{bmatrix}0_{\mathrm{nxn}} & \mathrm{I}_{\mathrm{nxn}}\\ -\mathrm{I}_{\mathrm{nxn}} & 0_{\mathrm{nxn}}\end{bmatrix}}_{\mathrm{J}}\underbrace{\begin{bmatrix}\frac{\partial\mathcal{H}(\mathrm{x})}{\partial \mathrm{q}}\\ \frac{\partial\mathcal{H}(\mathrm{x})}{\partial \mathrm{p}}\end{bmatrix}}_{\nabla_{\mathrm{x}}\mathcal{H}(\mathrm{x})} + \begin{bmatrix}0_{\mathrm{n}}\\ \mathrm{E}\end{bmatrix} \tag{2}$$

This classical model is a particular case of the PHS form (3), which explicitly shows the presence of dissipation in the system, and admits state variables other than q and p describing the system dynamics (van der Schaft and Jeltsema, 2014).

$$\dot{x} = [J(x,u) - R(x,u)]\,\nabla_x\mathcal{H} + g(x)\,u \tag{3}$$

Here $J(x,u)$ is the structure or interconnection matrix which conserves the skew-symmetric property of J in the classical formulation, but does not need to be composed by unitary submatrices. The dissipation matrix $R(x,u)$ is a symmetric, positive (semi–) definite matrix, and $g(x)$ is a matrix weighting the inputs u. This latter vector is composed of control and, possibly, disturbance inputs. The presence of x and u as arguments of J and R takes into account the fact that some interconnection or dissipation elements could be state-dependent (the magnetic flux modulating the electromechanical power exchange in an electric actuator, for instance) in the case of x, and that control actions u could be exerted through the interconnection structure (as in power electronic converters feeding electromechanical actuators) or the dissipation structure (as in a hydraulic control valve).

In this paper the BG approach, an energy-based graphical modeling formalism (Karnopp, Margolis and Rosenberg, 2006), is resorted to as the primary modeling tool. Indeed, starting from a first BG model, other BGs are derived aiming at different purposes, as well as an EL and a PHS model of the PMRM under consideration.

2.2. System description

The physical system, shown in Figure 1, is a PMRM consisting of a mobile base (MB) coupled to a manipulator arm (MA) with two rigid links connected by revolute joints. The MB is a circular platform driven by three independent Omniwheels symmetrically located at 120 degrees each other.

The following set of fourteen relevant coordinates can be distinguished in the sketch of Figure 1:

$$\{\Omega_{w1}, \Omega_{w2}, \Omega_{w3}, x_0, y_0, \phi_0, \phi_1, \phi_2, x_1, y_1, x_2, y_2, \theta_1, \theta_2\} \quad (4)$$

where: $\Omega_{w1,2,3}$ are the rotational angular positions of the three wheels; $x_{0,1,2}$ and $y_{0,1,2}$ are the positions of the centers of mass (CoM) of the MB and the two links with respect to the inertial Cartesian frame (X_F, Y_F); $\phi_{0,1,2}$ specify, respectively, the angular positions of the attachment point of link 1 to the MB and of links 1 and 2, all the three of them measured with respect to the X_F axis; and $\theta_{1,2}$ are the relative angular positions of the links, as indicated in Figure 1. This is not a set of independent coordinates. As the three wheels are independently actuated, a restriction between the subsets $\{\dot{\Omega}_{w1}, \dot{\Omega}_{w2}, \dot{\Omega}_{w3}\}$ and $\{\dot{x}_0, \dot{y}_0, \dot{\phi}_0\}$ can be derived, which can be expressed by an invertible matrix, as done in the next section. Under the assumption of holonomy of the MB, this relationship enforces a restriction among the respective coordinates. Also the subsets $\{\theta_1, \theta_2\}$ and $\{\phi_1, \phi_2\}$ are each other dependent. Finally, as shown in equation (5), $x_{1,2}$ and $y_{1,2}$ can be written in terms of $x_0, y_0, \phi_0, \phi_1, \phi_2$. As usual in Robotics, the short notations $s_\phi = \sin(\phi)$ and $c_{\phi_1} = \cos(\phi)$ have been used.

$$\begin{aligned}
x_1 &= x_0 + l_b c_{\phi_0} + l_1 c_{\phi_1} \\
y_1 &= y_0 + l_b s_{\phi_0} + l_1 s_{\phi_1} \\
x_2 &= x_0 + l_b c_{\phi_0} + l_1 c_{\phi_1} + l_1 c_{\phi_1} + l_2 c_{\phi_2} \\
y_2 &= y_0 + l_b s_{\phi_0} + l_1 s_{\phi_1} + l_1 s_{\phi_1} + l_2 s_{\phi_2}
\end{aligned} \quad (5)$$

The previous discussion shows that the system has five degrees of freedom (DoF). Different subsets out of the whole set of fourteen coordinates could be used according to the modeling approach followed. When modeling a manipulator in the EL context is a common practice to consider the relative angles between links as generalized coordinates, and the torques applied at the joints as generalized input forces. The corresponding choice of the vector q would then be $q_\theta = [x_0, y_0, \phi_0, \theta_1, \theta_2]$. As the EL model presented in this paper is derived from a BG, which considers inertial velocities, the angles ϕ_1, ϕ_2 measured with respect to the inertial frame will be used, instead of the relative angles θ_1, θ_2. To this choice corresponds the set $q = [x_0, y_0, \phi_0, \phi_1, \phi_2]^T$. No matter which set of independent generalized coordinates is used, the general form of a robot model derived from (1) is given in equation (6).

$$M(q)\,\ddot{q} + C(q, \dot{q})\,\dot{q} + R(q)\,\dot{q} + G(q) = W(q)\,T \quad (6)$$

Here $M(q)$ is the symmetric, positive definite inertia matrix, $C(q, \dot{q})$ is the Coriolis and Centrifugal matrix, $R(q)$ is the matrix of dissipative forces, $G(q)$ is the matrix of gravitational forces and $W(q)$ is a matrix that weights the input torques T. Notice that, for the robot under consideration, $R(q)$ and $G(q)$ are equal to zero since no friction and only planar movement on a horizontal base have been considered. Under these conditions, the model reduces to (7).

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = W(q)\,T \quad (7)$$

The first BG presented, from now on called *the base BG* and denoted BG1, will be constructed relating its 1-junctions to the velocities associated to the dependent set $q_{BG1} = [\Omega_{w1}, \Omega_{w2}, \Omega_{w3}, x_0, y_0, \phi_0, x_1, y_1, x_2, y_2, \phi_1, \phi_2]^T$. As it has 12 components, i.e., 7 more than the number of DoF, there are algebraic dependencies among their associated velocities, fact that will produce a BG with dependent storages, leading to what is called –in the jargon of the BG community – derivative causality or ZCP.

3. BG MODEL OF THE PLANAR ROBOTIC MANIPULATOR

The base BG is built following the standard procedure for mechanical systems briefly detailed below (Karnopp, Margolis and Rosenberg, 2006). It needs the relationships among the different velocities considered in the model, i.e., among the time derivatives of the components of vector q_{BG1}.

3.1. Construction of the base BG

Equation (8), expressing the relationships between the rotational velocities of the wheels, i.e. $\dot{\Omega}_w = \left[\dot{\Omega}_{w1}, \dot{\Omega}_{w2}, \dot{\Omega}_{w3}\right]^T$, and the velocities associated to the MB, i.e, $\dot{q}_{1-3} = \left[\dot{x}_0, \dot{y}_0, \dot{\phi}_0\right]^T$, can be obtained analyzing the geometry of the MB and the arrangement of the wheels. The shorthand notation $q_{1-3} := [x_0, y_0, \phi_0]^T$ has been used here. On the other hand, equation (9), obtained taking the time derivatives of (5), expresses the relationships between the Cartesian velocities of the CoM of the links, i.e. $\dot{z} := [\dot{x_1}, \dot{y_1}, \dot{x_2}, \dot{y_2}]^T$, and the vector of velocities associated to the coordinates $q = [q_{1-3}, \phi_1, \phi_2]^T$ of the MB-MA. $J_1(\phi_\mathrm{i})$ is the Jacobian associated to (5).

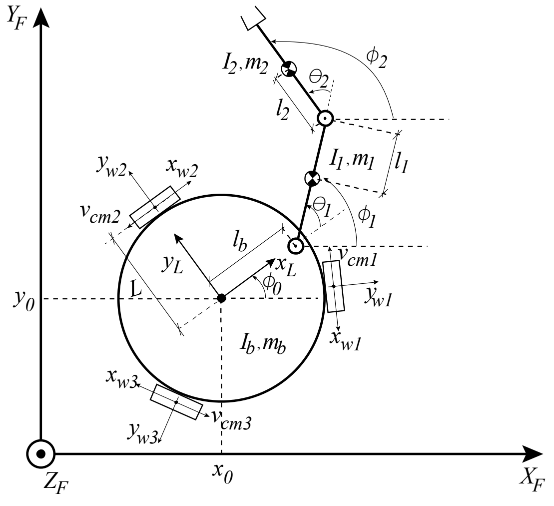


Figure 1: Schematics of the PMRM physical system.

$$\dot{\Omega}_w = \underbrace{\frac{1}{r}\begin{pmatrix} c_{\phi_0-\frac{\pi}{3}} & s_{\phi_0-\frac{\pi}{3}} & -L \\ c_{\phi_0} & s_{\phi_0} & -L \\ c_{\phi_0+\frac{\pi}{3}} & s_{\phi_0+\frac{\pi}{3}} & -L \end{pmatrix}}_{A_r(\phi_0)=\frac{1}{r}A(\phi_0)} \dot{q}_{1-3} \tag{8}$$

$$\dot{z} = \underbrace{\begin{pmatrix} 1 & 0 & -l_b s_{\phi_0} & -l_1 s_{\phi_1} & 0 \\ 0 & 1 & l_b c_{\phi_0} & l_1 c_{\phi_1} & 0 \\ 1 & 0 & -l_b s_{\phi_0} & -2l_1 s_{\phi_1} & -l_2 s_{\phi_2} \\ 0 & 1 & l_b c_{\phi_0} & 2l_1 c_{\phi_1} & l_2 c_{\phi_2} \end{pmatrix}}_{J_1(\phi_i)} \dot{q} \tag{9}$$

The BG interconnection structure is built using (8) and (9) by putting a 1-junction to represent each velocity, 0-junctions and MTFs to represent the sum of flows according to (9), and MTFs to represent the modulated power transfer among the different parts of the system. Finally the base BG, shown **in black color in** Figure 2, is completed by putting the I-elements (representing the storage of kinetic energy in the system) and the effort sources that model the torque inputs.

The causality assignment is carried out using the standard sequential causality assignment procedure (SCAP) (Karnopp, Margolis and Rosenberg, 2006). Integral causality can be assigned only to a proper subset of the I-elements of this BG, the choice being those attached to the 1-junctions associated with the $\dot{q}$ velocities. This restriction, as already anticipated due to the holonomic constraints (5) among the coordinates of the links, induces differential causality in some of the storage elements of the BG, with possible negative effects in numerical simulations of the model. To make it suitable for simulation tests, the BG of Figure 2 is adapted in the next subsection. For reasons of space and the sake of better understanding, the modifications –which consist in adding some elements and changing some causal strokes– are shown on the same Figure 2, with **the new elements represented in red color**.

3.2. BG for simulation purposes

Most current M&S software, particularly 20sim (Controllab Products, n.d.), a tool accepting model specification in the form of BG (among other usual formalisms), can deal with models featuring differential causality, i.e., in the presence of ZCPs. But not all M&S software applications are equipped with this tool, which is particularly necessary when the dependency among the storages is strongly nonlinear. This is the case of the BG1, where this dependency is given by the **MTF** structure having the matrices given in equations (8) and (9) as gains. Three different approaches helping to deal with ZCPs are treated next. The first one consists in breaking the ZCPs by adding effort residual sinks (**rSe**) elements. A residual sink is a computational device that injects the necessary effort (or flow) in order to make vanish the power conjugated variable entering into the sink, see the Appendix for a brief description. This element can be interpreted in the EL framework as the addition of Lagrange multipliers in the Lagrangian function. The second one breaks the causal loops by adding parasitic components between the statically coupled storages. As in this case these are I-elements, the pertinent parasitic components are C-elements, that must be of high stiffness in order to alter the dynamics the less possible. The third methodology eliminates the causal loops representing the whole energy storage by means of an *IC-Field* (Karnopp, 1992). Its equivalent is the use of a set of independent coordinates in the EL framework.

3.2.1. BG with rS to enforce integral causality

The addition of **rSe** done in red color in Figure 2 converts the implicit system of DAEs associated to the base BG into the explicit systems of DAEs, shown in (10), subject to restrictions (8) and (9).

$$\begin{aligned} M_1 \ddot{q} &= B(q)\,\lambda + W(q)\,T \\ M_2\,\ddot{z} &= C\,\lambda_{xy} \\ M_\Omega\,\ddot{\Omega}_w &= \tau_\Omega - \lambda_\Omega \end{aligned} \tag{10}$$

Where $M_1 = diag(m_b, m_b, I_b, I_1, I_2)$, $\lambda = [\lambda_\Omega, \lambda_{xy}]^T$, $\lambda_{xy} = [\lambda_{x1}, \lambda_{y1}, \lambda_{x2}, \lambda_{y2}]^T$, $\lambda_\Omega = [\lambda_{\Omega 1}, \lambda_{\Omega 3}, \lambda_{\Omega 3}]^T$, $\tau_\Omega = [\tau_{\Omega 1}, \tau_{\Omega 2}, \tau_{\Omega 3}]^T$, $T = [\tau_\Omega, \tau_1, \tau_2]^T$, $M_\Omega = diag(I_{w1}, I_{w2}, I_{w3})$, $M_2 = diag(m_1, m_1, m_2, m_2)$,

$$B(q) = \left[\begin{array}{ccccc} & -1 & 0 & 0 & 0 \\ A_r(\phi_0) & 0 & -1 & 0 & 0 \\ & l_b S_{\phi_0} & -l_b C_{\phi_0} & 0 & 0 \\ 0_{2X3} & l_1 S_{\phi_1} & -l_1 C_{\phi_1} & l_1 S_{\phi_1} & -l_1 C_{\phi_1} \\ & 0 & 0 & l_2 S_{\phi_2} & -l_2 C_{\phi_2} \end{array}\right]$$

$$W(q) = \left[\begin{array}{ccc} & 0 & 0 \\ {A_r}^T(\phi_0) & 0 & 0 \\ & -1 & 0 \\ 0_{2X3} & 1 & -1 \\ & 0 & 1 \end{array}\right], C = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The variables $\lambda_{x1,2}$, $\lambda_{y1,2}$ and $\lambda_{\Omega 1,2,3}$ are the outputs of the residual sinks, see Figure 2.

Remark 1. It is stressed here that (10) is equivalent to the operations performed by the simulator solver processing the BG model, i.e., obtaining this equation system is not a task for the user of the M&S software.

Even with M&S software able to solve numerically the added constraints at each integration step, as 20Sim does for this example, it is in general convenient to solve the algebraic restrictions explicitly offline, and then to add them to the model, as this strongly reduces the computational cost of the simulation. This is done by taking the time derivative over (8) and (9) which results in (11)

$$\begin{aligned} \ddot{z} &= J_1(\phi_i)\ddot{q} + \dot{J}_1(\phi_i)\dot{q} \\ \ddot{\Omega}_{\mathrm{w}} &= [A_r(\phi_0) \quad 0_{3x2}]\ddot{q} + [\dot{A}_r(\phi_0) \quad 0_{3x2}]\dot{q} \end{aligned} \tag{11}$$

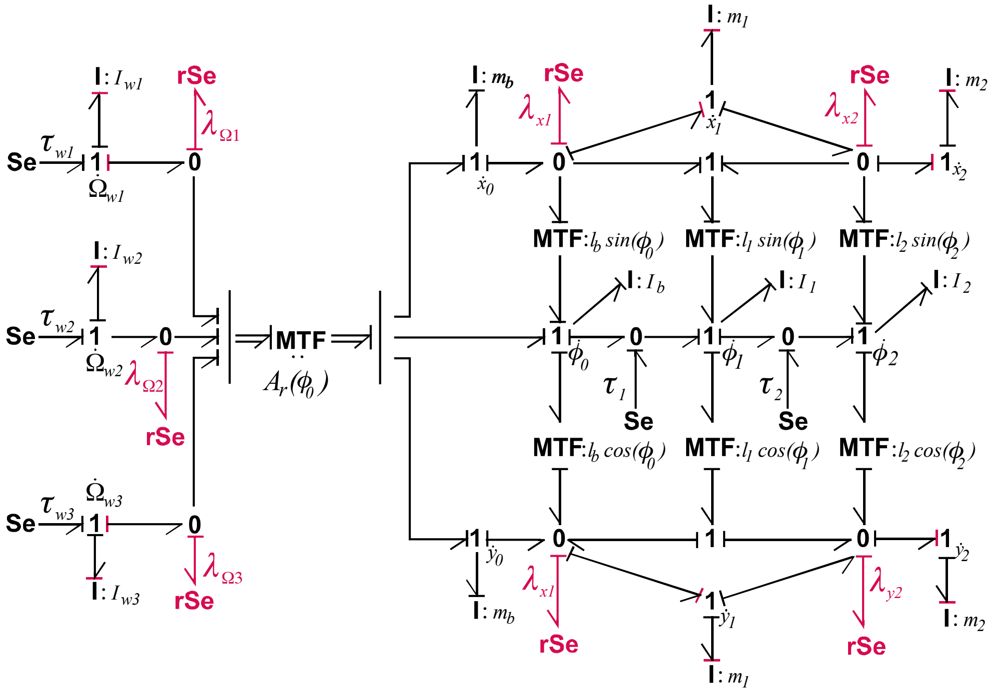


Figure 2: BG model of the planar robotic manipulator. **In black**: base BG with derivative causality. **In red**: residual sinks added to enforce integral causality in all the storage elements, and modified causal strokes for the originally dependent storages.

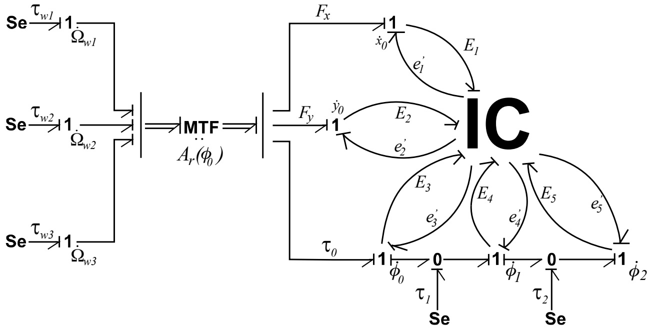


Figure 3: BG model using an *IC-Field*.

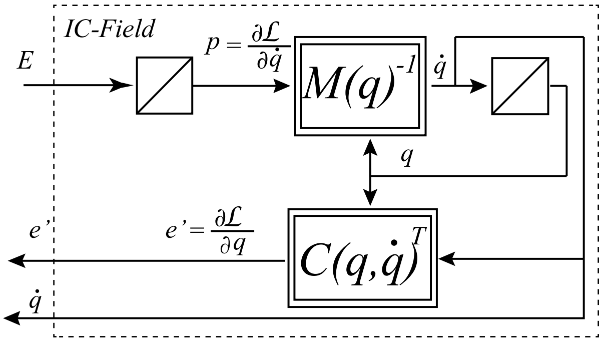


Figure 4: Block diagram of the *IC-Field*.

$$
\begin{gathered}
M_{bl}(q) = \begin{pmatrix}
m_b + m_1 + m_2 & 0 & -(m_1+m_2)l_b S_{\phi_0} & -(m_1+2m_2)l_1 S_{\phi_1} & -m_2 l_2 S_{\phi_2} \\
0 & m_b + m_1 + m_2 & (m_1+m_2)l_b C_{\phi_0} & (m_1+2m_2)l_1 C_{\phi_1} & m_2 l_2 C_{\phi_2} \\
* & * & I_b + m_1 l_b^2 + m_2 l_b^2 & (m_1+2m_2)l_1 l_b C_{\phi_1-\phi_0} & m_2 l_2 l_b C_{\phi_2-\phi_0} \\
* & * & * & I_1 + m_1 l_1^2 + 4m_2 l_1^2 & 2m_2 l_1 l_2 C_{\phi_2-\phi_1} \\
* & * & * & * & I_2 + m_2 l_2^2
\end{pmatrix} \\
C_{bl}(q,\dot{q}) = \begin{pmatrix}
0 & 0 & -(m_1+m_2)l_b C_{\phi_0}\dot{\phi}_0 & -(m_1+2m_2)l_1 C_{\phi_1}\dot{\phi}_1 & -m_2 l_2 C_{\phi_2}\dot{\phi}_2 \\
0 & 0 & -(m_1+m_2)l_b S_{\phi_0}\dot{\phi}_0 & -(m_1+2m_2)l_1 S_{\phi_1}\dot{\phi}_1 & -m_2 l_2 S_{\phi_2}\dot{\phi}_2 \\
0 & 0 & 0 & -(m_1+2m_2)l_1 l_b S_{\phi_1-\phi_0}\dot{\phi}_1 & -m_2 l_2 l_b S_{\phi_2-\phi_0}\dot{\phi}_2 \\
0 & 0 & (m_1+2m_2)l_1 l_b S_{\phi_1-\phi_0}\dot{\phi}_0 & 0 & -2m_2 l_1 l_2 S_{\phi_2-\phi_1}\dot{\phi}_2 \\
0 & 0 & m_2 l_2 l_b S_{\phi_2-\phi_0}\dot{\phi}_0 & 2m_2 l_1 l_2 S_{\phi_2-\phi_1}\dot{\phi}_1 & 0
\end{pmatrix}
\end{gathered}
\tag{12}
$$

$$
\begin{aligned}
\lambda = {} & \left(\begin{bmatrix} M_\Omega & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} A_r(\phi_0) & 0_{3x2} \\ & J_1(\phi_i) \end{bmatrix} M_1^{-1} B(q) - \begin{bmatrix} -I_{3x3} & 0_{3x4} \\ 0_{4x3} & C \end{bmatrix} \right)^{-1} \\
& \left\{ - \begin{bmatrix} M_\Omega & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} A_r(\phi_0) & 0_{3x2} \\ & J_1(\phi_i) \end{bmatrix} M_1^{-1} W(q)\tau - \begin{bmatrix} M_\Omega & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} \dot{A}_r(\phi_0) & 0_{3x2} \\ & \dot{J}_1(\phi_i) \end{bmatrix} \dot{q} + \begin{bmatrix} \tau_\omega \\ 0_{3x1} \end{bmatrix} \right\}
\end{aligned}
\tag{13}
$$

Then, replacing (11) into the second and third equation of (10) results in (14).

Further, taking the first equation of (10) into (14) results in an explicit solution for λ as in (13). Then using (13) in (10) an explicit set of state equations is obtained.

Remark 2. It is stressed here that the BG-processor takes care of solving (10) (implicit in the BG) and (13), which must be first obtained by the user and then programmed in the BG in place of the rS.

$$\begin{bmatrix} M_\Omega & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} A_r(\phi_0) & 0_{3x2} \\ \multicolumn{2}{c}{J_1(\phi_i)} \end{bmatrix} \ddot{q} + \begin{bmatrix} M_\Omega & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} \dot{A}_r(\phi_0) & 0_{3x2} \\ \multicolumn{2}{c}{\dot{J}_1(\phi_i)} \end{bmatrix} \dot{q} = \begin{bmatrix} -I_{3x3} & 0_{3x4} \\ 0_{4x3} & C \end{bmatrix} \lambda + \begin{bmatrix} \tau_\Omega \\ 0_{3x1} \end{bmatrix} \tag{14}$$

3.2.2. BG with parasite components to enforce integral causality

The rS can be interpreted as the limit case of a storage element with its internal parameter tending to zero (Borutzky and Cellier, 1996). Thus, a non-ideal implementation of a rS would be reached assigning a very low value to this parameter. As it would cause a response with high frequency contents, the addition of a R component of convenient value is suggested to quickly damp the fast dynamics. Moreover, these parasite components, can model the elasticity and the friction between links occurring at the bushings.

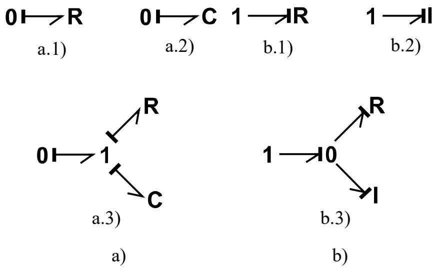


Figure 5: Residual sink replacing options for. a) **rSe**, b) **rSf**.

The use of parasite components to break algebraic loops is the most common solution among BG practitioners (Karnopp and Margolis, 1979) (Karnopp, Margolis and Rosenberg, 2006) as a method to eliminate derivative causality in multibody system models due to constraints introduced by mechanism joints, because its simplicity to achieve an explicit state equation set suitable for simulation without the need of extra calculations. As counterpart, the parametrization task of this extra component is usually difficult; moreover, this practice results in numerical stiff models. A parameter selection method based on the energetic activity of the parasitic components can be found in (Rideout and Stein, 2003) as well as an account of other techniques previously contributed within the bond graph community. Using this approach the BG model can be obtained replacing the rS components by the options depicted in Figure 5.

3.2.3. BG with storage IC-Field

The BG-theory recognizes explicit and implicit fields (Karnopp, Margolis and Rosenberg, 2006). The latter results when incorporating several (possibly single-port) components of akin type into a unique multiport device. This technique can be applied to BGs with derivative causality in order to eliminate it by subsuming the dependent storage elements with others in integral causality, building in this way one or more storage fields. Here all the energy stored in the system will be captured in a unique mixed-energy *IC-Field*. Even if there is no potential energy in the system, the C-part in the field appears because the topological coupling through the MTFs among the I-elements depends on the coordinates.

There are alternative ways of doing this, but all of them rely on the same rationale. It will be explained at an abstract level considering the EL model (7), taking into account that it can be extracted from the base BG of Figure 2, i.e, there is a direct correspondence between both descriptions. The details can be checked in (Karnopp, 1992). Equation (7) is a (vector) effort balance, which in a BG would occur at a (vector- or field-) 1-junction. The first term on the left-hand side would be the time derivative of the momentum-vector $p = \partial\mathcal{L}/\partial\dot{q}$ of an I-multiport attached to the 1-junction via a multibond, the second term would be the effort generated by a C-multiport attached to the same junction. Due to the dependence of both matrices M and C on q, both energy-storing fields constitute in fact a unique mixed *IC-field*. The second member of the equation is interpreted as the action of the (vector of) sources acting on the system through a $\boldsymbol{MTF}$ of matrix modulus $W(q)$ and the 1-junction. See the detailed expressions ofthe matrices $M(q)$ and $C(q,\dot{q})$ in equations (15), (16) and(12).

The resulting BG model is shown in Figure 3. Every vector component referred-to above, up to the *IC-Field*, is shown in its details, i.e., via its scalar components (the three 1-junctions on the left excluded, as they do not belong to the vector 1-junction; they are just shown to improve the understanding of the BG through the annotation of the wheels velocities). The constitutive relationships of the storage *IC-Field* defined in this way is depicted in the block diagram of Figure 4, where each integrator symbol stands for 5 scalar integration operations. This model solves the problems associated with differential causality assignment in the storage elements via the representation of the whole energy in a single storage field.

4. FROM BG MODELS TO EL AND PHS MODELS

In this section the method presented in (Karnopp, 1977) is employed to derive the EL model from the base BG. Then the PHS model is derived from the BG model with the storage *IC-Field.*

4.1. From BG to EL equations.

The procedure to obtain the EL equations of motions from a BG (Karnopp, 1977) is briefly detailed next:

1. Assign causality to all effort and flow sources (Se and Sf) and extend the causality through the structure of the model.

2. Choose a 1-Junction for which the flow has not been imposed and add an artificial flow source (Sf) to this junction.
3. Assign causality to the artificial flow source and extend it through the structure of the model.
4. Repeat the step 2 and 3 until all the bonds have been causally oriented.
5. Label the flows imposed by the artificial flow sources as $\dot{q}_i$.
6. Using the standard equation-reading procedure based on the causality assignment, read the incoming efforts to the artificial flow sources and express them in terms of the $\dot{q}_i$.

Placing the *artificial flow sources* at the 1-junctions associated with the vector $\dot{q} = \left[\dot{x}_0, \dot{y}_0, \dot{\phi}_0, \dot{\phi}_1, \dot{\phi}_2\right]^T$, i.e. *choosing* the outputs of the storage elements in integral causality in the BG of Figure 2 as generalized velocities $\dot{q}_i$, would yield $q = [x_0, y_0, \phi_0, \phi_1, \phi_2]^T$ as *the vector of generalized coordinates*. Because of space restrictions the procedure just described is not shown graphically here, but it is stressed that following it the EL equations derived from the base BG of Figure 2 are the same given in (7), with the following particular expressions for the matrices $M(q)$ and $C(q, \dot{q})$ (detailed expressions for $M_{bl}(q)$ and $C_{bl}(q, \dot{q})$ are given in (12)):

$$M(q) = M_{bl}(q) + \begin{bmatrix} A^T(\phi_0) M_\Omega A(\phi_0) & 0_{3X2} \\ 0_{2X3} & 0_{2X2} \end{bmatrix} \tag{15}$$

$$C(q, \dot{q}) = C_{bl}(q, \dot{q}) + \begin{bmatrix} A^T(\phi_0) M_\Omega \dot{A}(\phi_0) & 0_{3X2} \\ 0_{2X3} & 0_{2X2} \end{bmatrix} \tag{16}$$

4.2. From BG to PHS models.

The method presented in (Donaire and Junco, 2009) to obtain a PHS from the BG model of Figure 2 cannot be applied directly since the causal path that relates the storage elements in derivative and integral causality pass through a **MTF** which depends on the state variables. However, see (Donaire and Junco, 2009) for a detailed definition, the relationship between the BG and the PHS variables are the same, i.e. the inputs and outputs of the storage elements in integral causality are, respectively, the time derivatives $\dot{p}$ and $\dot{q}$ of the state vectors and the components of the gradient of $\mathcal{H}(p, q)$.

Here an explicit PHS is obtained from the BG of Figure 3 considering the *IC-Field* model of Figure 4. Reading through the causal paths on Figure 3, the relationship between the state vector and the gradient components can be obtained. From these relationships the skew symmetric structure matrix is computed. Then, reading through the causal paths from the sources to the *IC-Field*, the matrix that weights the inputs is obtained. Altogether this results in the PHS expressed in (17).

$$\begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & \mathrm{I}_{5x5} \\ -\mathrm{I}_{5x5} & 0 \end{pmatrix} \nabla_x \mathcal{H} + \begin{pmatrix} 0 \\ W(q) \end{pmatrix} T \tag{17}$$

5. SIMULATIONS RESULTS

All the models derived from the base BG model are well-suited for simulation and control system design (at least to tune the control law via simulation experiments), but not all of them are good for model-based controller synthesis. Here, the distinction between control system synthesis, referring to the derivation of the control law, and control system design, denoting its parameterization, has been made, following Wonham (Wonham, 1979). It is not the purpose of this paper to provide new control laws nor a new method to obtain them, but just to point out the suitability of the previous models to these aims. Tuning controllers with the help of simulation is a typical method used, for instance, to adjust the gains of a PD controller taking care of point-to-point missions. On the other hand, only the BG with the IC-field and, of course, the EL and PHS models, are suitable for control system synthesis, as this needs a proper system model. Indeed, the BG with the residual sink is not, as it shows all the energy variables in integral causality, thus having an artificially created excess of state variables. Nor is suitable for this purpose the BG with the parasitic spring-damper components, because it contains the spurious states associated to the spring energy variables. Readers interested in methods for control system synthesis and design of robotic manipulators affine to the modeling techniques considered in this paper, could refer to (Siciliano et al., 2009) and (Ortega et al., 2013) for classic and energy-based methods, and to (Merzouki et al., 2012) for methods in the BG-domain.

The model parameters of the links are: $m_1 = 0.5Kg$, $l_1 = 0.5m$, $I_1 = 0.0015Kgm^2$, $m_2 = 0.5Kg$, $l_2 = 0.25m$, $I_2 = 0.0012Kgm^2$. The simulation parameters of the MB are: $I_{\mathrm{w}1} = 0.1\ Kgm^2$, $I_{\mathrm{w}2} = 0.1Kgm^2$, $I_{\mathrm{w}3} = 0.1Kgm^2$, $r = 0.05m$, $l_b = 0.15m$ and $L = 0.2m$.

All the models were simulated in 20sim® (Controllab Products, n.d.), in closed loop with the same control law, a feedback-linearization based tracking controller, see (Siciliano et al., 2009) for a detailed description of the control scheme (resulting from an EL-model based design). The Modified Backward Differentiation Formula (MDF) has been selected as the integration method, with a tolerance for relative and absolute errors of 1e-5.

In this section two simulation scenarios are addressed. In the first scenario the desired end effector trajectory is a circumference of radius $R = l_1 + l_2$ with center at the origin of link 1. As the PMRM is redundant, infinite configurations of vector q can realize the desired trajectory. In this particular scenario, the end effector trajectory is achieved by moving only link 1 while the other coordinates remain constant. The performance of the model is analyzed taking into account the simulation time and the integral of the error e defined in (18), where (x, y) and (x_{ref}, y_{ref}) are the end effector position and desired end effector position respectively.

$$e = \sqrt{\left(x - x_{ref}\right)^2 + \left(y - y_{ref}\right)^2} \tag{18}$$

All the four BGs previously presented have been simulated in this scenario. Table 1 presents the results. The three first simulations present a similar behavior regarding the errors. Even though their values are quite similar, the simulation time of the BG model with derivative causality is higher. This behavior suggests that the numerical solution of BG with derivative causalities takes less time if ZCPs are broken adding residual sink efforts or parasitic components. However, in the latter case, as parasitic components are a non-ideal implementation of residual sinks, there will be a dependency of the simulation errors on the parameter tuning criteria. Finally, as it can be seen, the *IC-Field* method presents the smallest error, a consequence of the exact cancellation by the controller of nonlinearities in the *IC-Field* BG. But this advantage has a negative counterpart: implementing the *IC-Field* implies inverting the system inertia matrix in the BG model (precisely the nonlinearity cancelled by the controller), the cause of the increase in simulation time with respect to the two previous models. In this simple system this increase is not an issue, but may noticeably slow down the simulation in more complex systems.

Table 1: Simulation times and errors.

BG Simulations models	**Simulation Time [s]**	**Error**
BG with derivative causality	2.302	0.1838
BG with rS (*constraint*)	0.495	0.1839
BG with parasitic components	0.453	0.1861
BG with *IC-Field*	0.732	0.0021

The evolution of the positions in the simulation of the *IC-Field* model are shown in Figure 6. Starting from the initial condition $q_0 = [4.35,0,0,0,0]^T$ and zero velocities, the circular trajectory is followed. Figure 7 depicts the error (18).

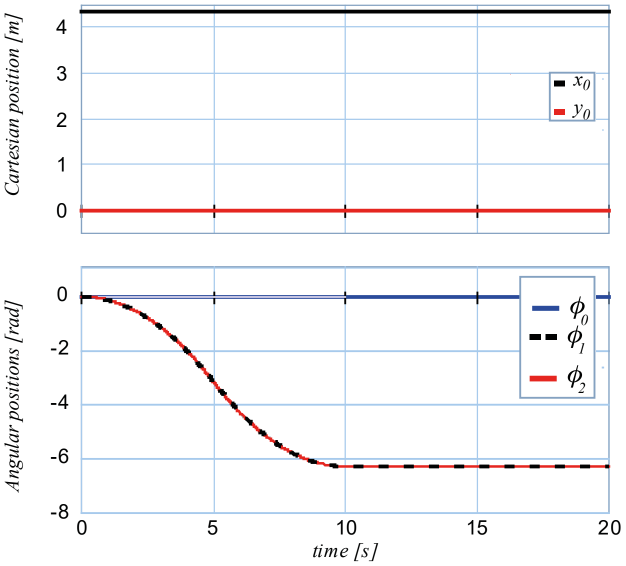


Figure 6: Evolution of the positions in the BG with *IC-Field*.

The second scenario has two stages: in the first, the MB moves from the initial condition $q_0 = [0,0,0,0,0]^T$ to the point $q = [4.35,0,0,0,0]^T$. In the second, once that point reached, the end effector describes a circumference of radio $R = l_1 + l_2$ by moving only link 1. The evolutions of all the generalized positions are depicted in Figure 8. Figure 9 shows the trajectory performed by both the end effector and the tip of link 1 for this scenario.

It is concluded that all the BG models are useful for the numerical test of the closed-loop performance.

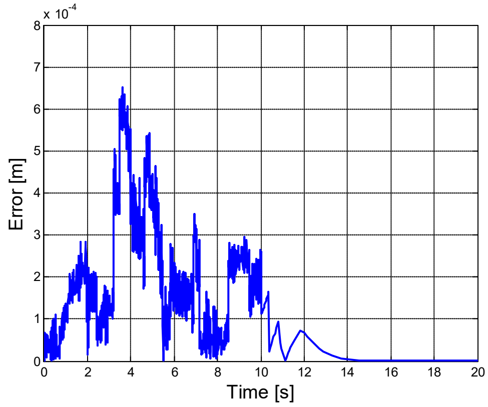


Figure 7: Evolution of the error in the *IC-Field* BG model.

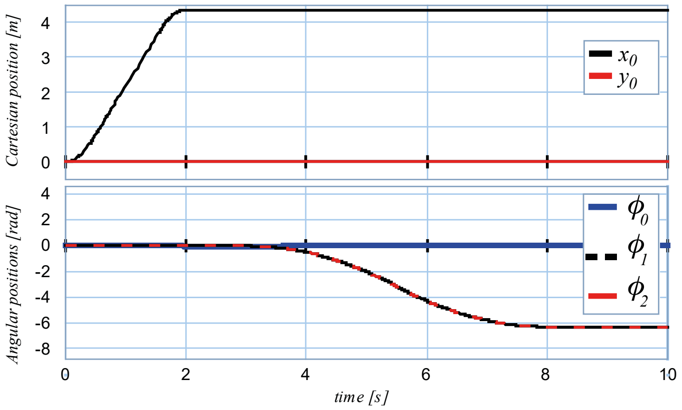


Figure 8: Evolution of the generalized positions in the BG with *IC-Field* in scenario 2.

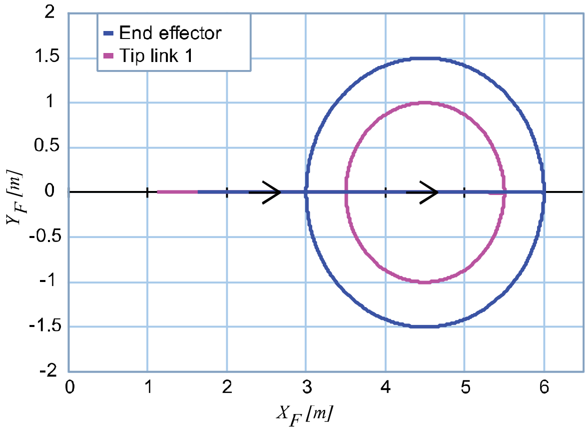


Figure 9: End efector and tip of link 1 in workspace.

6. CONCLUSIONS

In this paper a BG model of a planar mobile robotic manipulator was obtained with the standard BG-modeling procedure. This model presents nonlinear state-dependencies among storage elements that could be inconvenient for, or not manageable by numerical solvers of some simulation tools. As solutions to this problem, three modifications of the base BG model have been presented. These modifications consist in breaking the ZCP by adding parasitic elements or, alternatively, rS, and introducing storage fields with energy variables without constraints. Also, the inherent properties of BGs have been exploited to derive from the base BG Euler-Lagrange and Port-Hamiltonian models, the two main energy-based modeling formalisms used for control system design. Even though the main objective of this work was not the design of the controller for the planar mobile robotic manipulator, a model based control technique has been implemented in order to perform simulation allowing to validate the BG models.

ACKNOWLEDGMENTS

The authors wish to thank SeCyT-UNR for the support to this research through the financing of PID-UNR_ING502, and MinCyT (Argentina) and DST (India) for the financing of project IN/14/07 in the framework of the bilateral cooperation between Argentina and India.

APPENDIX: RESIDUAL SINKS

A residual sink element, graphically represented in Figure 10, can be interpreted as an energy-storing device whose parameter tends to zero. For example, an effort residual sink can be interpreted a C element in integral causality:

$$C\dot{e} = \Delta f \qquad (19)$$

If the parameter C tends to zero, then $\dot{e}$ is determined by the algebraic equation $\Delta f = 0$.

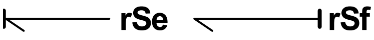


Figure 10: Effort and flow residual sinks.

REFERENCES

Borutzky, W. and Cellier, F.E. (1996) 'Tearing Algebraic Loops in Bond Graphs', *Trans. Soc. Comput. Simul. Int.*, vol. 13, #dec#, pp. 102-115, Available: ISSN: 0740-6797.

Cacho, R., Felez, J. and Vera, C. (2000) 'Deriving simulation models from bond graphs with algebraic loops.', *Journal of the Franklin Institute*, vol. 337, pp. 579-600, Available: ISSN: 0016-0032.

Controllab Products, B.V. (n.d) *20-sim. www.20sim.com*, Available: http://www.20sim.com.

Donaire, A. and Junco, S. (2009) 'Derivation of Input-State-Output Port-Hamiltonian Systems from bond graphs', *Simulation Modelling Practice and Theory*, vol. 17, pp. 137-151, Available: ISSN: 1569-190X.

Gawthrop, P.J. and Smith, L. (1992) 'Causal augmentation of bond graphs with algebraic loops', *Journal of the Franklin Institute*, vol. 329, pp. 291-303, Available: ISSN: 0016-0032.

Karnopp, D. (1977) 'Lagrange's Equations for Complex Bond Graph Systems.', *ASME. J. Dyn. Sys., Meas., Control.*, vol. 99, pp. 300-306.

Karnopp, D. (1992) 'An approach to derivative causality in bond graph models of mechanical systems', *Journal of the Franklin Institute*, vol. 329, pp. 65-75, Available: ISSN: 0016-0032.

Karnopp, D. and Margolis, D. (1979) 'Analysis and Simulation of Planar Mechanism Systems Using Bond Graphs',. *ASME. J. Mech. Des.*, vol. 101, pp. 187-191.

Karnopp, D., Margolis, D. and Rosenberg, R. (2006) *System Dynamics: Modeling and Simulation of Mechatronic Systems*, New York, NY, USA: John Wiley & Sons, Inc.

Merzouki, R., Samantaray, A.K., Pathak, P.M. and Bouamama, B.O. (2012) *Intelligent Mechatronic Systems: Modeling, Control and Diagnosis*, Springer Publishing Company, Incorporated.

Nacusse, M.A. and Junco, S.J. (2017) 'Controlled Switched Structures for Bond-Graph Modelling and Simulation of Hybrid Systems', in Borutzky, W. (ed.) *Bond Graphs for Modelling, Control and Fault Diagnosis of Engineering Systems*, Cham: Springer International Publishing.

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

Ortega, R., Schaft, A.J.V.D., Mareels, I. and Maschke, B. (2001) 'Putting energy back in control', *IEEE Control Systems*, vol. 21, Apr, pp. 18-33, Available: ISSN: 1066-033X.

Ortega, R., Spong, M.W., Gomez-Estern, F. and Blankenstein, G. (2002) 'Stabilization of a class of underactuated mechanical systems via interconnection and damping assignment', *IEEE Transactions on Automatic Control*, vol. 47, no. 8, Aug, pp. 1218-1233, Available: ISSN: 0018-9286 DOI: 10.1109/TAC.2002.800770.

Rideout, G. and Stein, J.L. (2003) 'An Energy-Based Approach to Parameterizing Parasitic Elements for Eliminating Derivative Causality', Proc. ICBGM'03, the Int. Conf. on Bond Graph Modeling and Simulation, Orlando, 121–128.

Siciliano, B., Sciavicco, L., Villani, L. and Oriolo, G. (2009) *Robotics: Modelling, Planning and Control*, Springer.

van der Schaft, A. and Jeltsema, D. (2014) 'Port-Hamiltonian Systems Theory: An Introductory Overview', *Found. Trends Syst. Control*, vol. 1, #jun#, pp. 173-378, Available: ISSN: 2325-6818.

van Dijk, J. and Breedveld, P.C. (1991) 'Simulation of system models containing zero-order causal paths—I. Classification of zero-order causal paths', *Journal of the Franklin Institute*, vol. 328, pp. 959-979, Available: ISSN: 0016-0032.

van Dijk, J. and Breedveld, P.C. (1991) 'Simulation of system models containing zero-order causal paths—II Numerical implications of class 1 zero-order causal paths', *Journal of the Franklin Institute*, vol. 328, pp. 981-1004, Available: ISSN: 0016-0032.

Wonham, W.M. (1979) *Linear Multivariable Control: A Geometric Approach*, 2nd edition, Springer-Verlag.

AUTHORS' INDEX