THE 12TH INTERNATIONAL CONFERENCE ON INTEGRATED MODELING AND ANALYSIS IN APPLIED CONTROL AND AUTOMATION

SEPTEMBER 18 – 20, 2019
LISBON, PORTUGAL

IMAACA

EDITED BY
AGOSTINO BRUZZONE
GENEVIÈVE DAUPHIN-TANGUY
SERGIO JUNCO

PRINTED IN RENDE (CS), ITALY, SEPTEMBER 2019

ISBN 978-88-85741-32-4 (Paperback)
ISBN 978-88-85741-31-7 (PDF)

© 2019 DIME UNIVERSITÀ DI GENOVA, DIMEG UNIVERSITY OF CALABRIA

RESPONSIBILITY FOR THE ACCURACY OF ALL STATEMENTS IN EACH PAPER RESTS SOLELY WITH THE AUTHOR(S). STATEMENTS ARE NOT NECESSARILY REPRESENTATIVE OF NOR ENDORSED BY THE DIME, UNIVERSITY OF GENOA OR DIMEG UNIVERSITY OF CALABRIA. PERMISSION IS GRANTED TO PHOTOCOPY PORTIONS OF THE PUBLICATION FOR PERSONAL USE AND FOR THE USE OF STUDENTS PROVIDING CREDIT IS GIVEN TO THE CONFERENCES AND PUBLICATION. PERMISSION DOES NOT EXTEND TO OTHER TYPES OF REPRODUCTION NOR TO COPYING FOR INCORPORATION INTO COMMERCIAL ADVERTISING NOR FOR ANY OTHER PROFIT – MAKING PURPOSE. OTHER PUBLICATIONS ARE ENCOURAGED TO INCLUDE 300 TO 500 WORD ABSTRACTS OR EXCERPTS FROM ANY PAPER CONTAINED IN THIS BOOK, PROVIDED CREDITS ARE GIVEN TO THE AUTHOR(S) AND THE CONFERENCE.

FOR PERMISSION TO PUBLISH A COMPLETE PAPER WRITE TO: DIME UNIVERSITY OF GENOA, PROF. AGOSTINO G. BRUZZONE, VIA OPERA PIA 15, 16145 GENOVA, ITALY OR TO DIMEG UNIVERSITY OF CALABRIA, PROF. FRANCESCO LONGO, VIA P.BUCCI 45C, 87036 RENDE, ITALY. ADDITIONAL COPIES OF THE PROCEEDINGS OF THE EMSS ARE AVAILABLE FROM DIME UNIVERSITY OF GENOA, PROF. AGOSTINO G. BRUZZONE, VIA OPERA PIA 15, 16145 GENOVA, ITALY OR FROM DIMEG UNIVERSITY OF CALABRIA, PROF. FRANCESCO LONGO, VIA P.BUCCI 45C, 87036 RENDE, ITALY.

ISBN 978-88-85741-32-4 (Paperback)
ISBN 978-88-85741-31-7 (PDF)

The 12th International Conference On Integrated Modeling And Analysis In Applied Control And Automation, IMAACA 2019

September 18 - 20 2019

Lisbon, Portugal

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

Hungarian Academy Of Sciences Centre For Energy Research

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

University of Minho

I3M 2019 Industrial Sponsors

CAL-TEK SRL

LIOTECH LTD

MAST SRL

SIM-4-FUTURE

I3M 2019 Media Partners

Inderscience Publishers – International Journal of Simulation and Process Modeling

Inderscience Publishers – International Journal of Service and Computing Oriented Manufacturing

IGI GLOBAL – International Journal of Privacy and Health Information Management (IJPHIM)

Halldale Media Group: The Military Simulation and Training Magazine

Halldale Media Group: The Journal for Healthcare Education, Simulation and Training

SAGE
Simulation Transaction of SCS

De Gruyter
International Journal of Food Engineering

MDPI - Sustainability

Euromerci: The Italian Monthly Logistics Journal

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 2019

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 12TH INTERNATIONAL CONFERENCE ON INTEGRATED MODELING AND ANALYSIS IN APPLIED CONTROL AND AUTOMATION, IMAACA 2019

GENERAL CO-CHAIRS

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

PROGRAM CHAIR

LOUCAS LOUCA, *UNIVERSITY OF CYPRUS, CYPRUS*

IMAACA 2019 International Program Committee

Jorge Baliño, Univ. of São Paulo, Brazil
Wolfgang Borutzky, BRS-Univ. Applied Sciences, Sankt Augustin, Germany
Agostino G. Bruzzone, University of Genoa, Italy
Nicolaï Christov, LAGIS, UST Lille, France
Mauro Carignano, UNR, Rosario, Argentina
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
Agenor Fleury, University of São Paulo, Brazil
Laurentiu Hetel, CNRS-EC-Lille, France
Hussein Ibrahim, ITMI, Québec, Canada
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, CAL-TEK, Italy
Norberto Nigro, CIMEC, UNL, S. Fe, Argentina
Rachid Outbib, LSIS, Marseille, France
Pushparaj Mani Pathak, IIT, Roorkee, India
Ricardo Pérez Correa, PUC, Chile
Xavier Roboam, INP Toulouse, France
Mónica Romero, FCEIA, UNR, Rosario, Argentina
Christophe Sueur, Centrale Lille, France
Armand Toguyeni, Centrale Lille, France
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

IMAACA 2019 - WELCOME MESSAGE

On behalf of the International Program Committee we are glad to welcome all the authors, delegates and attendees to IMAACA 2019, the 12th International Conference on Integrated Modeling and Analysis in Applied Control and Automation.

We keep enjoying our continuous affiliation to the I3M community, the International Multidisciplinary Modeling and Simulation Multiconference, which this year celebrates its 16th 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 Lisbon.

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.

Last but not least, our deepest thanks go to all the authors for their contributions and attendance to the conference. Their submissions to IMAACA 2019 constitute a valuable bend of application relevant solutions to control and automation problems, emphasizing the role of model analysis as integral part of the complete design of the automated system.

Finally, let us wish you a fully enjoyable conference in both its technical and social aspects.

Sergio Junco and Geneviève Dauphin-Tanguy

Sergio Junco
Universidad Nacional
de Rosario, Argentina

Geneviève
Dauphin-Tanguy
Ecole Centrale de
Lille
France

ACKNOWLEDGEMENTS

The IMAACA 2019 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 2019 IPC would like to thank all the authors as well as the reviewers for their invaluable work.
A special thank goes to all the organizations, institutions and societies that have supported and technically sponsored the event.

I3M 2019 INTERNAL STAFF

AGOSTINO G. BRUZZONE, *DIME, UNIVERSITY OF GENOA, ITALY*
ALESSANDRO CHIURCO, *CAL-TEK SRL, ITALY*
JESSICA FRANGELLA, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
CATERINA FUSTO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
LUCIA GAZZANEO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
FRANCESCO LONGO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
MARINA MASSEI, *DIME, UNIVERSITY OF GENOA, ITALY*
LETIZIA NICOLETTI, *CAL-TEK SRL, ITALY*
ANTONIO PADOVANO, *DIMEG, UNIVERSITY OF CALABRIA, ITALY*
CATALDO RUSSO, *CAL-TEK SRL, ITALY*
KIRILL SINELSHCHIKOV, *SIMULATION TEAM, ITALY*
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

A combined bond graph-based – data-based approach to failure prognosis 1
W. Borutzky

Failure prognosis of embedded systems based on temperature drift assessment 11
O. Djedidi, M. A. Djeziri, S. Benmoussa

Mechanical verification of dynamic musculoskeletal model with Muscle – Tendon Complex at jumping motion by bond graph 17
K. Suzuki, Y. Oida, T. Fukuoka, Y. Ito, T. Nemoto

Detection of stator winding inter-turn short circuit fault in induction motor using LS-SVM 24
M. Birame, S. Ahmed Bessedik, A. Naamane

Time series model improving with automatic Savitzky - Golay filter for Remaining Useful Life estimation 32
Y. Diaf, S. Benmoussa, M. Djeziri

Bond Graphs based formation control of holonomic robots 38
M. Nacusse, M. Crespo, S. Junco

Full modelling and sliding mode control for a quadrotor UAV in visual servoing task 48
C. Bensalah, N. K. M'Sirdi, A. Naamane

Path tracking for the convoy of autonomous vehicles based on a non-linear predictive control 58
M. M. M. Ahmed, A. Naamane, N. K. M'Sirdi

Defender-attacker-target game: first-order defender and attacker dynamics 65
V. Turetsky, V. Y. Glizer

Automation and cognition: a methodological process for human-centred design to minimize errors 73
J.-M. Mercantini

A new proposed shepherd model of a Li-ion open circuit battery based on data fitting 83
H. Hemi, N. K M'Sirdi, A. Naamane

Adaptive estimation of the thermal behavior of CPU-GPU SoCs for prediction and diagnosis 93
O. Djedidi, N. K. M'Sirdi, A. Naamane

An overview of State of Charge(SOC) and State of Health(SOH) estimation methods of Li-ion batteries 99
K. Saqli, H. Bouchareb, M. Oudghiri, K. N. M'Sirdi

Component approach based on Petri Nets for the design of the automatic control for railway system 105
A. Toguyéni

IDA-PBC controller for dc-dc power electronic converters with nonlinear load enhanced with additive disturbance estimation 115
J. Tomassini, S. Junco, A. Donaire

Comparison of trajectory tracking and obstacle avoidance strategies for a multi-agent dynamic system 124
M. Crespo, M. Nacusse, S. Junco

Author's Index 134

A COMBINED BOND GRAPH-BASED – DATA-BASED APPROACH TO FAILURE PROGNOSIS

W. Borutzky

Bonn-Rhein-Sieg University of Applied Sciences, St. Augustin, Germany

ABSTRACT

Given known control inputs and real sensor outputs or simulated measurements, the paper shows that numerical values of unknown parameter degradation functions can be obtained by evaluating equations derived from a bicausal diagnostic bond graph that are not analytical redundancy relations. Inspection of causal paths beforehand enables to decide whether potential parametric faults can be isolated with a number of sensors in given locations. The proposed approach can be applied in the case of multiple isolated simultaneous parametric faults. Numerical values of degradation functions can be computed concurrently to the constant monitoring of a system and the measurement of signals. Repeatedly projecting the time evolution of a degradation function into the future based on values in a sliding time window enable to obtain a sequence of remaining useful life estimates. The novel proposed combined bond graph-model-based, data-based approach is verified by an offline simulation study of a typical power electronic circuit.

Keywords: Sensor placement and fault isolation, a priori unknown parameter degradation functions, bicausal Bond Graphs, failure prognosis, remaining useful life.

1. INTRODUCTION

Nowadays, more and more mechatronic engineering systems are equipped with sensors and embedded systems so that they can process measured information, detect and isolate emerging faults and may reconfigure their control themselves. Beyond safety and reliability of safety critical engineering systems and processes, these capabilities are of significant importance for supervision, automation and condition based maintenance of industrial processes, for an intelligent communication and cooperation of networked robots, or for all kinds of emerging autonomous intelligent operating mobile systems such as unmanned aerial vehicles, or for cyber physical systems.

Accordingly, fault detection and isolation (FDI) has been a major subject in research and in various application fields. Approaches to FDI are commonly based on either measured data or on models In addition, recently also a combination of model-based and data-based approaches has been proposed Jha (2015).

With regard to fault isolation a question is how many sensors are to be placed in which locations in order to isolate a maximum of potentially faulty system components. This is still a subject of ongoing research. Various approaches to the sensor placement problem based on bipartite graphs Frisk et al (2009), on digraphs Alem and Benazzouz (2013), or on bond graphs Djeziri et al (2009); Benmoussa et al (2014); Chi and Wang (2015); Borutzky (2018b) have been reported in the literature.

Once, a fault has been diagnosed, another question is how long a system may safely continue its operation despite the presence of an incipient fault before the increasing affect of the fault on the dynamic system behaviour may lead to a component or even a system failure. Clearly, constant monitoring of the health of a system and a repeated prediction of the remaining useful life (RUL), i.e. failure prognosis is of technical and economical importance. Online failure prognosis is also a subject of ongoing research.

To anticipate the RUL as of the current time instant, it is necessary to know the degradation behaviour of a fault over time. To that end one may try to develop a model of the degradation process starting from physical principles. Difficulties, however, may be that the degradation process is not fully understood or that not all needed parameters of a degradation model can be determined.

Other options may be to obtain a degradation model from offline accelerated life tests Escobar and Meeker (2006) and to use the results in online health monitoring for the prediction of the RUL Medjaher, K. and Zerhouni, N. (2013), or to assume that a potential degradation function candidate is a member of a certain class of functions and to adapt the unknown parameters of the function by curve fitting. As measured signals carry noise, a RUL has to be considered a stochastic quantity Jha (2015).

Moreover, for systems operating in various modes, the degradation behaviour may change from mode to mode making it necessary to change to another class of potential degradation functions (Borutzky, 2016, Chap. 6).

In order to avoid the disadvantages pertaining to the development of a degradation model from physical principles and as well to curve fitting based on measurements, Borutzky recently proposed to estimate the numerical values of an unknown degradation function from the time series of analytical redundancy relations (ARRs) derived from a diagnostic bond graph (DBG) Borutzky (2018a). The approach uses a first stage and a second stage DBG. An evaluation of ARRs obtained from a DBG with nominal parameters and inputs from the real faulty system or a model of it must result in some residuals significantly different from zero. A second stage DBG accounts for the unknown degradation function of a component parameter. Accordingly, ARRs derived from the second stage DBG must be close to zero. The use of ARR residuals from the first stage DBG in the ARRs of the second stage DBG leads to an equation that determines the unknown degradation function.

A different approach recently reported in Prakash et al (2018) also evaluates ARRs but obtains degradation data by repeated updating a faulty parameter. Once a faulty parameter is identified, its value replaces the current parameter value which means that the model is adapted to the current faulty situation. Accordingly, an evaluation of the ARR sensitive to fault under consideration should provide a residual close to zero. However, as the magnitude of the fault progressively increases with time, a repeated evaluation of the ARR after some time step will result in new residual. Identification of the new faulty parameter value gives another estimated value of the unknown degradation function.

This paper continues the work in Borutzky (2018a) by showing that a more direct approach based on a *single bicausal* DBG can provide the same results and does not need ARRs. Determining the numerical values of a degradation function means to evaluate the magnitude of a fault for each time instant. Clearly, to that end, the faulty component must have been isolated. Therefore, in the following, first, the placement of sensors aiming at a fault isolation is addressed. It is assumed that the observed faulty system behaviour is due to a component parameter that has become time-varying as of some time instant.

Given known system inputs and either measurements from a real system or simulated measurements, the objective of the single bicausal DBG based approach proposed in this paper is to numerically determine an unknown parameter degradation function by estimating the numerical values of a faulty parameter for each time instant that can be used in data-based failure prognosis. As in Prakash et al (2018), the numerical values of an unknown degradation function are determined over a sliding time window of fixed size. For each time window, the data-based part of the approach identifies a mathematical function and projects it into the future to obtain an estimate of the current RUL. As a result, a sequence of RUL estimates is obtained with values that tend to zero as the considered faulty component reaches its end of life (EOL).

The approach proposed in this paper is explained by means of a typical power electronic circuit. Results are verified by an offline simulation study. The presentation continues by addressing the estimation of the RUL and concludes with a discussion of some aspects that may be subject of further considerations.

2. SENSOR PLACEMENT AND FAULT ISOLATION

In Borutzky (2018b), Borutzky proposes a graphical approach to the isolation of parametric component faults that aims at avoiding the limitations of an inspection of a structural fault signature matrix (FSM) and the computational costs of numerical methods. Briefly, the idea is to start from a DBG of a system with given sensors, consider causal paths and to add repeatedly detectors to the DBG so that a maximum of disjoint causal paths from detectors to possibly faulty elements is obtained. If are there causal paths from different detectors to a potentially faulty element, the set of these non-disjoint causal paths must be unique. A parameter fault in these elements can be isolated as can be verified by a FSM. As it is known, a structural FSM can be directly obtained from a DBG by following causal paths. As there may be technical limitations as to where sensors can be placed, a detector may not be attached to some junctions in the DBG. Moreover, some parameter faults cannot be isolated without inserting additional junctions and attaching a detector to it. For instance, a flow sensor is not enough to isolate the parameters of electrical elements connected in series.

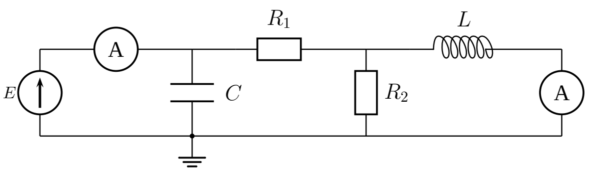


Figure 1: Circuit schematic

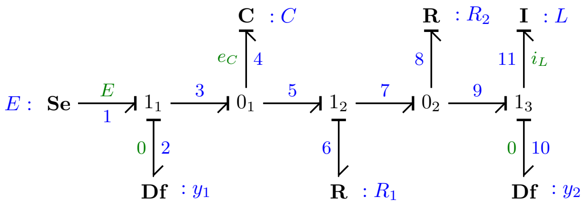


Figure 2: BG of the circuit in Fig. 1

The issue of fault isolation and sensor placement is illustrated by means of a small electrical circuit that has also been considered by Frisk et. al. in Frisk et al (2009).

The circuit schematic with two non-faulty flow sensors in Fig. 1 is easily converted into the BG in Fig. 2. The model is of order one as the capacitor must take derivative causality. As there is a causal path from the inductor $\mathrm{I} : L$ to the detector $\mathrm{Df} : y_2$ and since no storage element remains in integral causality when preferred derivative causality is assigned, the circuit is completely state observable with the sensor $\mathrm{Df} : y_2$. However, as to FDI, even if the voltage source $\mathrm{Se} : E$ and the flow sensor $\mathrm{Df} : y_2$ are assumed to be faultless, one sensor is certainly not enough to isolate more than one parametric fault. That is, additional real or virtual sensors are needed. For the two output variables, y_1, y_2, the following two equations can be derived from the BG in Fig. 2.

$$y_1 = -\frac{R_2}{R_1+R_2} i_L + \frac{1}{R_1+R_2} E + C\frac{d}{dt}E \qquad (1)$$

$$y_2 = i_L \qquad (2)$$

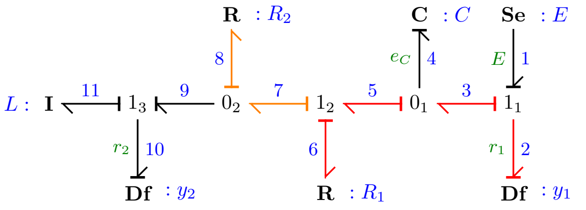


Figure 3: DBG of the circuit in Fig. 1

Table 1: Structural fault signature matrix of the circuit in Fig. 1 with two current sensors

Element	r_1'	r_2'	D_b	I_b
Se : E	1	0	1	0
C : C	1	1	1	0
I : L	0	1	1	①
R : R_1	1	0	1	0
R : R_2	1	1	1	0

Given the input E and the state variable i_L, (1) may be considered the equation of a virtual sensor that provides the output y_1. The question is, how many sensors are needed to isolate a maximum of number of potentially faulty element parameters.

Now, for FDI, the circuit schematic in Fig. 1 is transformed into the diagnostic bond graph (DBG) in Fig. 3 with detectors in inverted causality and storage elements in preferred derivative causality. As can be seen, there are two non-disjoint causal paths from detector Df : y_1 to resistors R : R_1 and R : R_2. That is, these parameters cannot be isolated. If one of the two resistors becomes faulty, the degradation of its resistance cannot be computed given the two sensors.

As there are two sensors in the circuit, two ARRs can be set up from the DBG in Fig. 3.

$$\begin{aligned}
\mathrm{ARR}_1: \quad r_1' &:= r_1 + (R_1 + R_2)C\dot{r}_1 \\
&= E + (R_1 + R_2)C\dot{E} - (R_1 + R_2)y_1 + R_2 y_2 \quad (3) \\
\mathrm{ARR}_1: \quad r_2' &:= r_2 - R_2 C\dot{r}_1 \\
&= R_2(y_1 - y_2) - R_2 C\dot{E} - L\dot{y}_2 \quad (4)
\end{aligned}$$

Accordingly, the structural FSM in Table 1 displays which element parameters contribute to which ARR. As the entry '1' highlighted in blue in the last column of Table 1 indicates, only a potentially faulty inductor I : L can be structurally isolated given the two flow sensors.

Following the graphical procedure proposed in Borutzky (2018b), attaching an additional flow detector Df : y_3 to junction 1_2 (Fig. 4) yields disjoint causal paths from detectors to elements except the two causal paths from detectors Df : y_2 and Df : y_3 to the resistor R : R_2. They indicate that R_2 contributes to the two ARR residuals r_2, r_3 giving rise to an element fault signature that is unique. As a result, all possibly faulty elements can be isolated given these three sensors[1]. The result of this bond graph based procedure is confirmed by the structural FSM in Table 2 and is in accordance with the result obtained by the bipartite approach in Frisk et al (2009).

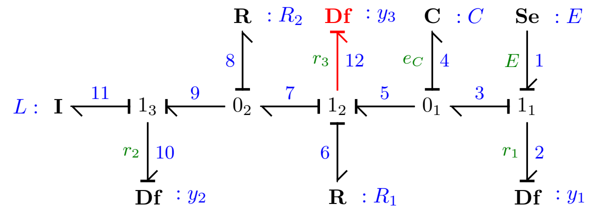


Figure 4: DBG of the circuit in Fig. 1 with an additional flow detector Df : y_3

Table 2: Structural fault signature matrix of the DBG in Fig. 3 with detectors Df : y_1, Df : y_2 and Df : y_3.

Element	r_1	r_2	r_3'	D_b	I_b
Se : E	1	0	0	1	1
C : C	1	0	1	1	1
I : L	0	1	0	1	1
R : R_1	0	0	1	1	1
R : R_2	0	1	1	1	1

Depending on the structure of a DBG, attaching additional sensors to its junctions in certain places can not always increase the number of disjoint causal paths from detectors to potentially faulty elements and thus increase the number of parametric faults that can be structurally isolated. As an example, consider the DBG model of a DC motor that moves a rotational mechanical load against an external torque T displayed in Fig. 5.

With the given flow detectors Df : i_a for the armature current and Df : ω_l for the angular load velocity none of the potentially faulty parameters can be isolated. The result is only partially improved by adding a flow detector Df : ω_m to junction 1_2 for the angular velocity of the motor connected to the load by a shaft with some flexibility C : C_s. The added flow detector enforces integral causality at the C-element C : C_s so that the ARR obtained by the sum of flows at junction 1_2 must be differentiated to get rid of the initial condition of the integration. Nevertheless, only two potentially faulty parameters can be isolated as indicated in the FSM of Table 3

As the voltage supply of the motor, the resistance R_a and the inductance L_a of the armature are in series, one flow detector for the current through these elements is not enough to isolate their parameters. Additional junctions $(0_2, 0_3)$ with detectors attached to them must be inserted into the DBG as shown in Fig. 6

So far, sensors have been assumed to be faultless. If this is not the case, a faulty sensor can be modelled by a

---

[1] Note, as Df : y_3 entails integral causality on C : C the ARR for r_3 is differentiated with respect to time to get rid of the initial condition.

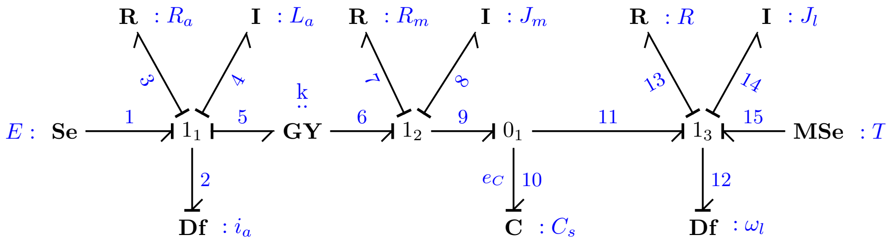


Figure 5: DBG of a DC motor drive with two flow sensors Df : i_a and Df : ω_l (Borutzky (2018b))

Table 3: FSM of the DBG in Fig. 5 with an additional flow detector Df : ω_m

Element	r_1	r_2	r_3	D_b	I_b
Se : E	1	0	0	1	0
MSe : T	0	1	0	1	0
R : R_a	1	0	0	1	0
I : L_a	1	0	0	1	0
GY : k	1	0	1	1	①
R : R_m	0	0	0	1	0
I : J_m	0	0	0	1	0
C : C_s	0	1	1	1	①
R : R	0	1	0	1	0
I : J_l	0	1	0	1	0

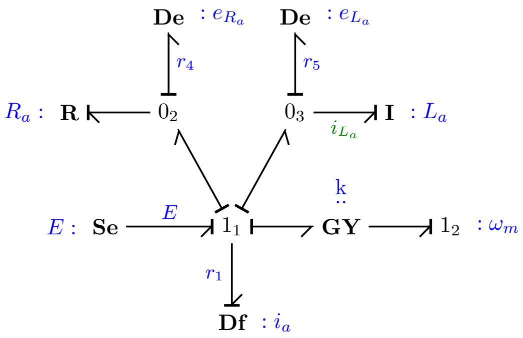


Figure 6: DBG fragment with inserted junctions 0_2 and 0_3 and attached effort detectors for further isolation of element parameters

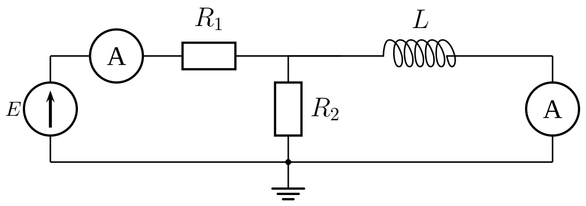


Figure 7: Circuit with a faulty sensor for the inductor current

detector of the faultless signal and a modulated sink that provides the faulty signal component. Clearly, a faulty sensor cannot be used for detection and isolation of parametric faults of system components. Therefore, another non-faulty sensor is needed that enables to isolate the sensor fault. For illustration, consider the small passive network in Fig. 7 with a faulty sensor for the inductor current. The DBG in Fig. 8 reflects that the sensor measures a faulty inductor current $\tilde{i}_L = i_L + \Delta i_L$. As can be seen, there is a direct causal path p_1 from the additional detector De : u_s to the modulated sink MSf : Δi_L and another path p_2 to the inductor I : L which means that the sensor fault Δi_L affects ARR residuals r_2 and r_3. Moreover, there is an indirect causal path p_3 from detector Df : y_1 via R : R_2 to the sink MSf : Δi_L. That is, Δi_L also contributes to ARR$_1$, hence, all three ARRs.

Moreover, there are the following direct causal paths

p_4: Df : $y_1 \rightarrow 1_1 \rightarrow Se : E$
p_5: Df : $y_1 \rightarrow 1_1 \rightarrow 0_1 \rightarrow R : R_2$
p_6: Df : $i_L \rightarrow 0_2 \rightarrow 1_2 \rightarrow 0_1 \rightarrow R : R_2$

The last two paths indicate that R_2 contributes to residuals r_1 and r_2. As a result, the sensor fault Δi_L can be structurally isolated. This can be verified by reading out the following ARRs from the DBG in Fig. 8 and by capturing their structural parameter dependencies in the FSM in Table 4.

$$\text{ARR}_1: \quad r_1 = E - R_2(y_1 - (i_L + \Delta i_L)) - R_1 y_1 \tag{5}$$

$$\text{ARR}_2: \quad r_2 = R_2(y_1 - (i_L + \Delta i_L)) - u_s \tag{6}$$

$$\text{ARR}_3: \quad \dot{r}_3 = \frac{d}{dt} i_L + \frac{d}{dt} \Delta i_L - \frac{1}{L} u_s \tag{7}$$

Potentially faulty elements Se : E and R : R_1 cannot be isolated. There are direct causal paths to these elements from detector Df : y_1 but no causal paths from the other detectors.

3. NUMERICAL DETERMINATION OF UNKNOWN DEGRADATION FUNCTIONS

Parametric degradation means that performance degradation of an engineering system is due to the fact that some of its parameters increasingly deviate from their nominal values with time following a function of which an exact analytical expression is mostly unknown.

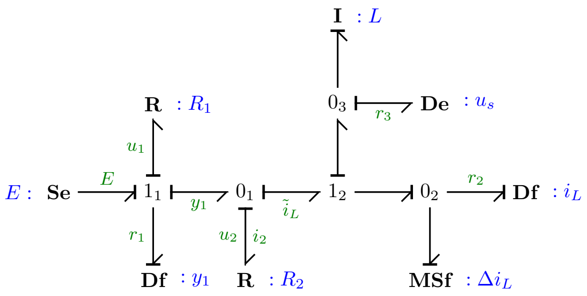


Figure 8: DBG of the circuit in Fig. 7

Table 4: Fault signature matrix of the DBG in Fig. 8

Element	r_1	r_2	r_3'	D$_b$	I$_b$
Se : E	1	0	0	1	0
R : R_1	1	0	0	1	0
R : R_2	1	1	0	1	1
I : L	0	0	1	1	1
MSf : Δi_L	1	1	1	1	1

One way to approximate a degradation trend, may be to develop a model based on physical laws. Problems, however, may be that the physics of a degradation process are not well understood or that values for some parameters of a degradation model are not available.

Alternatively, one may select a member of an appropriate class of potential degradation functions in analytical form and adapt its unknown parameters by curve fitting.

Another option may be to consider the determination of numerical values of an unknown degradation function as a parameter estimation problem.

To that end, a bicausal bond graph is used in this paper. Bicausal BGs were introduced by Gawthrop Gawthrop (1995) They extend the concept of computational causality by allowing that both co-variables, effort and flow of a bond may be inputs into a power port of an element. Accordingly, they may be used for parameter estimation and thus can be used for setting up an equation that determines the degradation function of a faulty element parameter $\Theta(t)$ at time instant t. Bicausal bond graphs have been used for FDI, e.g. in Samantaray and Ghoshal (2008). However, to the best of the author's knowledge, they haven't been used in online failure prognosis for the determination of numerical values of an unknown degradation function as proposed in this article. The approach is explained by means of a small power electronic circuit and verified by an offline simulation study in the next section.

3.1 A power electronic example

Consider the circuit schematic of a boost converter in Fig.9. It is assumed that the converter used, e.g. in power generation plants, operates in continuous conduction mode (CCM) with a sensor for the inductor current i_L and a sensor of the output voltage V. A fault in this system component may lead to a failure in a power distribution system or to a degradation of its performance.

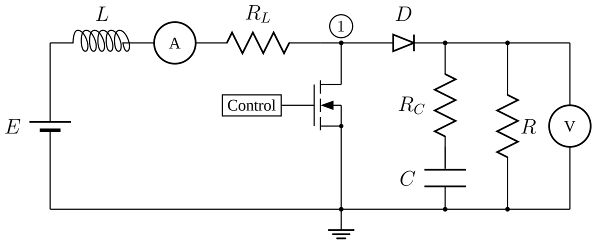


Figure 9: Circuit schematic of a boost converter (Borutzky (2018a))

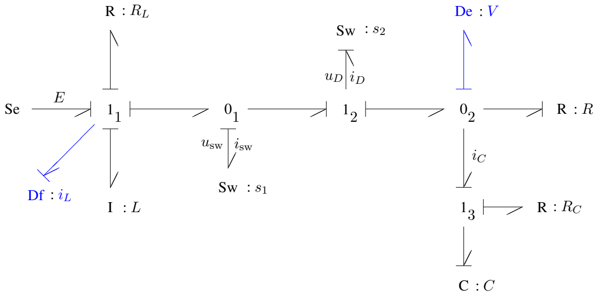


Figure 10: DBG of the boost converter in Fig. 9 (Borutzky (2018a))

If the MOSFET transistor and the diode are modelled as two conversely commutating ideal switches Sw : s_i, $i = 1,2$, then the circuit immediately transforms into the DBG in Fig. 10. If the small equivalent series resistance, R_C, of the capacitor is neglected and if variables are averaged over the switching period, then the circuit may be presented by the DBG in Fig. 11, in which d denotes the duty ratio. From the DBG in Fig. 11, the following two ARRs are easily derived.

$$\mathrm{ARR}_1 : r_1 = E - R_L - L\frac{d}{dt}\tilde{i}_L - (1-d)\tilde{V} \tag{8}$$

$$\mathrm{ARR}_2 : r_2 = (1-d)\tilde{i}_L - C\frac{d}{dt}\tilde{V} - \frac{1}{R}\tilde{V} \tag{9}$$

Their structural dependencies from element parameters is represented by the FSM in Table 5. As can be seen, all parametric faults can be detected by means of the two sensors but none can be isolated apart from the faulty duty ratio d.

In the following, two general cases are considered illustrated by means of the small boost converter circuit. Firstly, it is assumed that the parameter of a resistive element is degrading with time. The other scenario is that the parameter of a storage elements progressively deviates from its nominal value. For both cases it is shown how the values of the respective unknown degradation function can be estimated by means of known inputs and measured values or simulated measurements.

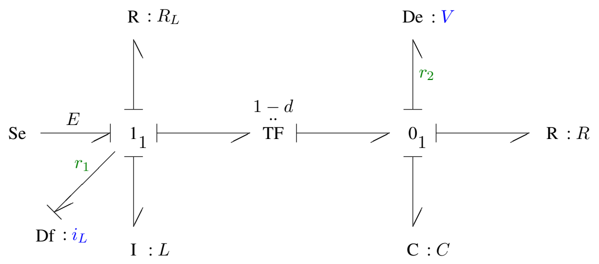


Figure 11: Averaged DBG of the boost converter in Fig. 9

Table 5: Fault signature matrix of the DBG in Fig. 11

Element	r_1	r_2	D_b	I_b
Se : E	1	0	1	0
R : R_L	1	0	1	0
I : L	1	0	1	0
TF : d	1	1	1	①
C : C	0	1	1	0
R : R	0	1	1	0

3.2 Degradation function of a resistor

Assume that by means of additional sensors the cause of an abnormal dynamic behaviour has been isolated and is attributed to a resistance R that is increasingly deviating from its nominal value R_n with time, i.e. $R(t) = R_n + \Phi_R(t)$. Given monitored measurements, the task is to determine the time-varying resistance $R(t)$. Accordingly, the bond attached to the port of the R-element is replaced by a bicausal bond as depicted in Fig. 12. As can be seen from the bicausal DBG in Fig. 12, there is a causal path from the flow detector Df : $\tilde{i}_L$ and another one from the effort detector De : $\tilde{V}$ to the power port of the R-element. That is, both port variables are determined by real measurements or simulated data provided by sensors into the DBG model so that the time evolution of the possibly nonlinear resistance $R(t) = R_n(t) + \Phi_R(t)$, i.e. numerical values of the degradation function $\Phi_R(t)$ can be computed. From the bicausal DBG, one obtains

$$\begin{aligned}\tilde{V} &= R(t)[(1-d)\tilde{i}_L - C_n\dot{\tilde{V}}] = [R_n + \Phi_R][(1-d)\tilde{i}_L - C_n\dot{\tilde{V}}] \\ &= R_n[(1-d)\tilde{i}_L - C_n\dot{\tilde{V}}] + \Phi_R[(1-d)\tilde{i}_L - C_n\dot{\tilde{V}}] \qquad (10)\end{aligned}$$

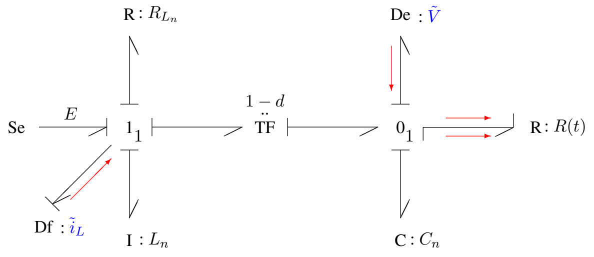


Figure 12: Averaged bicausal DBG of the boost converter

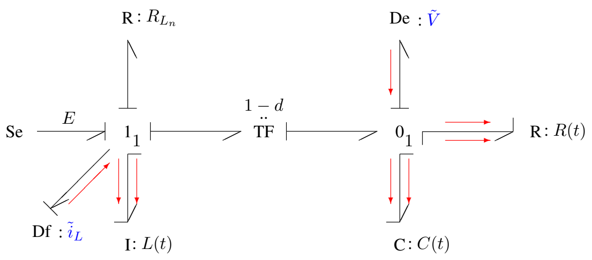


Figure 13: Averaged bicausal DBG of the boost converter

and as a result an implicit algebraic equation for the unknown degradation function $\Phi_R(t)$.

$$\Phi_R(t)[(1-d)\tilde{i}_L - C_n\dot{\tilde{V}}] = -R_n \underbrace{[(1-d)\tilde{i}_L - C_n\dot{\tilde{V}} - \frac{\tilde{V}}{R_n}]}_{=:\, r_2^1} \quad (11)$$

or

$$(\tilde{V} + R_n r_2^1)\Phi_R(t) = -R_n^2 r_2^1 \qquad (12)$$

Equation (12) is identical to (21) in Borutzky (2018a), in which r_2^1 denotes an ARR residual derived from the DBG of the first stage.

3.3 Degradation function of a storage element

In the bicausal DBG of Fig. 13, the bond attached to the power port of the C-element has also been turned into a bicausal bond. As a result, there are causal paths from the two detectors to the C-port so that the numerical values of a decaying capacitance $C(t)$ could be determined. However, these causal paths are not disjoint from the ones to the R-port so that it cannot be decided whether an abnormal dynamic system behaviour is caused by a degradation of the resistance R or of the capacitance C which is also expressed by the FSM in Table 5. This is not surprising, because both elements are in parallel, the voltage drop across both elements is the same. As addressed in Borutzky (2018b), another junction with an additional detector attached is to be inserted for isolation if both elements are faulty. Therefore, the following case assumes that only the capacitance, which may be nonlinear, follows an unknown degradation function, i.e. $C(t) = C_n(t) + \Phi_C(t)$. Observing the causal paths from the detectors to the C-element, the following implicit differential equation for the degradation function $\Phi_C(t)$ can be obtained from the DBG in Fig. 13.

$$\frac{d}{dt}(\Phi_C\tilde{V}) = \underbrace{(1-d)\tilde{i}_L - \frac{\tilde{V}}{R_n} - \frac{d}{dt}(C_n\tilde{V})}_{r_2^1} \qquad (13)$$

or

$$\Phi_C(t)\tilde{V}(t) = \int_{t_{f_2}}^{t} r_2^1(\tau)d\tau + \Phi_C(t_{f_2})\tilde{V}(t_{f_2}) \qquad (14)$$

where t_{f_2} denotes the time instant when the incipient fault exceeds an (adaptive) threshold and thus is detected. That is, $\Phi_C(t) \neq 0$ for $t > t_{f_2}$. Below that threshold the value of the capacitance may vary. However, a robust fault detection neglects small deviations from nominal parameter values in order to avoid false alarms.

Equation (14) equals the result (29) in Borutzky (2018a) obtained by an approach with two diagnostic bond graphs.

Finally, as can be seen from the bicausal DBG in Fig. 13, there are another two causal paths from the detectors to the inductor with a faulty inductance $L(t) = L_n(t) + \Phi_L(t)$. The causal path from the voltage detector to the inductor is not disjoint from the causal path to the resistor $\mathrm{R} : R$ and the one to the capacitor. That is, these parametric faults cannot be isolated in accordance to the FSM in Table 5. If it is only the inductor that has become faulty as of a time instant t_{f_1}, then similar to the computation of $\Phi_C(t)$ above, one obtains for the unknown degradation function $\Phi_L(t)$ from the bicausal DBG

$$\frac{d}{dt}(\Phi_L \tilde{i}_L) = \underbrace{E - R_{L_n}\tilde{i}_L - \frac{d}{dt}(L_n \tilde{i}_L) - (1-d)\tilde{V}}_{=:\, r_1^1} \qquad (15)$$

or

$$\Phi_L(t)\tilde{i}_L(t) = \int_{t_{f_1}}^{t} r_1^1(\tau)d\tau + \Phi_L(t_{f_1})\tilde{i}_L(t_{f_1}) \qquad (16)$$

where r_1^1 equals a first stage DBG ARR residual obtained by the two DBGs approach in Borutzky (2018a). The integration in (14), (16) may be performed by means of the trapezoidal rule.

4. OFFLINE SIMULATION STUDY

In the following, the above numerical determination of a capacitance degradation shall be verified in an offline simulation. That is, real measurements are replaced by simulated ones obtained from a BG model of the faulty circuit. Capacitor leakage is a typical fault. In Kulkarni et al (2010), it is reported that electrolytic capacitors in power electronic systems have a higher failure rate than other system components. In this study, it is assumed that the decay of the capacitance is exponentially with time according to the function

$$C(t) = \begin{cases} C_n & t < t_0 \\ \frac{1}{5}C_n + \frac{4}{5}C_n e^{-\lambda(t-t_0)} & t \geq t_0 \end{cases} \qquad (17)$$

That is, as of time instant t_0 the capacitance reduces exponentially with $t \to \infty$ to one fifth of its nominal value C_n.

The objective of the simulation is to recover this degradation from available 'measured' data $\tilde{i}_L$ and $\tilde{V}$ provided by a model with variables averaged over the switching period. Although averaging results in some smoothing, measurement noise is taken into account by adding 1% Gaussian noise to the output signals of the BG model. In a bicausal DBG, measurement uncertainties can be represented by modulated sinks (cf. Fig. 8).

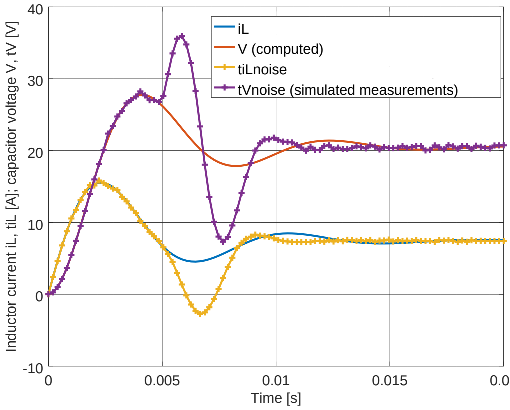


Figure 14: Effect of the capacitor degradation as of $t_f = 0.005s$ on the inductor current $\tilde{i}_L$ and the capacitor voltage $\tilde{V}$

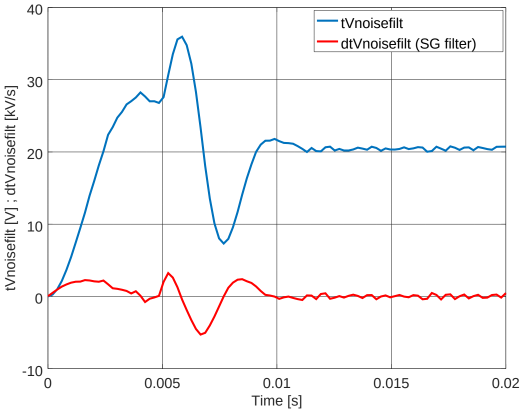


Figure 15: Filtered noisy capacitor voltage tVnoisefilt and its derivative dtVnoisefilt

The simulation performed by the free software GNU Octave 4.4.1 uses the parameters given in Table 6. The effect of the capacitance degradation on the inductor current $\tilde{i}_L$ and the capacitor voltage $\tilde{V}$ is displayed in Fig. 14 in which the tilde is substituted by the letter t prefixing the variable name. Simulated noisy measurements are obtained by means of the Octave function `randn()`.

```
tV = tV + 0.01*tV .* randn(linspace(tV))(18)
```

The noisy signals have been smoothed by a Savitzky-Golay filter Savitzky, A. and Golay, M.J.E. (1964) (Octave function `sgolayfilt()`). In (11), the derivative of the simulated measurement $\tilde{V}$ is needed. Differentiation and smoothing has also been carried out by a Savitzky-Golay filter. The result is displayed in Fig. 15.

Fig. 16 finally displays the recovered decay of the capacitance $rC(t)$.

As can be seen, the time evolution of the recovered capacitance $rC(t)$ is sufficiently close to the decay of the capacitance $C(t)$ deliberately introduced into the be-

Table 6: Parameters of the averaged DBG model in Fig. 11

Parameter	Value	Units	Meaning
E	12.0	V	Voltage supply
L	1.0	mH	Inductance
R_{L_n}	0.1	Ω	Resistance of the coil
C_n	500	μF	Nominal capacitance
R_n	5.0	Ω	Nominal load resistance
T_s	10^{-3}	s	Switching time period
d	0.45	–	Duty ratio
t_f	0.005	s	Capacitance starts decline
λ	500	s^{-1}	Rate of decline

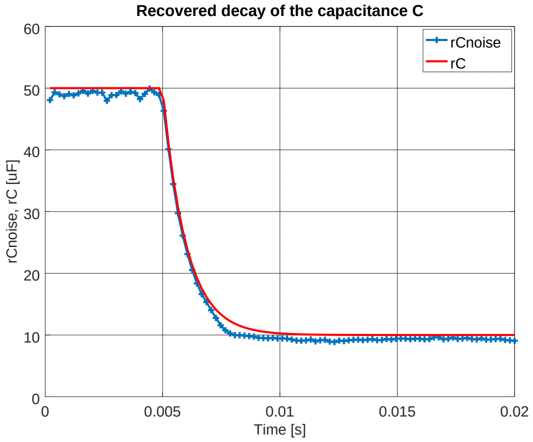


Figure 16: Recovered capacitance rC(t), rCnoise(t)

havioural model of the circuit.

5. REMAINING USEFUL LIFE ESTIMATION

As explained and illustrated in Section 3, equations for unknown parameter degradation functions for resistive as well as storage elements can be directly derived from a bicausal diagnostic bond graph by following causal paths from sources and detectors in inverted causality to the port of a faulty element. That is, inputs to these equations are only known control inputs and measurements. Numerical values of a parameter degradation function can be computed online concurrently to the monitoring of the health of a real system and the constant measurement of signals. As soon as n measured values of each needed signal are available and stored in a buffer, the time evolution of a faulty parameter $\Theta(t)$ can be approximated up to a time instant t and can be projected into the future to see when it would intersect with a failure threshold.

5.1 Data-driven failure prognosis

Once computed numerical values of an unknown function of the degradation of parameter Θ_i are available, they may be treated like degradation data of a feature extracted from measurement data. Direct measurement of degradation is often not possible without being invasive or destructive. The computation of a time-series of degradation data by an evaluation of equations derived from a bicausal BG can be considered the data acquisition phase.

Given n available degradation data $\Phi_\Theta(t_i^k)$, $i = 1, \ldots, n$ obtained from real measurements or simulated measurements in a sliding time window w^k that are stored in buffer of fixed size, a number of basic mathematical models, i.e linear, exponential, or power models with parameters to be determined may be tested to see which one fits best the data in current window w_k. This task of learning a mathematical model can be carried out e.g. by a commercial software such as the Matlab Predictive Toolbox The Mathworks (n.d.) and can be performed in parallel on a multiprocessor, multicore computer. As an evaluation criterion for the best fit, the root mean square error (RSME) may be used. The degradation function $\Phi_\Theta(t)$ found can then be used to determine a time point t_f^k at which the time evolution of the faulty parameter $\Theta(t)$ intersects with a given failure level threshold.

The time span from the current time t_c^k (current age of the system) to the time instant t_f^k where the time evolution of the parameter $\Theta(t)$ obtained from degradation data in the k^{th} window w^k intersects with a failure level threshold (end of life, EOL) gives an estimate of the remaining useful life RUL^k.

$$\text{RUL}^k := t_f^k - t_c^k \tag{19}$$

With progressing time new degradation values of a parameter Θ_i become available while older values drop out of the buffer. For a new time window w^k, the two steps, i.e. the determination of the best fit degradation model and its extrapolation are repeated.

As time has advanced, i.e. the system has become older, that is, t_c takes a new value and the intersection with the failure level threshold gives a new time to failure value. As a result, one obtains a new value for the RUL. Repeating these steps while time is progressing results in a sequence of k RUL estimates RUL$^k(\Theta_i)$ which ultimately converge to zero as a component reaches its EOL. This two step prognosis procedure consisting of a regression analysis of the degradation data in a window and an extrapolation that provides an estimate of the time to failure can be carried out simultaneously for multiple faulty components and in parallel on a multicore, multiprocessor computer. The global system-level RUL is then the infimum of all component RULs. An advantage of a repeated identification of a mathematical model for the degradation is that in case of a hybrid model for each system mode of operation a possibly different rate of degradation can be taken into account. In systems represented by a hybrid model, degradation of a component in ON mode may stop when the component switches into OFF mode. An example may be the mass flow through an increasingly contaminated valve. As long as the valve is open, its discharge coefficient, c_d, decreases with time. Clearly, when the valve is switched off, when this system component becomes inactive, then the last value of the discharge coefficient before closure is retained, degradation is stopped as long as the valve is in OFF mode, i.e. the decline of the time evolution $c_d(t)$ becomes zero. That is, extrapolating the time evolution of the faulty parameter from the current sliding window does provide no RUL estimate. In that system mode, the system behaviour is not affected by the faulty valve and nothing can be said about the RUL.

The numerical determination of a degradation function $\Phi_\Theta(t)$ and the projection of $\Theta(t)$ into the future requires a sufficient number of degradation data in the current window w^k in order to accurately identify the parameter of a potential degradation model. The amount of available degradation data, i.e. the size of the sliding window, affects the uncertainty in the values of the degradation model parameters and has an effect on the estimation of the time to failure. Software such as Weibull++ can compute upper and lower bounds for the time to failure with a certain confidence level. In order to meet given accuracy requirements for the parameters of the degradation model to be fitted, the size of the sliding window may vary. The boundaries for the time to failure become more narrow as the sliding time window moves on, i.e. the prediction of the time to failure becomes more accurate as a faulty component approches its EOL.

In the case of the capacitance degradation considered in Section 4, fitting of degradation data in each window gives the same exponential function $C(t)$ and its intersection with a failure threshold level C_{crit} the same time to failure t_f. Let α, β, γ be the identified parameters of the exponential function $\Phi_C(t)$ fitting the degradation data in a window. Then the time to failure t_f is determined by the condition

$$C(t_f) = \alpha C_n + \beta C_n e^{-\gamma(t_f - t_0)} = C_{\text{crit}} \tag{20}$$

Solving for t_f gives

$$t_f = t_0 - \frac{1}{\gamma}\left(-\ln\beta + ln\left(\frac{C_{\text{crit}}}{C_n} - \alpha\right)\right) \tag{21}$$

Equation 21 indicates that the time to failure and the RUL, $\text{RUL}(C,t) := t_f - t$ depend on the fitting parameters. The true RUL is obtained for $\alpha = 1/5$, $\beta = 4/5$ and $\gamma = 500$. These parameters and $C_{\text{crit}} = 2/5$ yield $t_f = 7.777\,\text{ms}$. Fig. 17 depicts the exact RUL.

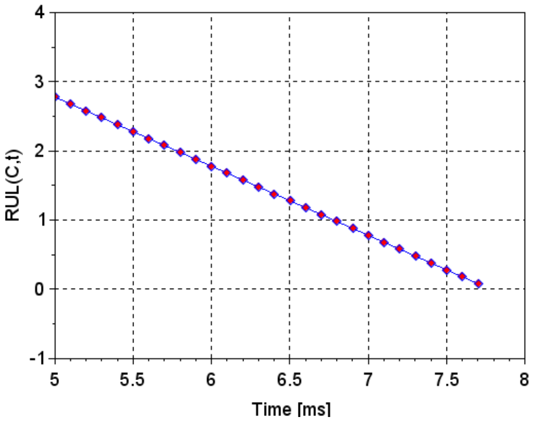


Figure 17: RUL of the decaying capacitance

5.2 Uncertainties in failure prognosis

There are some inherit uncertainties with the proposed non-residual based approach to failure prognosis.

First, the bicausal BG from which equations for unknown degradation functions are derived relies on modelling assumptions (model uncertainty).

Entries into the derived equations besides known control inputs are measurements that carry noise.

As the numerical computation of time-series degradation data is based on measurements, the parameters of a potential degradation function are to be considered random parameters for which a probability density function (pdf) has to be assumed which affects the prediction of the k^{th} time to failure t_f^k and the estimation of a RUL^k. As a result, RUL prediction does not provide a single value but a pdf. Assuming a distribution for the degradation model parameters upper and lower bounds for a RUL prediction at a time point given a required confidence level can be computed.

Prediction of the time to failure clearly depends on the failure threshold that has been set. With insufficient a priori knowledge the choice of an alarm threshold below the EOL failure threshold ensuring a safety margin is uncertain so that for a failure threshold a pdf has to be assumed. A proper choice of an alarm threshold is crucial as the intersection of an extrapolated degradation trend provides a time instant at which a decision on the action to taken must be made.

5.3 Advantages of the proposed approach

The proposed combined bond graph model-based, data-based prognosis approach has the following advantages.

The computation of numerical values of an unknown degradation function in the data acquisition phase by evaluating an equation derived from a bicausal DBG can be performed in parallel for multiple simultaneous parametric faults.

For the fitting of ‘measured’ degradated data pertaining to a faulty component, the parameters of various potentially appropriate basic mathematical functions can be computed in parallel by means of existing software. A criterion such as the RSME can single out the best fitting function among a set of potential candidates.

The repeated identification of a best fit degradation model on the data of a sliding window enables to account for possible changes of the degradation behaviour from window to window that may be due to changes of the system mode of operation or may be caused by changes in the system's environment. Extrapolating repeatedly the time evolution of a faulty parameter from a window to the subsequent one results in a sequences of values for the time to failure and the RUL.

CONCLUSION

The paper contributes to the sensor placement problem by proposing a graphical check whether potential faulty elements can be isolated by means of a given set of sensors and how their number can be increased by adding detectors in appropriate places of a DBG model. The issue of sensor placement has been addressed because parametric fault isolation is a prerequisite for failure prognosis. Faulty sensors can be modelled as discussed in Section 2. Parametric degradation in actuators will be accounted for in future work.

Furthermore, it has been shown that by following causal paths in a bicausal DBG from detectors in inverted causality to the power port of a possibly nonlinear element with the parameter $\Theta(t)$ identified as faulty, i.e. $\Theta(t) = \Theta_n(t) + \Phi_\Theta(t)$, an equation can be established that determines the numerical values of the unknown parametric degradation function $\Phi_\Theta(t)$.

As the novel proposed computation of numerical values of unknown degradation functions is based on known input signals and on sampled values of measured signals and may require the differentiation of some signals in discrete time, sophisticated signal processing is important.

In the data-based failure prognosis part, the parameters of degradation models that best fit measured data have to be considered random with a probability density function and affects the projection into future. Uncertainties in RUL prediction have been addressed, e.g. in Sankararaman and Goebel (2013) and are considered a subject of further research.

Furthermore, there is a time delay between the occurrence and the detection of an incipient parametric fault. A parameter value deviating from its nominal value must not only touch a time dependent adaptive fault threshold but must increasingly deviate from these adaptive boundaries with time in order to be identified as a progressive fault. As a result, it takes some time until the first sliding window can be filled with degradation data.

REFERENCES

Alem S, Benazzouz D (2013) Optimal sensor placement for detection and isolation by the structural adjacency matrix. Int J of the Physical Sciences 8(6):225–230, DOI 10.5897/IJPS12.547

Benmoussa S, Ould Bouamama B, Merzouki R (2014) Bond graph approach for plant fault detection and isolation: Application to intelligent autonomous vehicle. IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING 11(2):585–593

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

Borutzky W (2018a) Determination of a function for a degradation process by means of two diagnostic bond graphs. In: Proceedings of the 10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, IFAC, Warsaw, Poland

Borutzky W (2018b) Sensor Placement on Diagnostic Bond Graphs For Maximum Structural Isolation of Parametric Faults. In: Granda J, Karnopp D (eds) Proceedings of the 13th International Conference on Bond Graph Modeling and Simulation (ICBGM'2018), SCS, Bordeaux, France, Simulation Series, vol 50(12), pp 41–49

Chi G, Wang D (2015) Sensor Placement for Fault Isolability Based on Bond Graphs. IEEE Trans on Automatic Control 60(11):3041–3046

Djeziri M, Ould Bouamama B, Merzouki R, Dauphin-Tanguy G (2009) Optimal sensor placement for fault diagnosis. In: 2009 IEEE International Conference on Mechatronics, Malaga, Spain, 2009

Escobar LA, Meeker WQ (2006) A review of accelerated test models. Statistical Science 21(4):552 – 577, DOI 10.1214/088342306000000321

Frisk E, Krysander M, Åslund J (2009) Sensor placement for fault isolation in linear differential-algebraic systems. Automatica 45:364 – 371

Gawthrop P (1995) Bicausal Bond Graphs. In: Cellier F, Granda J (eds) ICBGM'95, International Conference on Bond Graph Modeling and Simulation, SCS Publishing, Simulation Series, vol 27(1), pp 83–88

Jha M (2015) Diagnostics and prognostics of uncertain dynamical systems in a bond graph framework. PhD Thesis, École Centrale de Lille, Université Lille Nord-de-France

Kulkarni C, Biswas G, Koutsoukos X, Goebel K, Celaya J (2010) Physics of failure models for capacitor degradation in DC-DC converters. In: The Maintenance and Reliability Conference, MARCON, Knoxville, TN, U.S.A.

Medjaher, K, Zerhouni, N (2013) Hybrid Prognostic Method Applied to Mechatronic Systems. International Journal of Advanced Manufacturing Technology 69(1–4):823–834, DOI 10.1007/s00170-013-5064-0

Prakash O, Samantaray A, Bhattacharyya R, Ghoshal S (2018) Adaptive prognosis for a multi-component dynamical system of unknown degradation modes. In: Proceedings of the 10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes,, IFAC, Warsaw, Poland

Samantaray A, Ghoshal S (2008) Bicausal bond graphs for supervision: From fault detection and isolation to fault accommodation. Journal of the Franklin Institute 345:1–28

Sankararaman S, Goebel K (2013) Why is the Remaining Useful Life Prediction Uncertain ? In: Annual Conference of the Prognostics and Health Management Society, pp 1 – 13, open-access article

Savitzky, A, Golay, MJE (1964) Smoothing and differentiation of data by simplified least squares procedures. Analytical Chemistry 38(8):1627 – 1639

The Mathworks (n.d.) Three ways to estimate remaining useful life. White paper, `https://www.mathworks.com/products/predictive-maintenance.html`

FAILURE PROGNOSIS OF EMBEDDED SYSTEMS BASED ON TEMPERATURE DRIFT ASSESSMENT

Oussama Djedidi$^{(a)}$, **Mohand A. Djeziri**$^{(b)}$, **Samir Benmoussa**$^{(c)}$

$^{(a)(b)}$Aix Marseille University, Université de Toulon, CNRS, LIS, SASV, Marseille, France

$^{(a)(b)}$Laboratoire d'Automatique et de Signaux de Annaba (LASA), University Badji Mokhtar Annaba, 23000 Algeria.

$^{(a)}$oussama.djedidi@lis-lab.fr, $^{(b)}$mohand.djeziri@lis-lab.fr, $^{(c)}$benmous2a.samir@gmail.com

ABSTRACT

The Systems-on-Chip provide a large capacity for calculation and monitoring, so they are increasingly integrated into risky processes such as aeronautical and power generation systems. However, embedded systems are subject to degradation caused by wear, that can be accelerated by the often hostile environment. This paper proposes a method of failure prognosis of embedded systems based on the estimation of the temperature drift under reference operating conditions, then the modelling of the drift trend using a support vector regression model. The remaining useful life is estimated using the integral of the probability density function of the time to failure. Experimental results, evaluated by performance analysis techniques, show the effectiveness of the proposed approach.

Keywords : Remaining Useful Life; support vector regression; Fault Diagnosis; Systems-on-Ship

1. INTRODUCTION

The development of computer science and artificial intelligence has contributed to the gradual replacement of several functions of management and control in systems at risk by embedded electronic devices. This evolution must be accompanied by the development of methods of fault diagnosis and failure prognosis to ensure the operation of these systems safely and prevent any damage or failure. In this context, part of existing work consider electronic cards as discrete systems integrating several components, and whose functioning depends on proper operation of all the components, assuming that there is a strong dependence of operation between each component (Steininger 2000). In this caser, the developed fault diagnosis methods are based on causal models such as multi-signal flow graph (Deb, Pattipati, Raghavan, Shakeri & Shrestha 1995), information flow model (Sheppard 1996), directed graph (Zhang 2005) and the fault tree (Wang, Shi & Wang 2011). A classification and a comparative analysis of existing fault diagnosis methods applied on multicore processor architectures are proposed in Gizopoulos, Psarakis, Adve, Ramachandran, Hari, Sorin, Meixner, Biswas & Vera (2011) where four categories are ditinguished: redundant execution approaches which exploit the inherent replication of processor cores and threads in a multicore processor architecture (Aggarwal & Ranganathan 2007, LaFrieda, Ipek, Martínez & Manohar 2007, Mukherjee, Kontz & Reinhardt 2002), periodic Built-In Self-Test (BIST) approaches which advocate leveraging the built-in test mechanisms of processors traditionally used for manufacturing testing (Shyam, Constantinides, Phadke, Bertacco & Austin 2006, Foutris, Psarakis, Gizopoulos, Apostolakis, Vera & Gonzalez 2010), dynamic verification approaches (Austin 1999, Meixner, Bauer & Sorin 2007), and anomaly detection approaches (Racunas, Constantinides, Manne & Mukherjee 2007, Wang & Patel 2006, Li, Ramachandran, Sahoo, Adve, Adve & Zhou 2008). The results of the comparative study highlight the effectiveness of the dynamic verification approaches in targeting both transient faults, permanent faults and design bugs.

Several methods of failure prognosis have been developed in recent years. They can be classified in three categories: modal-based (Feng, Kvam & Tang 2016), data-driven (Nguyen, Djeziri, Ananou, Ouladsine & Pinaton 2016), and hybrid approaches that combine physical knowledge with data analysis for failure prognosis (Benmoussa & Djeziri 2017, Djeziri, Nguyen, Benmoussa & M'Sirdi 2017). These methods have been applied to electrical systems, mechatronics and renewable energy systems, but to our knowledge, there has been no approach applied to processing units. So, we propose in this paper the application of a failure prognosis method to a processing unit based on the monitoring of the temperature drift generated by the SoC under reference conditions of use. The link between the drift of temperature and the degradation process is formally demonstrated and found experimentally in several works (Hong & Kim 2010, Djedidi, Djeziri & M'Sirdi 2018), it can therefore be used as a health index under reference operating conditions. To detect this drift, a reference model, data-driven, is built, then the trend of this drift is modelled by a Support-Vector Regression (SVR), and the online projection of the estimate up to a predefined maximum drift allows the Remaining Useful Life (RUL) prediction.

This paper is organised as follows: Section 2 is devoted to an overview of the proposed approach. The system, its operation, its inputs and outputs as well as its modelling for the estimation of the temperature are described in Section 3. In Section 4, the HI generation and evaluation are presented, then the tend modelling method is described in section 5 and the RUL prediction method in section 6. The results obtained are shown and analysed in Section 7, then a conclusion and perspectives are given in section 8.

2. RUL PREDICTION SCHEMA

The proposed approach is summarised in four steps as illustrated in Fig. 1. The first step is devoted to the temperature modelling using an Autoregressive-moving-average (ARMAX) model. The second step concerns the Health Index (HI) generation, which represent the difference between the measured SoC temperature online and the estimated one. The trend of the temperature drift is modelled in the third step using a Support Vector Regression (SVR) with a Gaussian kernel. The RUL expectation is then estimated in the fourth step using the time integration of the Probability Density Function (PDF) until the predefined time-to-failure.

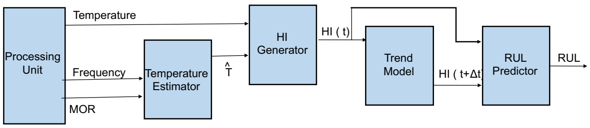


Figure 1: Overview of the RUL estimation process

3. TEMPERATURE MODEL

Experimental tests presented in (Hong & Kim 2010), show that the variation of the temperature of a Sytem-on-Chip (SoC) is influenced by two variables: the power consumed and the memory intensity (rate of access to the GDDR), and that it is the memory intensity which is the most influential on the variation of the temperature. This work also shows that the power consumed depends on the thermal voltage, and the thermal voltage is a function of the temperature, so there is an algebra-differential loop between the temperature and the power. Going back the cause-and-effect links between these two variables and the load, we find that the first common input is the frequency. Thus in the model proposed in this paper, the inputs considered are the frequencies of the CPU and the GPU, the Memory Ooccupation Rate (MOR) and the previous temperature of the SoC. The MOR is chosen because the memory intensity is not measurable.

To built a temperature estimator taking into account the regression property of the temperature, an autoregressive–moving-average (ARMAX) model is implemented. This models use the regression of inputs and previous outputs, along with the moving average to estimate the current output:

$$\begin{aligned} y(k) &= P_1 y(k-1) + \ldots + P_n y(k-n) \\ &\quad + Q_1 u(k-1) + \ldots + Q_m u(k-m) \\ &\qquad + e(k) + H_1 e(k-1) + \ldots + H_r e(k-r) \end{aligned} \qquad (1)$$

Equation (1) is the linear difference equation of an ARMAX (n,m,r) (Orders of the model), with $y(k)$ being the output to compute, u the exogenous (X) variable or system input, and e is the moving average (MA) variable (Fung, Wong, Ho & Mignolet 2003). In the case of this work, temperature T_{SoC} is the output $y(k)$ to be estimated, and the system input is $u(k) = [f_1, \ldots, f_4, f_{GPU}, MOR]$.

4. HEALTH INDEX GENERATION

In this work, the HI is generated by comparing the measured temperature with a reference one estimated by the model of normal operation presented above. This choice is motivated by the fact that this approach is well suited to detect a progressive deviation of the temperature drift from its value in normal operation. The HI is calculated by the following equation:

$$HI = (T_{estm}(t) - T_{meas}(t))/T_{estm}(t) \qquad (2)$$

The launch of the drift trend model for RUL prediction begins when a drift start is detected. To do so, HI is evaluated by a statistical method after analysing the data distribution. As the raw HI is normally distributed around its mean values, 99% of the data are inside the envelope delimited by the positive and negative value of mean plus three times the standard deviation. This property is used to calculate the normal operation thresholds.

Since the drift trend information is contained in the HI average and in order to reduce the false alarm rate, the raw HI is processed to extract its moving average HI_m. The averaged residual HI_m and the normal operating thresholds are expressed as follows:

$$\begin{cases} HI_m & = \frac{1}{n} \sum_{i=1}^{n} HI \quad , \\ th_{HI}^{+} & = \quad \| HI_m + 3 \times \sigma_{HI} \| , \\ th_{HI}^{-} & = - \| HI_m + 3 \times \sigma_{HI} \| \end{cases} \qquad (3)$$

σ_{HI} is the standard deviation of the HI signal.

5. TREND MODELLING

The long-term prediction of the evolution of a variable whose dynamics are unknown and complex, such as the SoC ageing process, can not be done by linear models like the $ARMA$ or $ARMAX$. For efficient prediction it is necessary to choose a model that supports nonlinear dynamics and variations of initial conditions. In this context, two tools stand out and are widely used, neural networks and SVR. In this work we choose the SVM as this tool is more suitable for unsupervised learning cases. The expression of the prediction by SVR at a step of the evolution of the health index is given as follows:

$$\hat{HI}(k+1) = \sum_{[i=(m-1)\tau+1]}^{N-1} \alpha_i^* K(HI(i), HI(k)) + b^* \quad (4)$$

where α_i^* are Lagrange multipliers and τ the delay. The standard SVR toolbox is used without making any special changes to the prediction of temporal overlays. The free parameters, C, ε, the size of the kernel (Gaussian), and the dipping dimension m are selected from a comprehensive search in the parameter space to optimize the performance of the prediction on the validation set. The available N observations are therefore shared between two sets of training and validation of respective sizes Ne and Nv. Values for which the prediction error at a step on the validation set is minimal are retained for the final prediction. Once the parameters are fixed, we read the prediction using all the N observations available. The predictions at several steps, ie, for the values $(k \geq N+1)$, are realized by the ration of the prediction at one step, using the estimated vectors $\hat{HI}(k)$ at the previous iterations and not the observations themselves.

6. RUL ESTIMATION

In this case study, the RUL is expressed as the expectation of the remaining time for HI to reach the threshold Th_p and given as follows:

$$Z_{t_j} = \inf\left\{x : HI\left(t+x\right) \leq Th_p \left| HI(t) > Th_p \right.\right\} \quad (5)$$

$$RUL\left(t_j\right) = E\left\{Z_{t_j}\right\} \quad (6)$$

The Probability Density Function (PDF) of Z_{t_j} is expressed as follows:

$$pdf_{Z_{t_j}}(x) = \frac{-\left(Th_p - HI(t_j)\right)}{\sqrt{2.\pi.x^3.\sigma^2}}.e^{-\left(\frac{\left(Th_p - HI(t_j) - \mu_{t_j}.x\right)^2}{2.x.\sigma^2}\right)} \quad (7)$$

Thus, $RUL_{t_j} = \int_0^{+\infty} pdf_{Z_{t_j}}(x)xdx$. The confidence interval from 5% to 95% probability is $[RUL_{t_j}^-, RUL_{t_j}^+]$ where $\int_0^{RUL_{t_j}^-} pdf(x)dx = 0.05$ and $\int_0^{RUL_{t_j}^+} pdf(x)dx = 0.95$.

7. APPLICATION

The development board used in this case study is shown in Fig 2. This device can run on Linux and Android, and is equipped with a ARM Cortex-A9 core processor and a core with 1 giga-byte of RAM. This card is not equipped with sensors to measure the power consumed, so an external ammeter and an oscilloscope are used to measure the power consumed (see Fig 2).

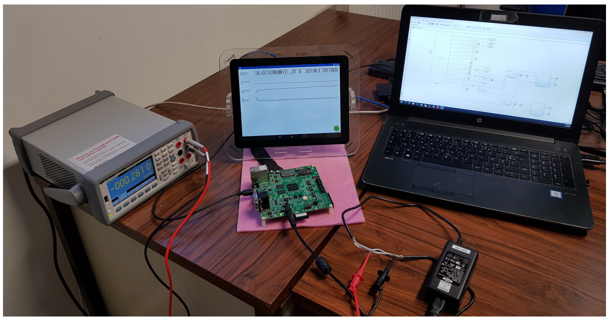


Figure 2: Overview of the application support including the development board, the display and the monitoring system.

In addition to the real SoC, a SoC simulator, developed and validated with real data in (Djedidi et al. 2018) is embedded in the monitoring system. The prediction algorithm proposed in this paper is validated using two types of degradation data: the first type are simulation data generated by the simulator. The second type is experimental data obtained on the real SoC over 4 years of use, these data are increased by data generated by the simulator to achieve the total failure. In This application case, the total failure is considered when the temperature drift is about 10 ° °C.

7.1. Validation of the diagnosis module

To validate the diagnostic part of the proposed approach, the algorithms for generating HI, moving-average filtering and generating normal operating thresholds are implemented in the supervisor. The results obtained are shown in Fig 3) for the normal operation and in Fig 4) for the case of degradation beginning emulated by overheating the SoC using a light projector of $1000W$.

7.2. Validation of the prognosis module

To validate the prediction model of the temperature drift, a SoC temperature drift was generated using the simulator. The result of the prediction is given in Fig 5. The SVM model inputs are the actual values of the drift. The result of RUL prediction follows the performance of the degradation prediction, and the real is estimated accurately. This first test shows that when the prediction is done over a short-time (with a permanent update of the measured drift) the result is very satisfactory with a negligible prediction error.

In the second test, it is assumed that the profile of HI is available until the time $t = 3000h$. From this moment, the inputs of the trend model are the values of HI estimated by the same model at the previous instants. The results are given in the Fig 6.

The results show that when the prediction is done in the medium term, the estimation error of the RUL is more important but remains acceptable as it represents $5,17\%$ of the real RUL.

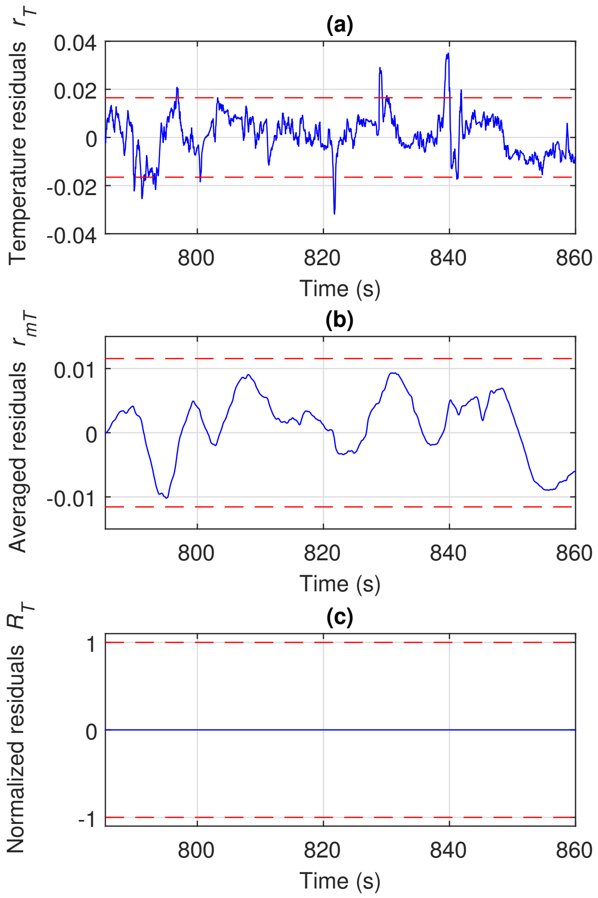


Figure 3: Health index profile in normal operation. (a): Raw HI, (b): Filtered HI, (c): Normalised HI

In the last test, we assume that HI's profile is available until $t = 2500h$. From this moment, the inputs of the trend prediction model are the values of HI estimated by the model at the previous instants. The results are given in the Fig 7.

The results show that when the prediction is long-term, the estimation error of the RUL is larger but is still acceptable as it represents 8% of the real RUL.

8. CONCLUSION

In this work, a generic method of failure prognosis of SoC is proposed in two major steps. The first step concerns the modelling of the SoC temperature and the generation of a health index characterising the SoC degradation process. This index is the drift of the temperature which represents the difference between the temperature generated by the SoC and a temperature reference generated by the ARMAX model. The second step is a prognostic one that concerns the estimation of the evolution of the short-term, medium-term and long-term temperature drift using a Support Vector Regression model. The RUL is estimated in this work using the PDF integration of the times to failure. The results obtained show the effectiveness of the proposed method on important prognostic horizons.

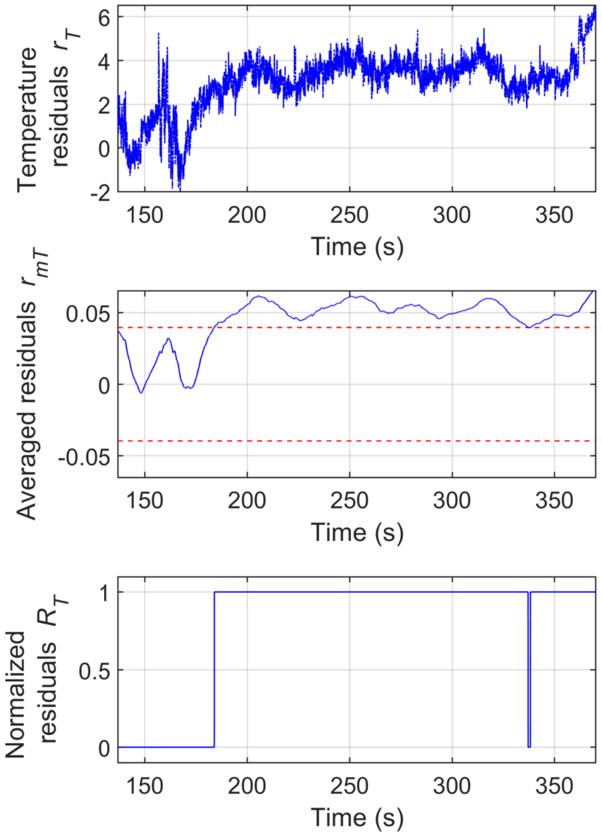


Figure 4: Health index profile at the beginning of the degradation. (a): Raw HI, (b): Filtered HI, (c): Normalized HI

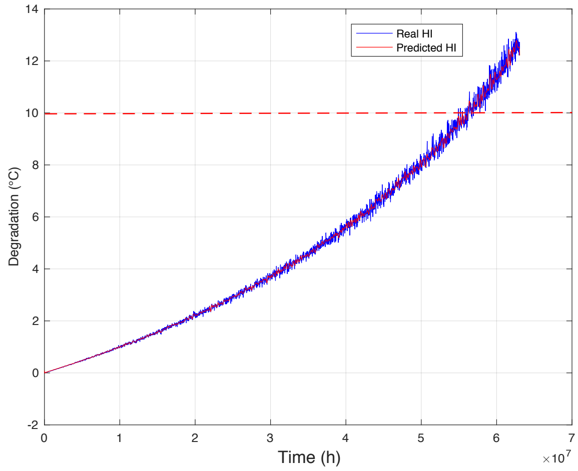


Figure 5: Degradation-trend and RUL prediction using one-step SVR model from the raw HI

REFERENCES

Aggarwal N. & Ranganathan P., (2007). Configurable isolation: building high availability systems with commodity multi-core processors. In: Acm Sigarch ..., vol. 35, New York, New York, USA: ACM Press, vol. 35, 470–481.

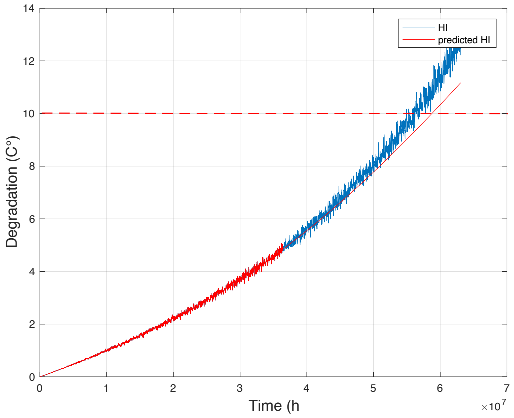


Figure 6: Degradation-trend and RUL prediction in the medium term using SVR model from the raw HI

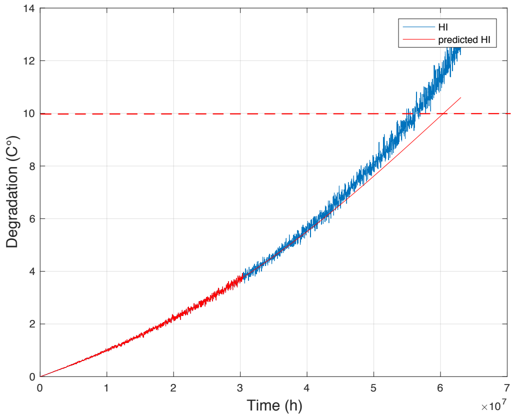


Figure 7: Degradation-trend and RUL prediction in the long-term using SVR model from the raw HI

Austin T.M., (1999). DIVA: A reliable substrate for deep submicron microarchitecture design. In: 32nd Annual Int. Symp. on Microarchitecture, (MICRO-32), IEEE Comput. Soc, November, 196–207.

Benmoussa S. & Djeziri M.A., (2017). Remaining useful life estimation without needing for prior knowledge of the degradation features. IET Science, Measurement & Technology, 11 (8):1071–1078. **URL:** *https://digital-library.theiet.org/content/journals/10.1049/iet-smt.2017.0005*

Deb S., Pattipati K.R., Raghavan V., Shakeri M., & Shrestha R., (1995). Multi-Signal Flow Graphs: A Novel Approach for System Testability Analysis and Fault Diagnosis. IEEE Aerospace and Electronic Systems Magazine, 10 (5):14–25.

Djedidi O., Djeziri M.A., & M'Sirdi N.K., (2018). Data-Driven Approach for Feature Drift Detection in Embedded Electronic Devices. IFAC-PapersOnLine, 51 (24):1024–1029.

Djeziri M., Nguyen T.B.L., Benmoussa S., & M'Sirdi N., (2017). Fault prognosis based on physical and stochastic models. In: 2016 European Control Conference, ECC 2016, IEEE, 2269–2274. **URL:** *http://ieeexplore.ieee.org/document/7810629/*

Feng J., Kvam P., & Tang Y., (2016). Remaining useful lifetime prediction based on the damage-marker bivariate degradation model: A case study on lithium-ion batteries used in electric vehicles. Engineering Failure Analysis, 70:323–342.

Foutris N., Psarakis M., Gizopoulos D., Apostolakis A., Vera X., & Gonzalez A., (2010). MT-SBST: Self-test optimization in multithreaded multicore architectures. In: Proceedings - International Test Conference, IEEE, 1–10.

Fung E.H., Wong Y., Ho H., & Mignolet M.P., (2003). Modelling and prediction of machining errors using ARMAX and NARMAX structures. Applied Mathematical Modelling, 27 (8):611–627.

Gizopoulos D., Psarakis M., Adve S.V., Ramachandran P., Hari S.K.S., Sorin D., Meixner A., Biswas A., & Vera X., (2011). Architectures for online error detection and recovery in multicore processors. 2011 Design, Automation & Test in Europe, (c):1–6.

Hong S. & Kim H., (2010). An integrated GPU power and performance model. In: ACM SIGARCH Computer Architecture News, vol. 38 of {ISCA} '10, New York, NY, USA: ACM, vol. 38 of {ISCA} '10, 280.

LaFrieda C., Ipek E., Martínez J.F., & Manohar R., (2007). Utilizing dynamically coupled cores to form a resilient chip multiprocessor. In: Proceedings of the International Conference on Dependable Systems and Networks, IEEE, 317–326.

Li M.L., Ramachandran P., Sahoo S.K., Adve S.V., Adve V.S., & Zhou Y., (2008). Understanding the Propagation of Hard Errors to Software and Implications for Resilient System Design. SIGOPS Oper. Syst. Rev., 42 (2):265–276.

Meixner A., Bauer M.E., & Sorin D., (2007). Argus: Low-Cost, Comprehensive Error Detection in Simple Cores. In: 40th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO 2007), IEEE, 210–222.

Mukherjee S.S., Kontz M., & Reinhardt S.K., (2002). Detailed design and evaluation of redundant multithreading alternatives. In: 29th Annual International Symposium on Computer Architecture, 2002. Proceedings, IEEE, 99–110.

Nguyen T.B., Djeziri M., Ananou B., Ouladsine M., & Pinaton J., (2016). Fault prognosis for batch production based on percentile measure and gamma process: Application to semiconductor manufacturing. Journal of Process Control, 48:72–80.

Racunas P., Constantinides K., Manne S., & Mukherjee S.S., (2007). Perturbation-based fault screening. In: Proceedings - International Symposium on High-Performance Computer Architecture, IEEE, 169–180.

Sheppard J., (1996). Maintaining diagnostic truth with information flow models. In: Conference Record. AUTOTESTCON '96, IEEE, 447–454.

Shyam S., Constantinides K., Phadke S., Bertacco V., & Austin T., (2006). Ultra low-cost defect protection for microprocessor pipelines. In: Proceedings of the 12th international conference on Architectural support for programming languages and operating systems - ASPLOS-XII, vol. 41, New York, New York, USA: ACM Press, vol. 41, 73.

Steininger A., (2000). Testing and built-in self-test – A survey. Journal of Systems Architecture, 46 (9):721–747.

Wang F.W., Shi J.Y., & Wang L., (2011). Method of diagnostic tree design for system-level faults based on dependency matrix and fault tree. In: 2011 IEEE 18th International Conference on Industrial Engineering and Engineering Management, IE and EM 2011, IEEE, PART 2, 1113–1117.

Wang N.J. & Patel S.J., (2006). ReStore: Symptom-based soft error detection in microprocessors. IEEE Transactions on Dependable and Secure Computing, 3 (3):188–201.

Zhang G., (2005). Optimum Sensor Localization/Selection In A Diagnostic/Prognostic Architecture. Phd dissertation, University System of Georgia.

MECHANICAL VERIFICATION OF DYNAMIC MUSCULOSKELETAL MODEL WITH MUSCLE – TENDON COMPLEX AT JUMPING MOTION BY BOND GRAPH

Katsuya Suzuki(a), Yukio Oida(b), Tatsuya Fukuoka(c), Yasumi Ito(d) , Tetsuya Nemoto(e)

(a)Cyukyo University, Institute for Advance Studies in Artificial Intelligence
(b) Cyukyo University, School of Health and Sport Sciences
(c)University of Yamanashi, Integrated Graduate School of Medicine, Engineering, and Agricultural Sciences
(d) University of Yamanashi, Graduate Faculty of Interdisciplinary Research
(e)National Center for Geriatrics and Gerontology, Center for Assistive Robotics and Rehabilitation for Geriatrics and Health

(a)kachandesu2002@yahoo.co.jp,(b) yoida@sist.chukyo-u.ac.jp, (c) g19dts06@yamanashi.ac.jp,
(d) yasumii@yamanashi.ac.jp, (e) nemo@ncgg.go.jp

ABSTRACT

Studies are being conducted to model jumping motion mechanically. Conventional studies have been conducted using only Muscle Tendon Complex (MTC), which consists of muscle and tendon. In this study, in addition to MTC, modeling was carried out considering lower limbs including the foot, ankle joint and bone as one system. In the modeling of the system, the idea of bond graph using not only force but power was considered. So far, studies have generally been done on the idea of using the ankle as a pivot. In this study, this idea was changed to the idea with the toe as the pivot. In modeling, this idea of using bond graph was very effective. Based on this method, modeling study was carried out and the model could be constructed. It confirmed that the results were comparatively agreed with the experimental results, and this model was confirmed that the results could be reproduced.

Keywords: bond graph, muscle-tendon complex, jumping motion, modeling

1. INTRODUCTION

Many studies on behavior of Muscle Tendon Complex (MTC) have been carried out so far. These studies which obtains the behavior by calculation, is examined using the mechanical model (Liber, R.L., et al. 1992) or the mathematical model (Maarten F. et al. 1986). Even in that study, many studies including the MTC have been done regarding jumping motion (Fukashiro, S. S. et al. 2005, Fukashiro, S., et al. 2006). For example, a study has been conducted to obtain the jumping height by simulation from the dynamic model mainly based on the MTC. On the other hand, since the development of the ultrasonography that can measure the length of MTC directly *in vivo*, the muscle fascicle length in the jumping motion has been measured (Fukunaga, T., 2002). Also, the relationship between the tension and the length in muscle and tendon, has been examined from the tension obtained from the floor reaction force. These studies have clarified the mechanism of the MTC (Kaneko 2011, Kawakami, et al. 2002, Kawakami and Fukunaga, 2006). In this study, we have studied to construct the model in order to calculate behavior of tension and length of muscle and tendon in MTC during jumping motion. Unlike the above mentioned method, the entire lower limbs including the MTC was considered as one physical system, and was studied based on mechanical engineering system. At the beginning of the study the mechanical model of MTC alone was constructed, and then the whole system including the ankle joint, the tibia, and the foot was constructed (Suzuki and Oida, 2015, 2016). In this system, it was considered to be important to clarify not only the forces transmission but also the power transmission flow. Modeling with the bond graph was performed based on this idea and the simulation was performed. In the first modeling method, the conventional idea in which the ankle joint was the pivot of the lever is considered. However, the calculated value of the tendon tension force was significantly different from the measured value (Kawakami 2002). Therefore, this idea of using the ankle joint as a pivot was studied in detail again. As a result, modeling method was changed to the new idea that the toe of the foot was the pivot. As a result of reconstructing the modeling, good results were obtained.

According to the literature, Thompson et al. have already reported that the action of the force around the ankle joint operates according to the second lever principle. However, there will be perhaps any questions in this idea after that. Moreover, it will be thought that the theoretical examination of this idea has not been done until now, so we report it in this paper. In addition, a simple experimental model was created, the model was dynamically and computationally verified, and the bond graph advantage was verified. This paper reports the result.

2. METHOD

In this study, we considered the whole lower limbs including the MTC as one mechanical system at the jumping motion. This system is constructed with mechanical components such as masses, springs, lever. This schematic models are represented by the bond graph which is one of the modeling methods of the system (Thoma, J. U. and Suda, N. 1996, Kanopp, D.C. et al. 2006). The analysis method of the bond graph is to describe the schematic models by representing the connection form of the components at first. Next, this schematic models is expressed by the word bond graph according to the power flow. Furthermore, the word bond graph is translated to the true bond graph, and analysis is performed by substituting variables into the model.

2.1. Overall composition of the model

In modeling, the musculoskeletal system of the lower limbs was considered as one system, and this system was composed of the MTC, the foot including ankle joint, tibia and fibula. This system is shown in Figure1. Figure 2 shows a schematic model which is rotated to the left by 90 degree from Figure1. In this model, one end of the MTC connects to the heel of the foot and the other end connects to the knee. Furthermore, the weight and the inertial mass of the body are acted on the knee. The knee is connected to the ankle joint through the tibia and fibula. With regard to the foot, it was assumed to be a rigid flat plate that one end is connected to the MTC with heel and the other end connected to the toe acting on the ground. The measuring plate of the force at the toe is set on the ground.

2.2. Modeling of the MTC

The MTC is an important factor, and it is assumed that the gastrocnemius muscle and soleus muscle in the human body is as one MTC. This MTC has two components, that is, muscle and tendon. According to the Hill-type model as shown in Figure 3, the muscle is composed of elastic element and power input element. On the other hand the tendon is composed of elastic element only. These can be represented by the word bond graph as shown in Figure 4. Furthermore, it is translated to the true bond graph as shown in Figure 5 In this true bond graph, the C element is an elastic element, and the SE element is the input source element that represents the power emitted by the chemical change such as glucide.
We explain the input force acting on the human body here. In the mechanical systems, the external forces generally act on the mass from the outside of the mass. On the other hand the input force acted by muscle act inside of the human body. Then the action force and the reaction force for this input act inside of the human body. Therefore the direction of input forces in the bond graph is shown in both side from 0-junction to 1-junction.

2.3. Detail consideration of the model relating to the ankle joint

As before described, the foot is assumed to be a flat rigid plate and that have each named heel, toe and ankle joint. In the initial study, it was considered to be acted by two moments, that was the moment by the toe force and the moment by heel force (the Achilles tendon force), being balanced at the ankle joint so as the pivot. These are shown in Figure 6. In this schematic and the bond graph model, the weight force acting on the ankle joint will be assumed to be ignored. From this assumption, the Achilles tendon force is obtained from the force acting on the "toe", where the I element means the force plate, that is, measuring plate and the ground. So it represents the earth mass, and the TF elements represent the ratio of the length from the ankle joint to the toe and the heel. Using the bond graph of Figure 6, the whole bond graph of the musculoskeletal model system can be represented as shown in Figure 7.

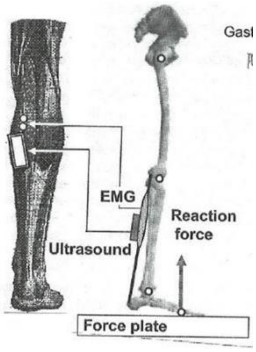


Figure 1 Musculoskeletal of the lower limbs (Fukashiro S. et al. 2006)

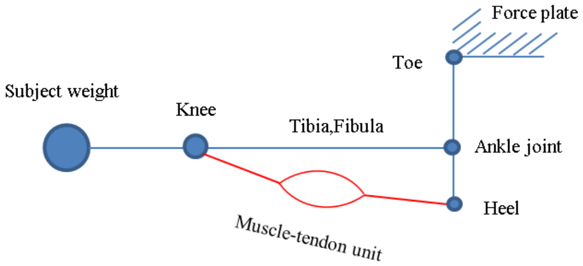


Figure 2 Schematics Musculoskeletal Model

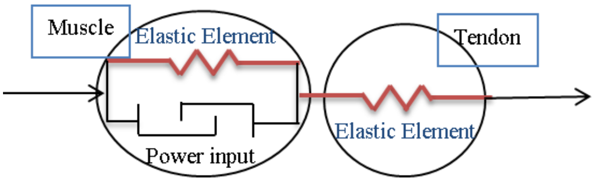


Figure 3 Schematics MTC model

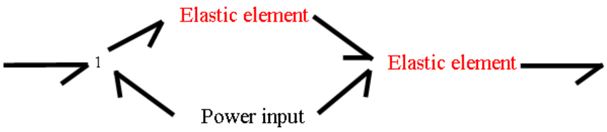


Figure 4 Word Bond Graph of MTC

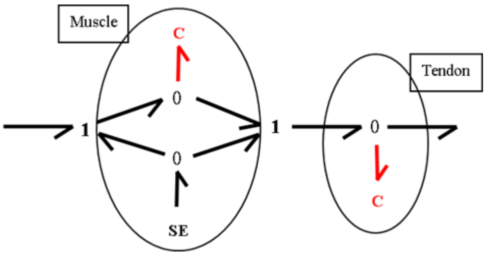


Figure 5 True Bond Graph of MTC

2.4. Comparison of calculated results and experiment value

This simulation results was compared with the experimental value 3081 ± 667 N (Kawakami et al. 2002). The simulation results are shown in Figure 8. The tendon force at the maximum value of the toe force was 350 N, which was about 1 / 8.5 times the experimental value of the tendon force

2.5. Reconsideration of the model relating to the foot and the ankle joint

Therefore, the model was reconsidered as follows. Unlike the conventional idea of sec. 2.3, the new idea that use the toe as the pivot instead of the ankle joint as the pivot, is considered to be a right one. Therefore it is thought that the force acting on the heel and the force acting on the ankle joint respectively act on the toe. As shown in Figure 9, the human weight and so on forces act on the point B (ankle joint), where the length l_1 is A to B. While the tendon force act on the point C (heel), where the length l_2 is A to C.

According to Figure 9, the schematic model and the bond graph model is represented as shown in Figure 10. The whole musculoskeletal system model including new idea is shown as the word bond graph in Figure 11. This word bond graph is translated to the true bond graph as shown in Figure 12.

The input power due to muscle contraction is generated from the SE element. One is transmitted from the heel to the toe by the transfer element TF of length l_2 through the C element of the tendon. The other muscle contraction power is input to the knee. In this knee, the SE element by the body weight and the I element of the inertial mass in the body, are added together and transferred to the ankle joint through the tibia and fibula. These combined powers are transmitted from the ankle joint to the toe by the transfer element TF of length l_1. Both powers

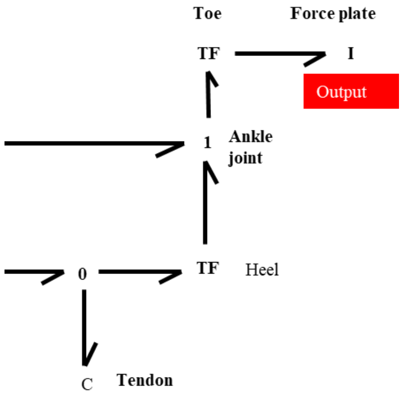


Figure 6 True Bond Graph where the Achilles Tendon Force Moment and the Toe Force Moment are balanced

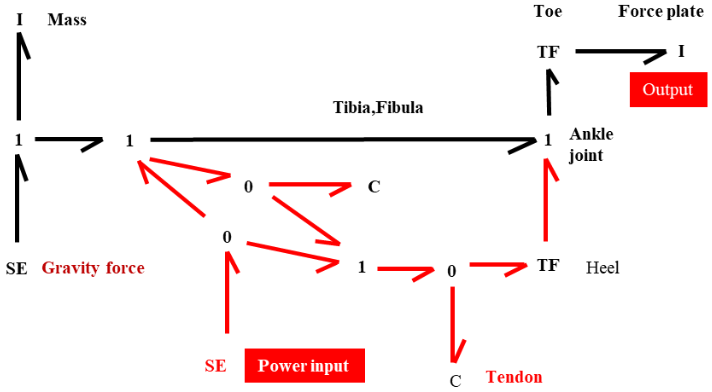


Figure 7 True Bond Graph of musculoskeletal model

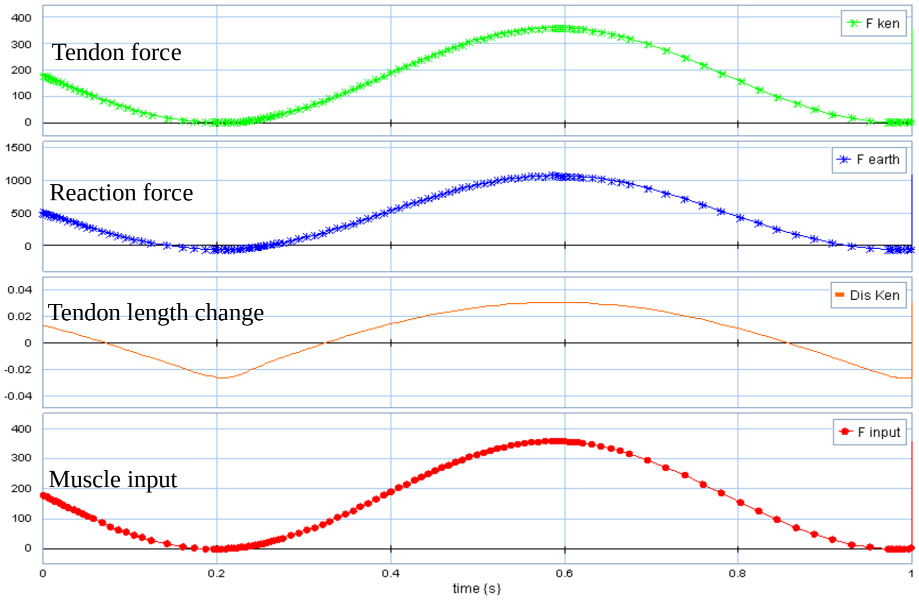


Figure 8 The Simulation Results

generated by MTC are combined and act on the toe force plate (measurement plate) to become an output power. In translating this word bond graph to the true bond graph in Figure 12, we add the direction of power flow. According to the mechanical bond graph construction procedure (System Dynamics: Karnopp et al., 2006), the downward speed of 1-junction (right direction in Figure 2) is defined as the positive direction in the direction of power flow. The spring is therefore positive in the compression direction.

2.6. Setting of calculation conditions

The following approximate conditions were assumed to carry out the calculation. In order to compare with the experimental results of Kawakami et al. (Figure 13), the output characteristics was the toe force (floor reaction force in Figure 13) acting on the force plate. The calculation was performed by approximately setting the following condition. That is, if the output characteristic of the experimental data (Kawakami, 2002) approximately are assumed to be a sine curve, then the muscle contraction force (that is input force) is also to be a sine curve. The sine curve of this muscle contraction force *F* is shown in Equation (1).

$$F = C + A\sin(\omega t) \tag{1}$$

where the amplitude *A*, the frequency ω and the constant *C* were identified to fit the experimental data These numerical values are A = -1300 N, ω = 5.5 rad / s, and C = -1300 N. Moreover the characteristics of the tendon stiffness were set on as following conditions. Based on the experimental results of Fukunaga as shown in Figure 14 (Fukunaga, 2002), it was assumed that the tendon tension was proportional to the square of the tendon length, and the stiffness C of the tendon was set. Furthermore, other element coefficients were set as shown in Table 1. From the experimental conditions of Kawakami et al., the lengths, l_1 and l_2 from the toe are 0.15 m and 0.2 m, respectively. Based on these lengths, coefficients for TF are 1 / 0.15 = 6.7 and 1 / 0.2 = 5.

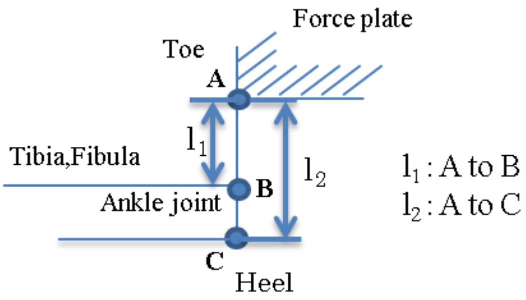


Figure 9 Detailed model around the ankle joint

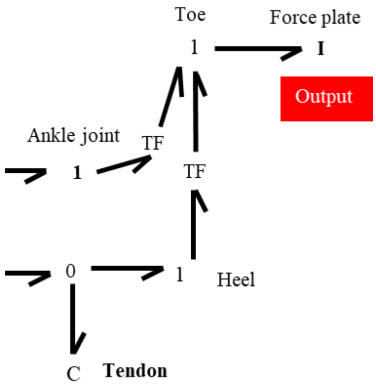


Figure 10 True Bond Graph around the ankle joint

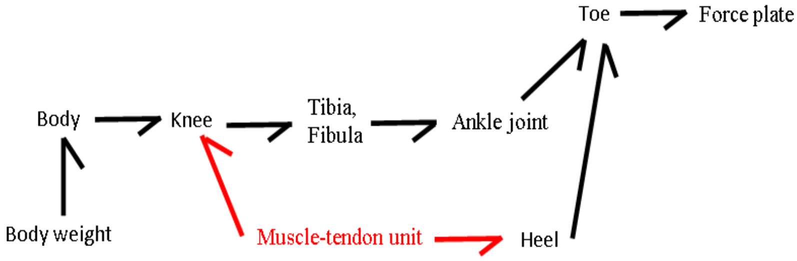


Figure 11 Word Bond Graph of Musculoskeletal Model

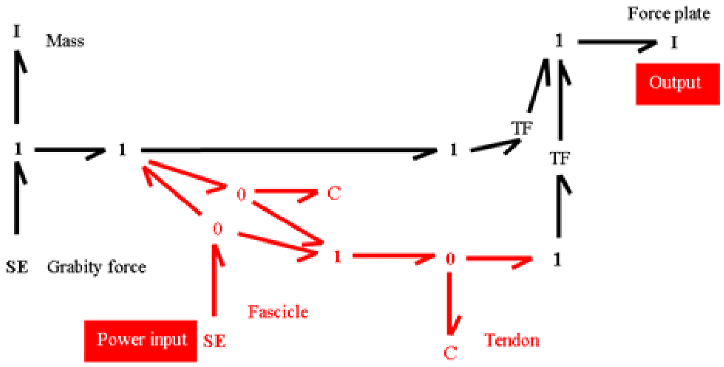


Figure 12 True Bond Graph of Musculoskeletal Model

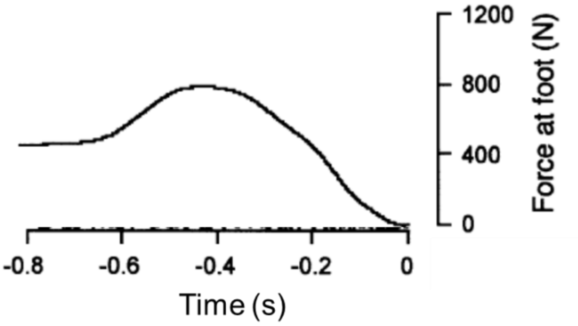


Figure 13 Force at foot (Kawakami et al., 2002)

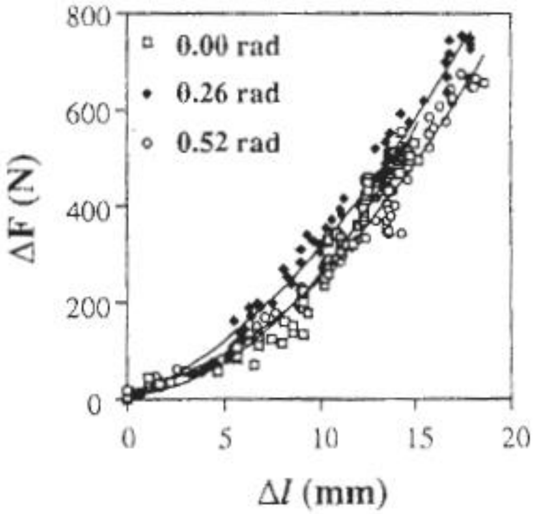


Figure 14 Relationship between tension and length of tendon in Fukashiro S. et al. Experiment (Fukuashiro, S. et al. 1995)

Table 1 Input value of element coefficient

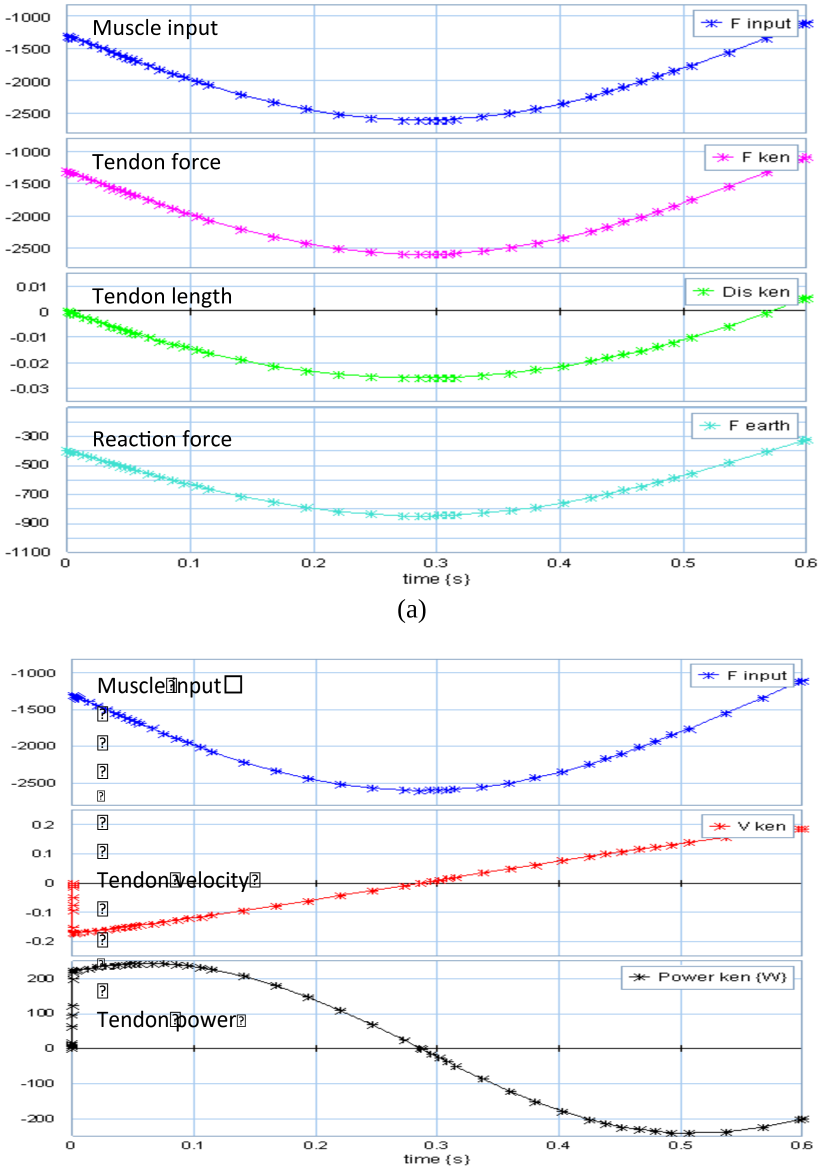
TF（Between point A and B）	6.7
TF（Between point A and C）	5.0
Se [N]	400
I (Force plate) [kg]	10^8
I (Weight) [kg]	40
C (Muscle) [m / N]	0.1
C (Tendon) [m / N]	3.0×10^{-6}

3. COMPARISON OF CALCULATION RESULTS VS. EXPERIMENTAL RESULTS AND CONSIDERATION ENERAL GUIDELINES

At first, the muscle tension is generally compressive force, then the input tension is negative (-). However in this paper, the input tension is set positive in the section 2: method.

The simulation was performed using modeling and simulation software (20-sim) where input specification was muscle contraction force (-) and major other specification were shown in Table 1. As a result, tendon force, tendon length and reaction force were obtained by continuous characteristics in time as shown in Figure 15 (a). The maximum tendon force (in this simulation result) was 2600 N. The experimental measure & estimated result was 3081 ±667 N. About the maximum tendon length, the calculated length was 25.8 mm, however the experimental measure & estimated result was about 30 mm, From these results, difference between the maximum force and length showed the same tendency for both the calculated and experimental results. From these results, this Musculoskeletal Model is considered to be almost valid.

In the paper of Kawakami et al. 2002, tendon elongation speed and tendon power have been obtained as estimated some points calculated from experimental measurements. However by use of this model, tendon elongation speed and tendon power could be obtained as continuous lines of the time function as shown in Figure 15 (b). Focusing on the speed and power that change with time, the speed reached its maximum immediately after the input force supply, then gradually decreased, and changed to speed in the negative direction. In addition, the tendon power was accumulated with positive energy in the first half. When the extension speed became negative, the power became negative, and it was possible to observe the progress of the stored energy being released. In this way, we were able to estimate characteristics of tendon speed and consumption energy that were difficult to estimate using conventional experiments alone. It will be considered that this estimation by use of this model has roughly established the fundamental foundation that enables quantitative prediction of the required characteristics in the future study of jumping motion. However, it cannot be said that the response was sufficiently obtained with respect to the experimental results of Kawakami et al. 2002 in which the muscle elongation changes in conjunction with the tendon. It is presumed that this is because the tension-extension properties of the muscle are not expressed as a function of simple displacement. Also, the angles of the ankle joints cannot be modeled. These are future issues. The operation principle of ankle joint in the lower limbs is described in section 1: introduction by Thompson et al. according to the second lever principle. This principle will be considered to have been proved by theoretical consideration in this paper.



(a)

(b)

Figure 15 Calculation Results of Tendon Characteristics

4. CONCLUSION

Dynamic mechanical model was constructed for the lower limbs of the human body including the MTU, and simulation calculations were performed by bond graphs. As a result, the tendon extension length and tension force in the MTU (muscle-tendon unit) at the jumping motion could almost be reproduced same as the experimental values (Kawakami et al. 2002). On the other hand, instead of that the tendon extension speed and power have been estimated, continuous characteristics of them were made possible by the graph. It was possible to see from the graph how the extension speed and power of the tendon change in extension and contraction at the jumping motion.

An experimental model that reproduces the movement of the human body was created and the dynamic model will be dynamically verified from now on.

ACKNOWLEDGMENTS

The authors thank Prof. Yasuo Kawakami in Waseda Univesity for helpful discussion and reference papers offered and thank Prof. Shinji Sakurai in Chukyo University for lever principle.

REFERENCES

Lieber, R. L., Brown, C. G. And Trestik, C. L., 1992. Model of Muscle-Tendon Interaction during frog Semitendinosis fixed-End Contractions, J. Biomechanics, 25 (4), 16-21.

Bobbert, M. F., Hujimg, P. A. and van Ingen Schenau, G. J., 1986. An Estimation of Power output and Work done by The Human triceps Surae Muscle- Tendon Complex in Jumping, J of Biomechanics, 19 (11), 899-906.

Borutzky W., Editor, 2011. Bond Graph Modelling of Engineering Systems, Springer

Fukashiro, S., Hay, D. C., Yoshino, S. and Nagano, A., 2005. Simulation of Muscle-Tendon Complex During Human muvements, International journal of Sport and Health Science, 3, 152-160.

Fukashiro, S., Hay, D. C., and Nagano, A., 2006. Simulation of Muscle-Tendon Complex During Human muvements, Journal of Applied Biomechanics, 22, 131-147.

Fukunaga, T., 2002. Shintai no Katachi to Chikara heno Kyoumi –Kenkyu Nakama ni Kansya no Kimochi wo Komete- (in Japanese), Prof Fukunaga Tetsuo Taikan Kinenshi Hensyu Iinkai, 174-184.

Kaneko, M., 2011. Sport Energy Gaku Jyosetsu, Kyorin-Shoin (in Japanese), 121-128 and 165-168.

Karnopp, D.C., Margolis, D. L., Rosenberg, R. C., 2006. System Dynamics, JOHN WILEY & SONS, INC.

Kawakami, Y., Muraoka, T., Ito, S., Kanehisa, H. And Fukunaga, T., 2002. *In vivo* muscle fiber Behavior during couter-movement exercise in humans reveals a significant role for tendon elasticity, Journal of Physiology, 540 (2), 635-646.

Kawakami, Y. And Fukunaga, T., 2006. New insights into *in vivo* Human Skeletal Muscle function, Exercise and Sport Scienses Reviews, 34 (1), 16-21.

Robertson, D. E., Caldwell, G. E., Research Methods in Biomechanics.

Roddeck, W., 2013. Grundprinzipien der Mechatronik, Springer

Rolf Wirhed., 1982. Anatomi och rearelselara inom idrotten (Sweden), Athletic Ability & the Anatomy of Motion (English).

Suzuki, K. and Oida, Y., 2015. Dynamic Characteristic of Muscle and Tendon Unit at jumping Motion, The Proceedings of the Symposium on sports and human dynamics. JSME 2015.

Suzuki, K. and Oida, Y., 2016. Dynamic Musculoskeretal Model with Muscle-Tendon Unit atJumo Motion by Bond Graph, The Proceedings of the Symposium on sports and human dynamics. JSME 2016.

Suzuki, K., Nakamura, I., and Tanaka, K., 1996. Analysis of Dynamic Characteristics of Torque Converter by Bondgraph, Journal of JSME 62 (595), 1067-1083

Suzuki, K., Tanaka, K., Yamakawa, Shimizu, F., Fuchiwaki, M., 2010. Fuel Consumption Improvement of Vehicle with CVT by Bond Graphs, 2010 7th IMAACA, Proceeding, 21-26

Thoma, J. U.,1990. Simulation by Bondgraphs, Springer-Verlag

Thoma, J. U. and Suda, N., 1996. Bond Graph ni yori Simulation, CORONA PUBLISHING CO.,LTD.

Tateishi, T., 2010, Harmony between Mechanics and Biology, Medicine. Ohmsha.

Thompson, C.W., Floyd, R.T. 2001. Manual of Structural Kinesiology, 2001, the McGraw-Hill, Inc.

20-sim, Controllab Products B.V., 7521 AN Enschede, The Netherlands

AUTHORS BIOGRAPHY

Katsuya Suzuki
Dr. Eng.
Cyukyo University
Institute for Adavance Studies in Artificial Intelligence

Yukio Oida
Ph. D
Prof. Cyukyo University
School of Health and Sport Sciences

Tatsuya Fukuoka
M. E.
University of Yamanashi
System Integration Engineering Course
Department of Engineering, Doctoral Course,
Integrated Graduate School of Medicine, Engineering, and Agricultural Science

Yasumi Ito
Ph. D
Prof. University of Yamanashi
Graduate Faculty of Interdisciplinary Research

Tetsuya Nemoto
Ph. D
National Center for Geriatrics and Gerontology
Chief, Section of Medical Technology
Center for Assistive Robotics and Rehabilitation for Geriatrics and Health

DETECTION OF STATOR WINDING INTER-TURN SHORT CIRCUIT FAULT IN INDUCTION MOTOR USING LS-SVM

M’hamed Birame[(a)], Sid Ahmed Bessedik[(b)], Aziz Naamane[(c)]

[(a)]LEDMASED Laboratory, University Of Laghouat, 03000, ALGERIA.
[(b)]LACoSERE University of Laghouat, 03000, ALGERIA.
[(c)]LSIS - Laboratory Paul Czanne University, FRANCE.
[(a)]bir_moh@yahoo.fr , [(b)]s.bessedik@lagh-univ.dz, [(c)]aziz.naamane@lis-lab.fr,

ABSTRACT

A variety of approaches have been proposed for monitoring the state of machines based on intelligent techniques such as neural network, fuzzy logic, neuro-fuzzy, pattern recognition. However, the use of LS-SVM for machine condition monitoring and fault diagnosis is still rare. For this reason, LS-SVM approach has been investigated in this study for inter-turn fault detection in stator winding of induction motor. The proposed method uses as input the stator current and decides the motor condition as output by indicating the severity of the short-circuits fault.

Keywords: Induction Motor, Inter-turn short circuit, Fault diagnosis, least square support vector machine (LS-SVM)

1. INTRODUCTION

Induction motors (IM) are the mainstay for every industry. However like any other machine, they will eventually fail because of heavy duty cycles, poor working environment, installation and manufacturing factors, etc. With escalating demands for reliability and efficiency, the field of fault diagnosis in induction motors is gaining importance [1-3]. The diagnosis and oversight of a device are generally passed through the knowledge of his healthy behaviour, total control of the various modes of operation is then essential when considering an advanced monitoring of the process

For this purpose, their maintenance and their diagnosis have become an economic issue. Therefore, it is recommended to detect faults early in order to remedy them in the shortest possible time to minimize the effects on the electrical installation or on the machine itself. Broadly, an induction motor can develop either internal fault or external fault. With reference to the origin, a fault may be mechanical or electrical. Fault can be classified as stator fault or rotor fault depending on the location of the fault. Faults associated with the moving parts like bearing and cooling faults are categorized as rotor faults [1-3]. Specifically, induction motor faults can be broadly classified into bearing failures, stator faults, rotor faults, air gap eccentricity, mechanical vibrations, etc. Over the recent years, there has been increasing efforts dedicated in establishing AI based models to predict real-world time-dependent data. Various AI approaches such as the Artificial Neural Network (ANN), Adaptive Network based Fuzzy Inference System (ANFIS), Support Vector Machine (SVM), and Least Squares Support Vector Machine (LS-SVM) approaches have been applied to cope with time series prediction in various domains [4-6]. Support vector machines have been utilized as a popular algorithm realized from the machine learning [7]. The basic idea of Support Vector Machines (SVM) is to find a hyper-plane in an N-dimensional space (N number of features) that differentiates classifies the data points. In order to converts not separable problem to separable problem, SVM used functions called kernels which transform low dimensional data space to a higher dimensional space features. LS-SVM is a modification of the original SVM where the resulting optimization problem has half the number of parameters and the model is optimized by solving a linear system of equations instead of a quadratic programming [7-8].

In this paper, LS-SVM technique is proposed for fault diagnosis and classification of the short-circuit in the stator phases of an induction machine using information provided by the stator current signature. This monitoring system will provide information on the operation of the machine to the operators who operate it. It is also able to cause in severe cases a shutdown of the machine or to allow the production system to continue to operate in degraded mode in case of problems do not require an immediate shutdown. The rest of this paper is structured as follows. Sections 2, 3 and 4 give a short presentation of different types of stator winding faults, modelling of a healthy squirrel-cage induction motor and modelling with inter-turn short circuit in stator phase. Section 5 describes the LS-SVM approach followed by the proposed methodology in section 6. Section 7 is devoted to simulation results and conclusion of this study is conducted in the last section.

2. STATOR FAULTS

Stator winding faults in squirrel cage induction motor are generally due to insulation failure. The stator short-circuits faults occurs, when the stator windings get shorted. The different types of stator winding faults can be classified as [12-13] : **(a)** short circuit between two turns of same phase-called turn-to-turn fault, **(b)** Short circuit between two coils of same phase-called coil to coil fault, **(c)** Short circuit between turns of two phases called phase to phase fault, **(d)** Short circuit between turns of all three phases, **(e)** Short circuit between winding conductors and the stator core-called coil to ground fault, **(f)** open-circuit fault when winding gets break.

Different types of stator winding faults are shown in Figure. 1.

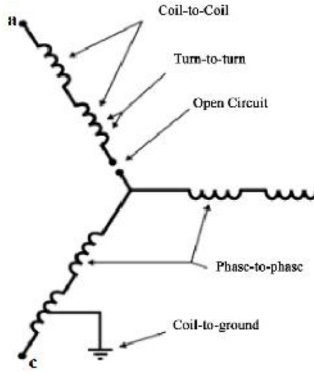


Figure1: Star-connected stator showing different types of stator winding fault [13].

3. MODELLING THE "HEALTHY" ASYNCHRONOUS MACHINE

The modelling of the induction machine is an essential and necessary phase for the various control applications, and also for the diagnosis and monitoring. The mathematical modelling allows observing and analysing different evolutions of its electrical and electromagnetic greatness [13].

3.1 Electricals equations:

The general expressions of the machine according to the flows and currents are:

$$\begin{cases} [V_{sabc}]=[R_s][i_{sabc}]+\frac{d}{dt}[\Psi_{sabc}] \\ [V_{rabc}]=[R_r][i_{rabc}]+\frac{d}{dt}[\Psi_{rabc}] \end{cases} \quad (1)$$

Where:

$$[V_{sabc}]=[V_{sa} \quad V_{sb} \quad V_{sc}]^T \qquad [V_{rabc}]=[0 \quad 0 \quad 0]^T$$

$$[i_{sabc}]=[i_{sa} \quad i_{sb} \quad i_{sc}]^T \qquad [i_{rabc}]=[i_{ra} \quad i_{rb} \quad i_{rc}]^T$$

$$[\Psi_{sabc}]=[\Psi_{sa} \quad \Psi_{sb} \quad \Psi_{sc}]^T \qquad [\Psi_{rabc}]=[\Psi_{ra} \quad \Psi_{rb} \quad \Psi_{rc}]^T$$

$$[R_s]=\begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \qquad [R_r]=\begin{bmatrix} R_r & 0 & 0 \\ 0 & R_r & 0 \\ 0 & 0 & R_r \end{bmatrix}$$

3.2 Torque equations

The electromagnetic torque is given by the following equation

$$T_{em}=\frac{1}{2}[i_{sabc}]^T \left\{\frac{d}{d\theta_s}[M_{sr}][i_{rabc}]\right\} \quad (2)$$

Where

θs is the angular position of the stator

$$[M_{sr}]=M\begin{bmatrix} \cos\theta_s & \cos\left(\theta_s+\frac{2\pi}{3}\right) & \cos\left(\theta_s-\frac{2\pi}{3}\right) \\ \cos\left(\theta_s-\frac{2\pi}{3}\right) & \cos\theta_s & \cos\left(\theta_s+\frac{2\pi}{3}\right) \\ \cos\left(\theta_s+\frac{2\pi}{3}\right) & \cos\left(\theta_s-\frac{2\pi}{3}\right) & \cos\theta_s \end{bmatrix}$$

4. MODELLING OF THE ASYNCHRONOUS MACHINE "IN THE PRESENCE OF THE DEFECT":

The type of defect that will be treated in this part concerns the short circuit between turns of the same phase.

To model this defect, we will assume that a number of turns "*n*" among those of phase "*a*" is short-circuited. This section of short-circuited turns is defined by the ratio "*cc*" between the number of shorted turns and the total number of turns of the phase "*a*", which is introduced into the mathematical model governing the operation of the machine. (Figure 2) shows the 3 stator windings with short circuit. Therefore the inductance and the resistance of the faulty phase changes and the mutual inductance between this phase and all other windings of the machine [13-14]

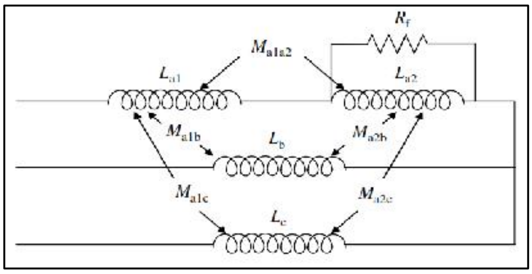


Figure 2 : Equivalent circuit of the stator winding with phase a inter-turn short circuit fault

The short-circuit fault in the stator winding causes a high current to flow in the short-circuited turns. This fault current can influence the currents of the other phases and produces a phase-to-earth and phase-to-phase short circuit, and subsequently leads to damage to the machine. [13-14]. Therefore, the detection of these defects is essential to avoid operation at dangerous conditions and reduce downtime. Modelling of the defective *IM* consists in introducing a resistance "R_f" in parallel with the short-circuited turns in the infected phase (Figure2).

A voltage will be induced in this short-circuit mesh, this induced voltage circulates a current in the short-circuited turns called fault current; the latter has a relationship of proportionality with the fault resistance and the induced voltage. The high resistance "R_f" corresponds to the case of the beginning of the deterioration of the insulation. The new form of the stator voltage equations in the natural basis frame is then rewritten as follows [14]:

$$\left[V_{sabc}\right]_f = \left[R_s\right]_f \left[i_{sabc}\right]_f + \frac{d}{dt}\left[\text{c}_{sabc}\right]_f \qquad (3)$$

Where:

$$\left[V_{sabc}\right]_f = \left[V_{sa} \quad V_{sb} \quad V_{sc} \quad 0\right]^T$$

$$\left[i_{sabc}\right]_f = \left[i_{sa} \quad i_{sb} \quad i_{sc} \quad i_f\right]^T$$

$$\left[\text{c}_{sabc}\right]_f = \left[\text{c}_{sa} \quad \text{c}_{sb} \quad \text{c}_{sc} \quad \text{c}_{sd}\right]^T$$

$$\left[R_s\right]_f = \begin{bmatrix} (R_s - R_f) & 0 & 0 & R_f \\ 0 & R_s & 0 & 0 \\ 0 & 0 & R_s & 0 \\ 0 & 0 & 0 & R_f \end{bmatrix} \qquad (4)$$

However, we keep the voltage equations of the rotor unchanged. If we denote by "cc" the fraction of the number of shorted turns of phase "*a*", then we have a healthy portion of fraction (*1-cc*) of turns and we assume phases "*b*" and "*c*" healthy. We will have the new matrix of the following stator inductors.

$$\left[L_{ss}\right] = \begin{bmatrix} L_s & M_{a1b}+M_{a2b} & M_{a1c}+M_{a2c} & -(L_{a2}+M_{a1a2}) \\ M_{a1b}+M_{a2b} & L_s & M_s & -M_{a2b} \\ M_{a1c}+M_{a2c} & M_s & L_s & -M_{a2c} \\ -(L_{a2}+M_{a1a2}) & -M_{a2b} & -M_{a2c} & L_{a2} \end{bmatrix}$$

$$\left[L_{ss}\right] = L_s \ diag \ \left[(1\text{-}cc) \ \ 1 \ \ 1 \ \ cc\right] +$$

$$M_s \begin{bmatrix} (1\text{-}cc)^2 & -\frac{1\text{-}cc}{2} & -\frac{1\text{-}cc}{2} & cc\,(1\text{-}cc) \\ -\frac{1\text{-}cc}{2} & 1 & -\frac{1}{2} & -\frac{cc}{2} \\ -\frac{1\text{-}cc}{2} & -\frac{1}{2} & 1 & -\frac{cc}{2} \\ cc\,(1\text{-}cc) & -\frac{cc}{2} & -\frac{cc}{2} & cc.^2 \end{bmatrix} \qquad (5)$$

Therefore, the matrix of mutual inductances becomes:

$$\left[M_{sr}\right] = \qquad (6)$$

$$M \begin{bmatrix} (1\text{-}cc)\cos\bar{\ }_s & (1\text{-}cc)\cos\left(\bar{\ }_s + \frac{2®}{3}\right) & (1\text{-}cc)\cos\left(\bar{\ }_s - \frac{2®}{3}\right) \\ \cos\left(\bar{\ }_s - \frac{2®}{3}\right) & \cos\bar{\ }_s & \cos\left(\bar{\ }_s + \frac{2®}{3}\right) \\ \cos\left(\bar{\ }_s + \frac{2®}{3}\right) & \cos\left(\bar{\ }_s - \frac{2®}{3}\right) & \cos\bar{\ }_s \\ cc\cos\bar{\ }_s & \cos\left(\bar{\ }_s + \frac{2®}{3}\right) & \cos\left(\bar{\ }_s - \frac{2®}{3}\right) \end{bmatrix}$$

Where:

L_s and M_s are the stator self and mutual inductances of the healthy machine.

L_{a2} is the self-inductance of the faulty winding a_{s2} (Figure 2). M_{a1b} and M_{a1c} are mutual inductances between a_{s2} and the windings b_s and c_s. In addition, M_{a1a2}, M_{a2b} and M_{a2c} are, respectively, the mutual inductances between a_{s2}, and the windings a_{s1}, b_s and c_s. [14].

5. LEAST SQUARES SUPPORT VECTOR MACHINES

A brief description of *LS-SVM* is presented in this section while a detailed explanation can be found in the reference [15]. The *LS-SVM* can solve the small-sample, nonlinear and high-dimensional problems, but when it is used to solve nonlinear problems, the selection of kernel function directly affects the final classification result [9-11].

Given *a* training set of *N* data points $\{ y_k , x_k \}_{k=1}^{N}$, where $x_k \in R^n$ is the *k-th* input pattern and $y_k \in R$ is the *k-th* output pattern, the classifier can be constructed using the support vector method in the form

$$y(x) = sign\left[\sum_{k=1}^{N} \sim_k y_k K(x, x_k) + b\right] \qquad (7)$$

Where $\sim_k$ called support values and *b* is *a* constant. *K*(*x*, *x*$_k$) indicates a kernel function whose value equals the inner product of *x* and *x*$_k$ vectors in the feature space *φ*(*x*) and *φ*(*x*$_k$) [10-11, 16]. The basic features of a kernel function are derived from Mercer's theorem. All function satisfies Mercer's condition is defined as kernel function. Typical examples of the kernel function are:

$K(x, x_k) = x_k^T x$ (linear).

$K(x, x_k) = (x_k^T x + 1)^d$ (Polynomial).

$K(x, x_k) = \tanh[s\, x_k^T x + \bar{\ }]$ (Multilayer perceptron).

$K(x, x_k) = exp\{-\|x - x_k\|_2^2 / ±^2\}$ (RBF).

Where:

T, *d*, *K*,¯ and σ are constants [9-10,16].

For example in case of two classes, the classifier is obtained as follows:

$$\begin{cases} w^T \ddot{\ } (x_k) + b \geq +1 & if \ y_k = +1 \\ w^T \ddot{\ } (x_k) + b \leq -1 & if \ y_k = -1 \end{cases}$$

This can also be written as

$$y_k[w^T\varphi(x_k)+b]\geq 1,\ k=1,...,N$$

Where $\varphi(\,)$ is a nonlinear function mapping of the input space to a higher dimensional space. *LS-SVM* classifiers\

$$\min_{w,b,e} J_{LS}(w,b,e)=\tfrac{1}{2}w^T w+\gamma\tfrac{1}{2}\sum_{k=1}^{N}e_k^2 \tag{8}$$

Subjects to the equality constraints

$$y_k[w^T\varphi(x_k)+b]=1-e_k,\ k=1,...,N$$

The Lagrangian is defined as

$$L(w,b,e;\alpha)=J_{LS}-\sum_{k=1}^{N}\alpha_k\left\{y_k[w^T\varphi(x_k)+b]-1+e_k\right\}$$

With Lagrange multipliers $\alpha_k\in R$ (called support values).

The conditions for optimality are given by

$$\begin{cases}\frac{\partial L}{\partial w}=0 & \rightarrow \quad w=\sum_{k=1}^{N}\alpha_k y_k\varphi(x_k)\\ \frac{\partial L}{\partial b}=0 & \rightarrow \quad \sum_{k=1}^{N}\alpha_k y_k=0\\ \frac{\partial L}{\partial e_k}=0 & \rightarrow \quad \alpha_k=\gamma e_k\\ \frac{\partial L}{\partial \alpha_k}=0 & \rightarrow \quad y_k[w^T\varphi(x_k)+b]-1+e_k=0\end{cases} \tag{9}$$

For $k=1,\ldots N$. After elimination of w and e one obtains the solution

$$\begin{bmatrix}0 & Y^T\\ Y & ZZ^T+\gamma^{-1}I\end{bmatrix}\begin{bmatrix}b\\ \alpha\end{bmatrix}=\begin{bmatrix}0\\ 1_v\end{bmatrix} \tag{10}$$

With

$$Z=[\varphi(x_1)^T y_1;...;\varphi(x_N)^T y_N],$$
$$Y=[y_1;...;y_N],1_v=[1;...;1],$$
$$e=[e_1;...;e_N]$$

and $\alpha=[\alpha_1;...;\alpha_N]$. Mercer's condition is applied to the matrix $\Omega=ZZ^T$ with

$$\Omega_{kl}=y_k y_l\varphi(x_k)^T\varphi(x_l)=y_k y_l K(x_k,x_l)$$

The kernel parameters, i.e. σ for *RBF* kernel, can be optimally chosen by optimizing an upper bound on the *VC* dimension. The support values αk are proportional to the errors at the data points in the *LS-SVM* case, while in the standard *SVM* case many support values are typically equal to zero. When solving large linear systems, it becomes needed to apply iterative methods [10].

The RBF kernel function is usually employed as the kernel function. In this study, RBF kernel was opted for *LS-SVM* inter-turn fault detection.

The *RBF* width σ^2 and regularization parameter γ, which affect *LS-SVM* generalization performance should be carefully selected [15-16].

6. METHODOLOGY

This section discusses the proposed methodology which includes feature extraction; Database selection and the proposed *LS-SVM* for stator winding short-circuit fault diagnosis.

6.1. Feature extraction

The proposed *LS-SVM* approach is trained and tested to identify the stator winding inter-turn short circuit fault. It evaluates the input stator current of the same phase and decides the motor condition as output by indicating the percentage of inter-turn short circuit fault occurred in the motor (Figure 3).
The input variable (*Max*(i_{sa})) of the *LS-SVM* algorithm represents the values of the maximum amplitude of the stator current $i_{sa}(t)$ in different working conditions of the motor. The output variable takes five values describing the indication of the short-circuits fault:

- 0: Healthy motor.
- 1: 10% shorted turns.
- 2: 20% shorted turns.
- 3: 30% shorted turns.
- 5: 50% shorted turns.

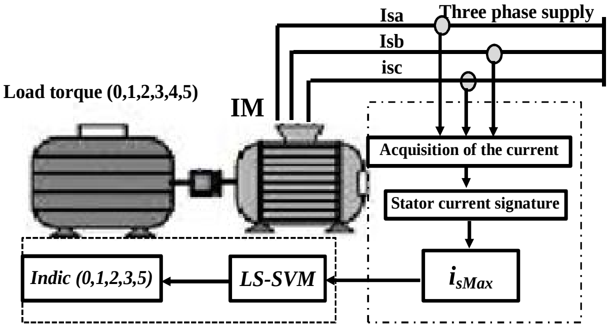


Figure 3: Block diagram of LS-SVM for stator winding short-circuit fault diagnosis

6.2. Database selection

A database constituted by inputs and output data sets has been applied to train and test the *LS-SVM* approach, the inputs-outputs data are collected through simulations in *Matlab* environment. The data set utilized derived from simulation are composed of *60* samples. The training set is composed of *30* samples representing the maximum amplitude of the stator current i_{saMax} under different load conditions $T_L=0$, 1, 2, 3, 4 and *5 Nm.* The test set is composed of 30 samples representing the maximum amplitude of the stator current i_{saMax} under different load conditions $T_L=0.5$, *1.5, 2.5, 3.5, 4.5* and *5.5Nm* are used to test its performance. Each pattern of the training and testing set comprises one input stator current signature i_{saMax} and one output which represent the indication of severity of the short-circuits fault (*Indic*).

It is noted that, the induction machine is simulated in open-loop and the LS-SVM detection method is used to evaluate the input stator current (ias) and decides the motor condition as output by indicating the severity of the short-circuits fault.

Table 1: Database for training and testing

cc %	Training set		Testing set		
	T_L	*Input Max(ias)*	T_L	*Input Max(ias)*	*Indication (indic)*
0%	0	3,5574	0.5	3,5602	0
0%	1	3,5775	1.5	3,6086	0
0%	2	3,6537	2.5	3,7123	0
0%	3	3,7841	3.5	3,8684	0
0%	4	3,9647	4.5	4,0722	0
0%	5	4,1903	5.5	4,3184	0
10%	0	9,5708	0.5	9,7547	1
10%	1	9,9494	1.5	10,1347	1
10%	2	10,3357	2.5	10,5335	1
10%	3	10,7400	3.5	10,9522	1
10%	4	11,1699	4.5	11,3933	1
10%	5	11,6228	5.5	11,8587	1
20%	0	18,0303	0.5	18,2184	2
20%	1	18,4315	1.5	18,6132	2
20%	2	18,8375	2.5	19,0337	2
20%	3	19,2530	3.5	19,4834	2
20%	4	19,7235	4.5	19,9746	2
20%	5	20,2383	5.5	20,5166	2
30%	0	26,7037	0.5	26,8866	3
30%	1	27,1105	1.5	27,2743	3
30%	2	27,5216	2.5	27,6961	3
30%	3	27,9371	3.5	28,1595	3
30%	4	28,4147	4.5	28,6888	3
30%	5	28,9864	5.5	29,3148	3
50%	0	44,1895	0.5	44,3518	5
50%	1	44,6003	1.5	44,709	5
50%	2	45,0148	2.5	45,1097	5
50%	3	45,4332	3.5	45,5751	5
50%	4	45,8556	4.5	46,1498	5
50%	5	46,5135	5.5	46,9782	5

6.3. Fault detection of stator using LS-SVM

As mentioned above, the database of learning inputs / outputs which based on the simulation results derived from the motor behaviour without and with faults is used to train the proposed approach and to test its performance.

Once the data is grouped, we present them as input to the *LS-SVM* approach, the system performs its learning so that it can be ready to predict the severity of the short-circuit fault. We perform a test to validate the performance of the system by presenting as input the new data that are not part of the learning base. Once the test has been successfully completed (acceptable prediction error), the system is ready to classifier the severity of the short-circuit fault. In this paper, the parameters of LS-SVM approach have been selected after several tests. Therefore, the parameters adopted for this study are: γ=1000, σ^2=0.01.

7. RESULTS AND DISCUSSION

7.1. Healthy motor operation:

The simulation of the model of the healthy machine made it possible to draw the curves of the electromechanical quantities (stator current, torque and speed) with an introduction of a resistant pair of *Tr = 3 Nm* at the moment *0.7s*

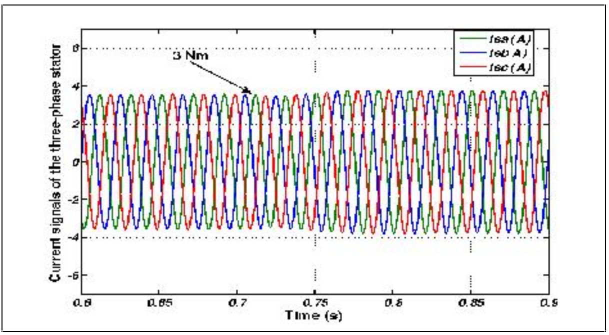


Figure.4: Stator current of the healthy motor under load (T_L = *3 Nm at t=0.7s*)

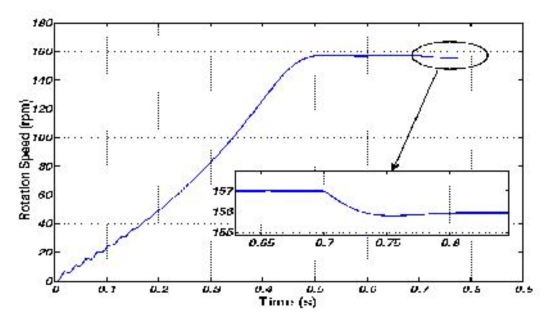


Figure 5 : Speed of rotation of the healthy motor under load (T_L = *3 Nm at t=0.7s*)

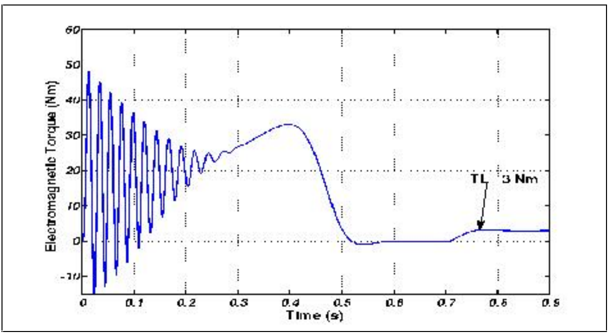


Figure 6 : Electromagnetic torque of the healthy motor under load (T_L = *3 Nm at t=0.7s*)

7.2. Operation with short circuit fault 20%:

We will now present the simulation results for an operation of the *IM* with short-circuit fault between stator turns, the degree of the short circuit is *20%* at the instant of *0.9s*.

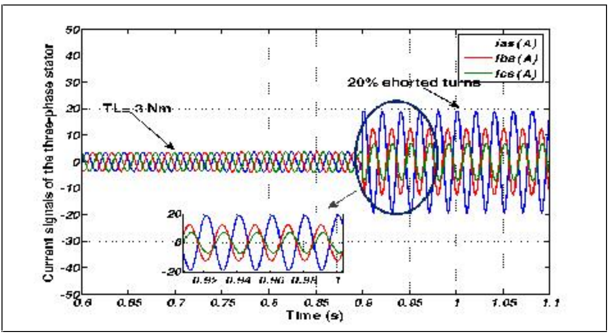


Figure 7 : Stator current motor with short circuit (*20% at t=0.9s*) under load (T_L = *3 Nm at t=0.7s*)

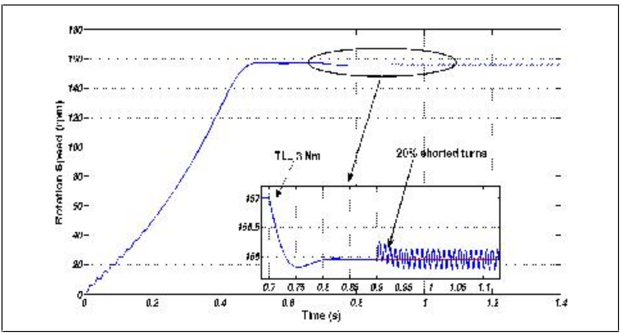
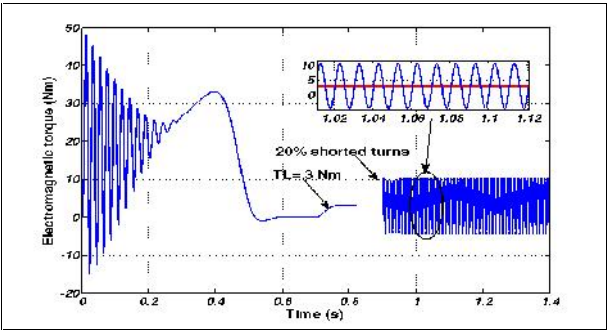


Figure 8 : Speed of rotation with short circuit (*20% at t=0.9s*) under load (T_L = *3 Nm at t=0.7s*)



. Figure 9 : Electromagnetic torque with short circuit (*20% at t=0.9s*) under load (T_L = *3 Nm at t=0.7s*)

In the initial stage, until *t=0.9s*, the short circuit leve is set to zero, representing a healthy *IM* without any faults. A load torque equal to *3Nm* is applied at *0.7s*. A default is applied at *t=0.9s* (*20%*). To test the severity of the fault. The fault impact appears on the stator currents (Figure 7), the speed of rotation (Figure 8) the electromagnetic torque (Figure 9), with increasing oscillations.

7.3. Influence of short circuit fault on the stator current

In what follows we present the simulation of the operation of the motor with short circuit of the turns of a coil with (10%, 20%, 30% and 50%) for each load (from 0 Nm to 5 Nm), to record the maximum values of the stator current

It can be revealed that, with the increase in the defect ratio of the affected phase (as) and with the increase of the load torque, the amplitude (or the max value) of the stator phase current i_{saMax} increases.

7.4. Results from the LS-SVM approach

As it has been descripted in section 6, the proposed monitoring methodology is developed to detect the stator winding inter-turn short circuit fault. The simulation results of proposed approach are shown in figures 11 and 12 respectively, in which *30* data are used to train the model and other *30* data are used to test its performance (Table 1).

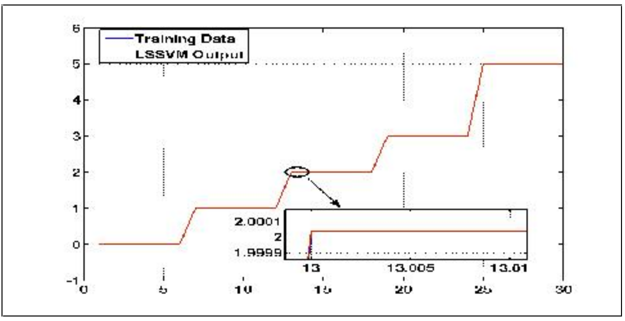


Figure 10 : Variation of degree of defect according to vector number for the training set.

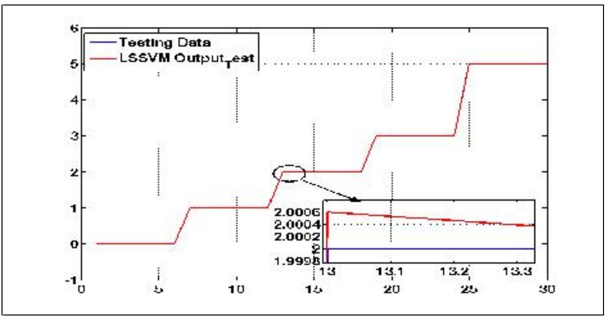
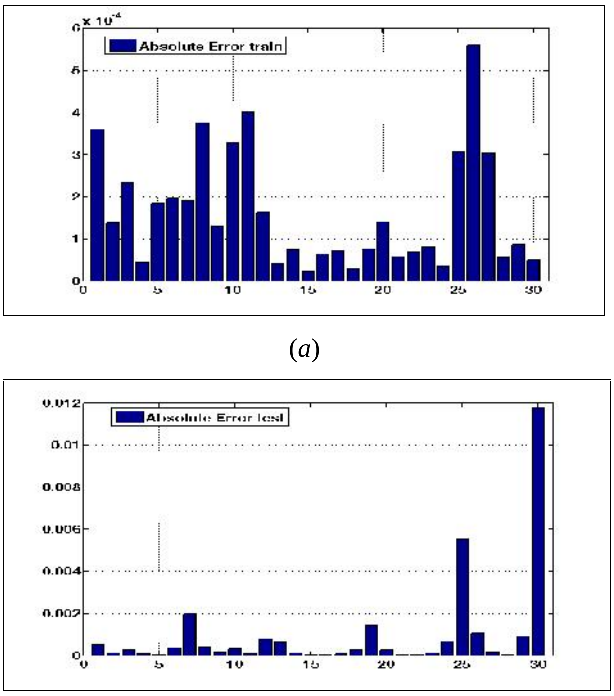


Figure 11: Variation of degree of defect according to vector number for testing set.

From Figure 10 and 11, it is very clear that the *LS-SVM* algorithm gives values that are almost identical (very good adaptation) to those desired (targets). In addition, Figure 12 (a and b) displays clearly that the errors corresponding to the training and the test are very low with higher values of absolute error of 0.0118 for testing stage.



(*a*)

(*b*)

Figure 12 : Curves of : a) absolute error training

b) absolute error testing

8. CONCLUSION

In this paper, a *LS-SVM* approach has been proposed as monitoring system to detect stator winding inter-turn short circuit fault of the induction motor, in which the maximum amplitude of the stator current under different load conditions is used as input variable from motor. Due to its strong generalization capability, *RBF* kernel function is used in order to augment the generalization performance of *LS-SVM* for classification task. The simulation shows very good adaptation based on *LS-SVM* to the database used for the operation of training and testing. The proposed method could also be applied for other fault examination.

Machine settings

Rs=2.89 [Ω] Strength of a stator phase

P=2 Number of pole pairs

J=0.007 [Kg.m2] Moment of inertia

Nr=28 Number of rotor bars

Ns=464 Number of turns

Ls=0.341 [H] Leakage inductance of a stator phase

Lr=0.344 [H] Leakage inductance of a stator phase

Un: 220/380 [V] Nominal voltage

In: 4,3/2,5 [A]. Rated current

Nr=1425 [tr/mn]. Rated speed

Pn=11 [kW]. Nominal power

Cn= 7 [N.m] Rated torque

REFERENCES

SINGH G.K and Al Kazzaz S.A.S, 2003 Induction Machine Drive Condition Monitoring and Diagnostic Research-A Survey, Electric Power Systems Research, vol. 64, pp. 145-158.

Sin M.L, Soong W.L and Ertugrul N. 2003 Induction Machine On-Line Condition Monitoring and Fault Diagnosis A Survey, AUPEC2003, Australasian Universities Power Engineering Conference, Christchurch, New Zealand, pp. 1-6.

Bhowmik S.P, Pradhan S and Prakash M, 2013 Fault diagnostic and monitoring methods Of induction motor: a review, International Journal of Applied Control, Electrical and Electronics Engineering. (IJACEEE) Volume 1, Number 1.

Bonnet A.H. and Soukup G.C. 1992 Cause and Analysis of Stator and Rotor Failures in Three Phase Squirrel Cage Induction Motors, IEEE Transactions on Industry Application, vol. 28, no. 4, pp. 921-937.

Shi Z.J and Li G. 1998 Fine tuning support vector machines for short-term wind speed forecasting, Energy Conversion and Management, vol. 52, no. 4, pp.

Yan W, Qiu H, and Xue Y, 2009 Gaussian process for long-term time-series forecasting, in Proceedings of the International Joint Conference on Neural Networks (IJCNN '09), pp. 3420–3427.

Melin P, Soto J, Castillo O and Soria J. 2012 A new approach for time series prediction using ensembles of ANFIS models, Expert Systems with Applications, vol. 39, no. 3, pp. 3494–3506.

Suykens J, Gestel J. V, Brabanter J. D, Moor B. D and Vandewalle J.2002. Least Square Support Vector Machines, World Scientific Publishers, Singapore,

Christianini. N. and Taylor J, 2003 Support Vector Machine and Other Kernel Learning Methods. London: Cambridge University Press.

Dubois J. P and Abdul-Latif O. M, 2005 SVM-Based Detection of SAR Images in Partially Developed Speckle Noise, World Academy of Science, Engineering and Technology.

Mahdjoubi A, Zegnini B, Belkheiri M,2019 Prediction of critical flashover voltage of pollutedInsulators under sec and rain conditions using least Squares support vector machines (LS-SVM), DIAGNOSTYKA, Vol. 20, No. ..

Jannati M, Idris N.R.N and Salam Z, 2012 A new method for modeling and vector control of unbalanced induction motors, In Energy Conversion Congress and Exposition (ECCE), pp. 3625–3632.

Karmakar S, Chattopadhyay S, Mitra M, 2016 Induction motor fault diagnosis approach through current signature analysis. 1st ed. Singapore: Springer, ISBN 978-981-10-0623-1

Vaseghi B, Takorabet N, Meibody-Tabar F, 2010 Stator turn fault study and parameter extraction of induction machines using FEM, COMPEL International Journal of Computations and Mathematics in Electrical 29(3):885-899.

Bessedik S. A and Hadi H, 2013. Prediction of flashover voltage of insulators using least squares support vector machine with particle swarm optimisation, Electric Power Systems Research, vol.104, pp. 87-92.

Bessedik S. A, Djekidel R and Ameur A, 2018. Performance of different kernel functions for LS-SVM-Grey Wolf Optimiser to estimate flashover voltage of polluted insulators, IET Science, Measurement & Technology, vol. 12, no. 6, pp. 739-745.

TIME SERIES MODEL IMPROVING WITH AUTOMATIC SAVITZKY-GOLAY FILTER FOR REMAINING USEFUL LIFE ESTIMATION.

Youssouf Diaf [(a)], **Samir Benmoussa** [(b)], **Mohand Djeziri** [(c)]

[(a)] Agence Spatiale Algérienne, Bouzareah Alger, 16000 Algeria.
[(b)] Laboratoire d'Automatique et de Signaux de Annaba (LASA), University Badji Mokhtar Annaba, 23000 Algeria.
[(c)] Laboratoire des Sciences de l'Information et des Systemes (LSIS), UMR CNRS 7296 France.

[(a)]ydiaf@asal.dz, [(b)] benmous2a.samir@gmail.com, [(c)]mohand.djeziri@lis-lab.fr

ABSTRACT

The performance of fault diagnosis and failure prognosis methods is directly related to the quality of sensor measurements. When the systems evolve under extreme conditions of use as in the aeronautical field, the sensors measure are often impacted by strong noise and disturbances. This paper deals with the using of an automatic Savitzky-Golay filter to smooth the sensors data for time series model training and remaining useful life estimation. A proposed window value calculation for the Savitzky-Golay filter is used to improve signal quality. The NARX neuronal network is used as a data-driven approach to model the trend of the smoothed time series data. The proposed approach is successfully applied for degradation-trend modeling and remaining useful life prediction in turbo engines. These systems are considered as a high noise device which mean more challenge in data processing. The results show an improvement in the prediction of the remaining useful life compared to previous works.

Keywords: Remaining Useful Life, Savitzky-Golay filter, NARX neuronal network, smoothing, Gas Turbine Engines

1. INTRODUCTION

Nowadays, competitiveness and the race for performance are increasingly encouraging scientists and industrialists to replace preventive and corrective maintenance strategies with predictive and conditional ones. The deployment of these new maintenance strategies requires an understanding of the current health state of the systems and the future evolution of their degradation process. This knowledge is provided by fault diagnosis and failure prognostic methods (Vichare and Pecht. 2006).

Recently, the common for all the prognostic and diagnostic methods is prognostic and health management. The prognostic and health management (PHM) predict the state of health of equipment or system which improve our knowledge on it and provide a strategy to manage the best CBM maintenance actions. The PHM is based on either physical model, data-driven model or both. Remaining Useful Life (RUL) of the system is one of the key outputs of the methods of failure prognosis

Physical models are not available in most practical cases of industrial systems, because it is very difficult to make modeling assumptions, and formalize the cause-and-effect relationships when the interaction of components and physical phenomena are complex, strongly non-linear and sometimes unknown.

Data-driven failure prognostic methods are the most used in practical cases, and are becoming more and more efficient thanks to developments in data storage and analysis tools and artificial intelligence. The data-driven methods are based on data collected manually or remotely from the system. The data-driven methods create some new challenging problems for scientists, which is sensors.

Actually, in PHM the sensors technology offers more and more accurate measurements and less space to be implemented. These advantages in PHM applications are faced by the problems of how much we need indicators to accurately estimate the state of health and if we have the right indicators, how to prevent the noise effect.

In the most of application data is certainly contaminated with noise, particularly in a harsh environment (high pressure, temperature, vibration ….), where noise becomes very intense and random. In PHM the most comment method used to avoid the noise is smoothing. The smooth methods are multiple and every method has its advantages and disadvantages.

2. SMOOTHING IN DATA-DRIVEN APPROACHES

Recent works on data-driven prognostic algorithms are gating more and more accuracy in damage prognostic. The data-driven methods such neuronal network (Heimes 2008, javed and al. 2011, Bektas and Jones

2016), support vector machine (Bluvband and Porotsky 2016, Schölkopf and Smola 2001, Michael 2001), similarity-based prognostic (Tianyi and al. 2008) and others (Djeziri and Benmoussa and Sanchez 2018, Benmoussa and Djeziri 2017) are used to predict the RUL.

Smoothing take an essential part before any training process in prognostic. In (Bluvband and Porotsky 2016), the authors used a fitting technic to remove the noise. They use a monotonic fitting by non-linear regression methods, which used two types of smoothed functions the polynomial functions and the exponential functions. Because the smoothing function didn't guarantee permanent stability, they propose the selection of the right function for each sensor. In (Bektas and Jones 2016), a 9th and a 4th degree polynomial regression was used to filter the data and each polynomial degree is used either for training or testing. (Coble and Hines 2012) propose for reducing the noise a quadratic fitting technic.

In data analysis aspects, using these methods can sometimes lead to an undershoots and a loss of critical information (peaks). Additionally, the performance depends on the appropriate selection of the polynomial order or function which is difficult.

In this work we tend to improve RUL estimation by an automatic smoothing process. An automatic Savitzky-Golay filter (Abraham and Golay 1964, Dombi and Dineva 2018, Zhu and al. 2017) is proposed.

3. METHOD

3.1. Savitzky-Golay Filter

The core of Savitzky-Golay (SG) algorithm is fitting a low degree polynomial in least squares sense on the samples within a sliding window (Abraham and Golay 1964; Steinier, Termonia and Deltour 1972; Dombi and Dineva 2018; Zhu and al. 2017). The main advantage of the SG-filter in contrast to the classical filters which require the characterization and model of the noise process, is that both the smoothed signal and the derivatives can be obtained by a simple and fast calculation.

According to (Abraham and Golay 1964; Steinier, Termonia and Deltour 1972; Dombi and Dineva 2018), The SG filter formalism is given by (1). The input data x_j, y_j is assumed to be equally spaced were $j = 1, \ldots., n$ with n the amount of data.

$$g_i = \sum_{i=-m}^{m} c_i \, y_k + i \qquad (1)$$

where the window length $M = 2m + 1$, $i = -m, \ldots, \lambda, \ldots, m$ and λ denotes the index of the center point. A table of SG coefficients is obtained using $M = 2m + 1$ as a window length and k denotes the polynomial degree (Abraham and Golay 1964; Steinier, Termonia and Deltour 1972; Dombi and Dineva 2018).

Usually the Savitzky-Golay filters perform well by using a low order polynomial with long window length or a low degree with a short window. Nonetheless, it is possible to further improve the efficiency with an adaptive smoothing approach based on the classic SG filtering technique that ensures acceptable performance independent of the type of noise process (Dombi and Dineva 2018).

The contribution made in this work tend to automatically calculate the sliding window M value which conserve only the appearance of the signal by annulling noise and outlier's components. The polynomial order is fixed as 2 to avoid any intense wiggling in the smoothed signal. The proposed calculation of the M value is based on the similarity between Kernel estimation and SG filter.

As mentioned in (Bowman and Azzalini 1997), for kernel smoothing regression in case of a normal data vector y the kernel estimator is written as in (2).

$$\tilde{f}(y) = \frac{1}{n} \sum_{i}^{n} w\,(y - y_i; h) \qquad (2)$$

Where $\tilde{f}$ the kernel estimator, n the amount of data y and w the kernel function whose variance is controlled by the parameter h. h is called the smoothing function or the bandwidth.

In order to smooth $\tilde{f}$, the optimum formula to calculate h is given by (3).

$$h = \left(\frac{4}{3n}\right)^{1/5} \sigma \qquad (3)$$

where σ denotes the standard deviation of the data. The assumption of normality can cause problems of oversmoothing when dealing with non-normal data. To reduce the oversmoothing σ must be adjusted as in equation (4).

$$\tilde{\sigma} = \frac{median\{|y_i - \tilde{\mu}|\}}{0.6745} \qquad (4)$$

Where $\tilde{\mu}$ denotes the median of the sample.

In the kernel smoothing regression, the kernel function and h are chosen which is similar to the SG filter where the degree polynomial and the window M must be chosen. This broad analogy between the kernel smoothing regression and the SG filter in the sense that the value of h and M in both methods is reacting the same. For the kernel smoothing regression or the SG filter decreasing M or h makes the data smoother, while decreasing them makes the estimation wiggly. The choose of M is driven by this close similarities between the two methods. The value of M is calculated as in equations (3) and (4).

3.2. Data Sets

The data used for this work is from the Prognostics Data Repository, which is a collection of data sets that have been donated by various universities, agencies, or companies. Mostly these are time series data from some nominal state to a failed state (NASA s.d.).

The data is an engine degradation simulation that was carried out using C-MAPSS (Commercial Modular Aero-Propulsion System Simulation). Four different sets are simulated under different combinations of operational conditions and fault modes. The data set was provided by the Prognostics CoE at NASA Ames (Saxena and Goebel 2008).

The C-MAPSS data set consists of multiple multivariate time series which is divided into training and test subsets. Each subset contains a time series data of 100 engines (a fleet of engines of the same type). In each cycle the data record: the engine number, the cycle number, three operational setting and 21 sensors measurement. The measured data is contaminated with high sensor noise (Saxena, Goebel, Simon and al. 2008).

Our objective is to enhance the prediction of the remaining operational cycles and to provide the vector of true Remaining Useful Life (RUL) values for the test data using only the sensor data. Only the first set of data will be used to test the proposed approach.

3.3. Data-Driven Method

Regarding it successful use in many works, the nonlinear autoregressive network with exogenous inputs (NARX) is used in this work to estimate directly the RUL value from data. The NARX network was successfully applied to estimate the RUL value from operation data in various applications (Bektas and Jones 2016; Rai and Upadhyay 2017; Santoso, Prahasto and Widodo 2013).

NARX is a recurrent dynamic network which has exogenous inputs with feedback connections enclosing several layers of the network (Billings 2013). As shown in figure 1, NARX model is based on the linear autoregressive network model, which is commonly used in time-series modeling. The NARX model is defined by equation (5).

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-n\_y), u(t-1), \dots, u(t-n\_u)) \qquad \mathbf{(5)}$$

where the next value of the dependent output signal $y(t)$ is regressed on previous values of the output signal and previous values of an independent (exogenous) input signal. The NARX network can predict next value for an input signal, filtering nonlinear noise or model a nonlinear dynamic system.

As mentioned in (Bektas and Jones 2016; Narendra and Parthasarathy 1991), the best training processes use the true output instead of feeding back the estimated output, figure 2. This training process enhance the output accuracy of the network and the training is simply performed by static backpropagation. More details on the NARX model used in this work is available in (Bektas and Jones 2016; Billings 2013).

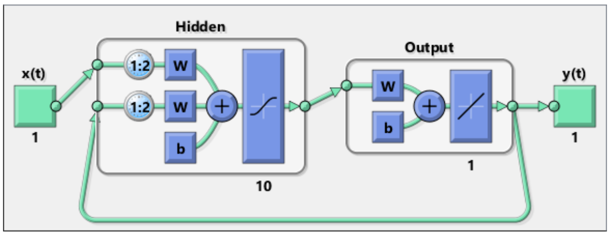


Figure 1: A MATLAB representation of NARX network, where X(t) is the network inputs (measured data) and Y(t) is the network output.

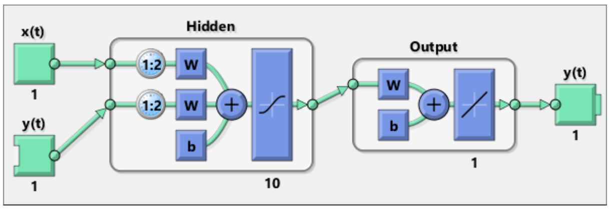


Figure 2: training the NARX network in MATLAB where x(t) is the training data inputs y(t) is the training targets.

4. RESULTS AND DISCUSSION

All the simulations are done using MATLAB R2016a. The first simulation part is testing the proposed SG filter on some signals and compare the smoothed signal with the kernel smoothing regression. Figure 3 shows the application of the proposed SG filter Vs a kernel smoothing regression on benchmark signal (Cichocki and Amari 2003). the filtered signal with SG filter offers more smoothness for the original signal. In figure 4 the signal used is from C-MAPSS data set (Saxena, Goebel, Simon and al. 2008). The resultants signal is very smooth compared to the kernel smoothing regression.

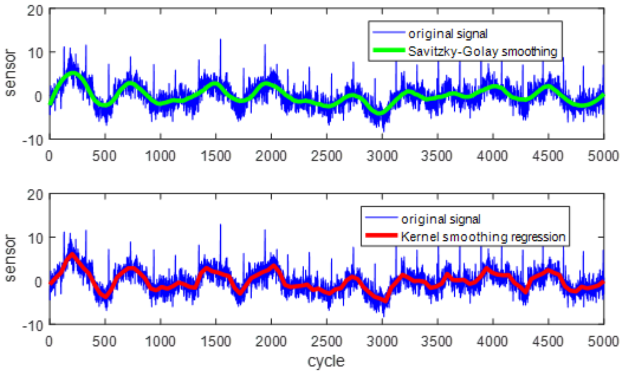


Figure 3:shows the application of the proposed SG filter Vs a kernel smoothing regression on benchmark signal (Cichocki and Amari 2003).

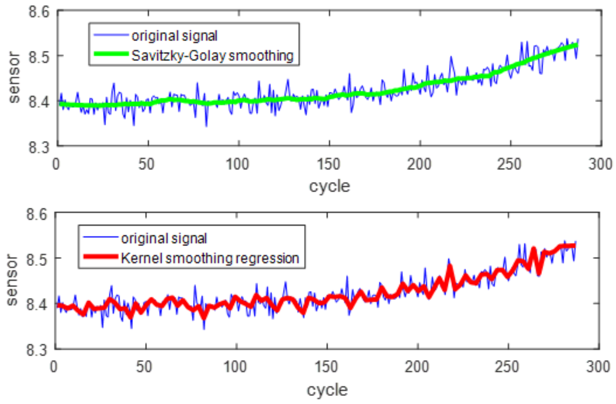


Figure 4:shows the application of the proposed SG filter Vs a kernel smoothing regression on C-MAPSS data set (Saxena, Goebel, Simon and al. 2008).

The simulation steps are:

1. Loading the training data file
2. Choose the best indicators for RUL prediction based on data trendability (Al-Dahidi and al. 2016).
3. Smooth the sensor data with Savitzky-Golay filter.
4. Train multiple NARX networks architecture on open loop (Bektas and Jones 2016).
5. Chose the best network based on the means of training prognostic metrics (Saxena, Celaya and al. 2008).
6. Load the test data.
7. Smooth the test data.
8. Calculate the RUL vector of the test data.

Figures 5 show clearly the performance of the SG-filter on a monotonic indicator of the first engine in the first sets of training data. It can be observed that the smoothing is very good.
The automatic window calculation gives a smoothing which conserve only the appearance of the signal and shows clearly the monocity of data series.

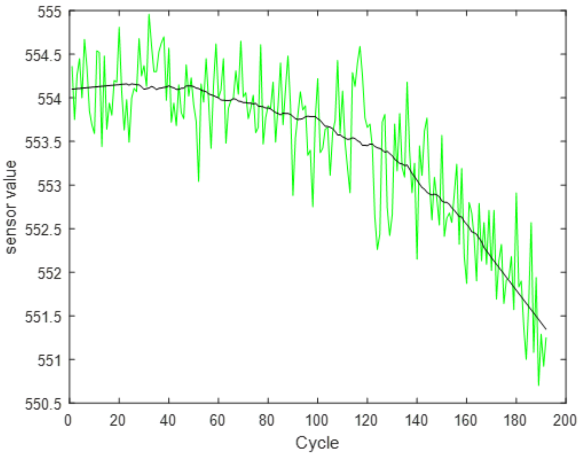


Figure 5: Performance of the SG filter on sensor signal C-MAPSS data (Saxena, Goebel, Simon and al. 2008). clear chart: original sensor signal - black line: smoothed signal.

Prognostic metrics were calculated for evaluation, table 1 contains metric results for test data. The metrics are summarized in detail in (Saxena, Celaya and al. 2008). The results show an excellent prediction for the NARX model when trained with filtered data from SG method. Table 2 shows the twenty-first RUL predicted compared to (Bektas and Jones 2016). It's noticed the superiority of the proposed approach compared to the previous work.

Table 1: Summary of result on test data using (Saxena, Celaya and al. 2008) metrics

Metrics	Results
Score	205
FPR (%)	48
FNR (%)	52
MSE	37
MAE	4.4
MAPE (%)	12.6
Accuracy (%)	79

Table 2: Results of estimated RUL compared to (Bektas and Jones 2016) for the 20 first test engines

Test engine N°	Our work	[4] work	True RUL
1	110	116	112
2	94	112	98
3	64	43	69
4	80	79	82
5	92	88	91
6	87	111	93
7	95	93	91
8	93	107	95
9	107	118	111
10	104	93	96
11	91	85	97
12	131	78	124
13	102	84	95
14	106	98	107
15	80	97	83
16	86	100	84
17	53	52	50
18	23	39	28
19	84	113	87
20	15	26	16

5. CONCLUSION

An approach based on an automatic SG filter and NARX model is introduced in this work. The combination of a powerful data-drive model such the NARX model, applied successfully in many applications, with a high-performance filter with automatic parameters lead to excellent results in modeling health state of a system in PHM application.

The proposed approach is validated using the first data set of C-MAPSS data for gas turbo engine and compared to previous works using the same model. The result shows a considerable improvement on RUL prediction accuracy that indicates the importance of selecting the proper filtering process before training.

The challenge is to adopt this approach with the other data sets of C-MAPSS.

REFERENCES

Abraham, Savitzky, and Marcel J. E. Golay. 1964. "Smoothing and Differentiation of Data by Simplified Least Squares Procedures." Analytical Chemistry (American Chemical Society) 36 (8): 1627–1639. doi:10.1021/ac60214a047.

Al-Dahidi, Sameer, Francesco Di Maio, Piero Baraldi, and Enrico Zio. 2016. "Remaining useful life estimation in heterogeneous fleets working under variable operating conditions." Reliability Engineering & System Safety 156: 109–124.

B., Scholkopf, and Smola A. J. 2001. Learning with kernels:support vector machines, regularization, optimization, and beyond. The MIT Press.

Bektas, Oguz, and Jeffrey A. Jones. 2016. "NARX Time Series Model for Remaining Useful Life Estimation of Gas Turbine Engines." EUROPEAN CONFERENCE OF THE PROGNOSTICS AND HEALTH MANAGEMENT SOCIETY 2016.

Benmoussa, Samir, and Mohand Arab Djeziri. 2017. "Remaining useful life estimation without needing for prior knowledge of the degradation features." IET Science, Measurement & Technology 11 (8): 1071 - 1078.

Billings, Stephen A. 2013. Nonlinear System Identification: NARMAX Methods in the Time, Frequency and Spatio-Temporal Domains. Wiley.

Bluvband, Zigmund, and Sergey Porotsky. 2016. Prognostic and Health Management: God Is in Details.

Bowman, Adrian W., and Adelchi Azzalini. 1997. Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations . New York: CLARENDON PRESS OXFORD.

Cichocki, Andrzej, Shun-ichi Amari, Krzysztof Siwek, and Toshihisa Tanaka and al. n.d. http://www.bsp.brain.riken.jp/ICALAB. ICALAB Toolboxes.

Coble, Jamie, and J. Wesley Hines. 2012. "Identifying Suitable Degradation Parameters for Individual-Based Prognostics." In Diagnostics and Prognostics of Engineering Systems_ Methods and Techniques, 145. IGI Global.

Djeziri, Mohand Arab, Samir Benmoussa, and Roberto Sanchez. 2018. "Hybrid method for remaining useful life prediction in wind turbine systems." Renewable Energy 116: 173 - 187.

Dombi, József, and Adrienn Dineva. 2018. "Adaptive Multi-round Smoothing Based on the Savitzky-Golay Filter." International Workshop Soft Computing Applications. 446-454.

Dombi, József, and Adrienn Dineva. 2018. "Adaptive Multi-round Smoothing Based on the Savitzky-Golay Filter." In Advances in Intelligent Systems and Computing, by József Dombi and Adrienn Dineva, 446-454.

Heimes, Felix O. 2008. "Recurrent neural networks for remaining useful life estimation." Proceedings of the Annual Conference on Prognostics and Health Management Society.

javed, kamran, rafael gouriveau, ryad zemouri, and noureddine zerhouni. 2011. "Improving data-driven prognostics by assessing predictability of features." Proceedings of the Annual Conference of the Prognostics and Health Management Society.

K., Javed, Gouriveau R., Zemouri R., and Zerhouni N. 2011. "Improving data-driven prognostics by assessing predictability of features." Proceedings of the Annual Conference of the Prognostics and Health Management Society.

M, Tipping. 2001. "Sparse Bayesian learning and the Relevance Vector Machine." Journal of Machine Learning Research 1 (3): 211–244.

Michael, E. Tipping. 2001. "Sparse Bayesian learning and the Relevance Vector Machine." Journal of Machine Learning Research 1 (3): 211–244.

Narendra, Kumpati S., and Kannan Parthasarathy. 1991. "Learning Automata Approach to Hierarchical Multiobjective Analysis." IEEE Transactions on Systems, Man and Cybernetics 20 (1): 263–272.

NASA. n.d. prognostic-data-repository. NASA. https://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/.

O., Heimes F. 2008. "Recurrent neural networks for remaining useful life estimation." Proceedings of the Annual Conference on Prognostics and Health Management Society.

Rai, Akhand, and S.H. Upadhyay. 2017. "The use of MD-CUMSUM and NARX neural network for anticipating the remaining useful life of bearings." Measurement 111: 397-410.

Santoso, Gunawan Budi, Toni Prahasto, and Achmad Widodo. 2013. "Prognosis of bearing damage performance to industrial system using nonlinear autoregressive with exogenous (NARX)." 2013 International Conference on QiR.

Saxena, Abhinav, and Kai Goebel. 2008. Turbofan Engine Degradation Simulation Data Set. Edited by NASA Ames Research Center. Moffett Field, CA. http://ti.arc.nasa.gov/project/prognostic-data-repository.

Saxena, Abhinav, Jose Celaya, Edward Balaban, Kai Goebel, Bhaskar Saha, Sankalita Saha, and Mark Schwabacher. 2008. "Metrics for evaluating performance of prognostic techniques." 2008 International Conference on Prognostics and Health Management. 1-17.

Saxena, Abhinav, Kai Goebel, Don Simon, and Neil Eklund. 2008. "Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation." Proceedings of the Ist International Conference on Prognostics and Health Management (PHM08). Denver CO.

Schölkopf, Bernhard, and Alexander J. Smola. 2001. Learning with kernels:support vector machines, regularization, optimization, and beyond. The MIT Press.

Steinier, Jean, Yves Termonia, and Jules Deltour. 1972. "Smoothing and differentiation of data by simplified least square procedure." Analytical Chemistry 44 (11): 1906-1909. doi:10.1021/ac60319a045.

T., Wang, Yu J., Siegel D., and Lee J. 2008. "A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems." Proceedings of the Annual Conference on Prognostics and Health Management Society.

Tianyi, Wang, Jianbo Yu, David Siegel, and Jay Lee. 2008. "A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems." Proceedings of the Annual Conference on Prognostics and Health Management Society.

Vichare, Nikhil M., and Michael G. Pecht. 2006. "Prognostics and Health Management of Electronics." IEEE Transactions on Components and Packaging Technologies 29 (1): 222-229.

Zhu, Gao Feng, Hong Qiu Zhu, Chun Hua Yang, and Wei Hua Gui. 2017. "Improved Savitzky-Golay filtering algorithm for measuring a pharmaceutical vial's oxygen content based on wavelength modulation spectroscopy." Journal of Optical Technology 84 (5): 355.

BOND GRAPHS BASED FORMATION CONTROL OF HOLONOMIC ROBOTS

Matías Nacusse [(a)], **Martín Crespo** [(a,b)], **Sergio Junco** [(a)]

[(a)] Laboratorio de Automatización y Control (LAC), Departamento de Control, FCEIA, UNR. Rosario, Argentina
[(b)] CONICET: Consejo Nacional de Investigaciones Científicas y Técnicas. Argentina

[(a)]nacusse@fceia.unr.edu.ar, crespom@fceia.unr.edu.ar, sjunco@fceia.unr.edu.ar

ABSTRACT

This paper tackles the problem of formation control for a group of holonomic vehicles using the Bond Graphs formalism. The control law design follows an energy based approach in which the agents are connected each other by means of virtual springs and dampers. The obtained control law is then robustified using a disturbance observer. The properties are studied in the port Hamiltonian (pH) formalism which allows to show that the resulting closed-loop system is l_2 *weakly string stable* with respect to disturbances. The desired behavior of the closed-loop system is illustrated with some numerical simulation experiments.

Keywords: Formation Control, Bond Graph, Port Hamiltonian System, Interconnection and Damping assignment, Robust Control.

1. INTRODUCTION

The coordinated control of autonomous robots is an important area of research and its field of application is broad, encompassing problems such as Formation Control, sensor deployment (Tuna, Gungor and Gulez, 2014), map generation and capture (Tuna, Gungor and Gulez, 2014), (Tuna, Güngör and Potirakis, 2015), performing search and rescue tasks of people in hazard environments (Ollero et al., 2007), building monitoring and surveillance (Feddema, Lewis and Schoenwald, 2002), ground cleaning (Galceran and Carreras, 2013), lawn mowing (Yuming et al., 2011), crops harvesting (Ji et al., 2014), and ground mineral deposits detecting (Hameed, 2014), etc.

This paper tackles the problem of Formation Control (FC) (Soni and Hu, 2018) for a group of holonomic vehicles, which are represented as point masses in the plane. This group of vehicles, or platoon, moves at the same speed maintaining a desired geometry, which is specified by a desired inter-vehicle space.

A common and no desirable effect of these kind of systems is the *accordion* effect or *string instability* (Swaroop and Hedrick, 1996) (Swaroop and Hedrick, 1999). This effect takes place when the fluctuation of the speed of one vehicle, caused by a variable speed of the leader for example or by the action of external disturbances acting on the vehicles, propagates through the network increasing the distance among the vehicles especially downstream. These problems were well treated in the literature with multiple approaches, depending on the sensing capabilities of each agents and the desired topology, to mention: the *Leader-follower* approach (Gao et al., 2018), where each agent has the knowledge of the position and velocity of the leader, i.e. the leader must broadcast its position, velocity and, possibly, its acceleration in a speed tracking configuration, to all its followers. This methodology has two main drawbacks which are the lack of inter-vehicle information feedback throughout the group which can cause collisions among agents and the fact that the loss of leader information causes a fail on the entire group. Another methodology that requires less demand from the communication network is the *Predecessor-Following* approach (Knorn and Middleton, 2013) approach, where each agent has the knowledge of the relative position and velocity only of its predecessor agent. In (Seiler, Pant and Hedrick, 2004) the authors demonstrates that this configuration is always string unstable measuring only the relative position of the agents. Another approach that results as the combination of the previous two is the Leader-Predecessor-Following (Xiao, Gao and Wang, 2009) and guaranties string stability demanding more requirements to the communication network or other approaches that uses the information of the relative velocity and acceleration among the agents. The Predecessor-Successor approach or also known as *Bidirectional topology*, in which the control law of each agent is defined by the information of its Predecessors-Successors agents, i.e. the information propagates both upstream and downstream in the platoon. In (Barooah, Mehta and Hespanha, 2009) and (Seiler, Pant and Hedrick, 2004) it is shown that linear symmetric and bidirectional string measuring only the relative position of the agents is string unstable. The reader must refer to (Soni and Hu, 2018), (Zheng et al., 2016), (Middleton and Braslavsky, 2010), (Knorn and Middleton, 2013) for a sound review of these topologies and others.

The problem of FC can be attacked using a centralized or a decentralized approach. The first one demands the use of a global communication network that allows the exchange of information among vehicles and the computation of each control law. While, in the second approach, each agent computes its local control law using only local information, i.e. the i^{th} vehicle receives information only from its neighbor vehicles. The main goal in this work is to solve the FC problem using only local information, i.e. the i^{th} vehicle receives information only from its neighbor vehicles, rather than centralized controllers (Arcak, 2007), this reduces the requirements of the communication network to onboard sensors like radars, and the interconnection structure among the agents is closely related to the way that an agent acquires and process the information of its surrounding agents. To that end the desired behavior of the group pursued in this work, which is physically inspired, consists in a network of N masses coupled by virtual spring and dampers in parallel where the effect of these virtual elements is bidirectional. Different configurations between vehicle couplings are considered, from the weakly coupled configuration to the strongly or fully coupled. Due to direct connection with physics, the Bond Graph (BG) (Karnopp, Margolis and Rosenberg, 2006) formalism is used as a tool for network modeling and control law design.

The control law design follows an energy based approach, of the kind of interconnection and damping assignment (IDA-PBC) (Ortega and García-Canseco, 2004), completely designed in the BG domain (Junco, 2004) where: first, the closed-loop specifications are expressed by a so-called Target Bond Graph (TBG) representing the equivalent closed-loop behavior of the network. Then, in order to obtain the control law, the controlled sources –which provide the manipulated variables in the BG model of the plant– are prototyped (meaning that their behavior is expressed through BG components) in such a way that their power-interconnection with the rest of the plant BG –which is called a Virtual BG (VBG)– matches the TBG. Finally, the control law is obtained from the VBG by simply reading the outputs of the prototyped sources with the help of the causal assignment in the VBG.

To make the control law robust against external disturbances and the controlled system string stable with respect to input bounded disturbances, an extra control law based in the construction of a disturbance observer (DO) (Radke and Gao, 2006) is added.

The DBG was proposed by (Samantaray et al., 2006) for numerical evaluation of analytical redundant relationships. These are calculated to perform fault detection and isolation in an active fault tolerant control framework. Here the analytical redundant relationships or residues obtained from a closed-loop DBG are used to robustify the control law. The closed-loop DBG has been used to robustify control law against modelling error, parameter dispersion and external disturbances that acts in the same channel as the control input in (Nacusse and Junco, 2011) and (Nacusse and Junco, 2015). Recently, in (Nacusse, Donaire and Junco, 2018), this approach was formalized and extended, for disturbances with relative degree greater than one, in the pH framework with the form of DO.

The paper is organized as follows: Section 2 formulates the problem to be solved. Section 3 presents some tools and the metholodogy to be used. Section 4 presents the major result of the paper. Finally, in Section 5, some simulation results are provided to show the good performance of the control system.

2. PROBLEM FORMULATION

In this work a group of N of holonomic vehicles moving in a workspace $W \subset \mathbb{R}^2$ is considered. This group of vehicles, or platoon, moves at the same speed maintaining a desired geometry which is specified by a desired inter-vehicle space.

The equation of motion of each vehicle or agent is described by the double integrator, i.e. $\ddot{q}_i = u_i$ (with $i = 0, \dots, N$), being $q_i \in \mathbb{R}^2$the position of the i^{th} agent and $u_i = [u_{xi} \quad u_{yi}]^T$ the control input, and represented in the pH framework as in (1), being $q_i = [x_i \quad y_i]^T \in \mathbb{R}^2$ the Cartesian position of the i^{th} agent and $p_i = m_i[\dot{x}_i \quad \dot{y}_i]^T \in \mathbb{R}^2$ the linear momentum of the i^{th} point mass.

$$\begin{aligned} \begin{bmatrix} \dot{q}_i \\ \dot{p}_i \end{bmatrix} &= \begin{bmatrix} 0_2 & I_2 \\ -1_2 & 0_2 \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \end{bmatrix} + \begin{bmatrix} 0_2 \\ 1_2 \end{bmatrix} (u_i + d_i) \\ y &= \nabla_p H \end{aligned} \tag{1}$$

Where 0_2 and I_2 are the $2x2$ zero and identity matrices respectively. $H(q_i, p_i) = \frac{1}{2} p_i^T M_i^{-1} p_i$ is the storage function, $\nabla_q H = \partial H / \partial q$, $\nabla_p H = \partial H / \partial p$, $M_i = diag(m_i, m_i)$, $d_i = d_{ci} + d_{di}(t)$ is the perturbation input, where $d_i = [d_{xi} \quad d_{yi}]^T$, $d_{ci} = cte$ and $d_d(t)$ is bounded and variable with respect to time.

The FC Problem can be tackled using a centralized or a decentralized approach. The first one demands the use of a global communication network that allows the exchange of information among vehicles and the computation of each control law. In the second approach, each agent computes its local control law using only local information, i.e. the i^{th} vehicle receives information only from its neighbor vehicles. This work is framed in this last approach defining a physically inspired behavior of the group which consists in a network of N masses coupled by virtual springs and dampers in parallel. The connection among agents is bidirectional except for the leader which has its own control law independent of the other agents.

Two agents that are closer than a distance D_i are considered *neighbors* and have access to relative information. Being n_i the number of agents inside the *neighborhood*, each agent is connected to other and the number of coupling is indicated through a coupling index $k_i = \{1,2, \dots n_i\}$, which is defined as the number of bidirectional couplings. From the aforementioned, the following definitions are given.

Definition 1: the i^{th} agent is said to be a *Fully Coupled Agent* with distance D_i if its coupling index is $k_i = n_i$.

Definition 2: the i^{th} agent is said to be a *Partially Coupled Agent* with distance D_i and index k_i if its coupling index is equal to $k_i = n < n_i$.

Definition 3: a network with N agents is said to be a *Fully Coupled Network* with distance D_i if all its agents are fully coupled agents, otherwise is a *Partially Coupled Network.*

Figure 1 shows, without loss of generality, an array of $N = 25$ equally-spaced agents, where the i^{th} agent defines its *neighborhood* with a distance $D_i = 1.5$. The *neighborhood* it is composed by eight agents, i.e. $n_i = 8$, which are represented, in Figure 1, as the black dots inside the dashed circle. Notice that, if the i^{th} agent is a *partially coupled agent* then there are several coupling combinations among the i^{th} agent and its *neighbors* inside the dotted circle.

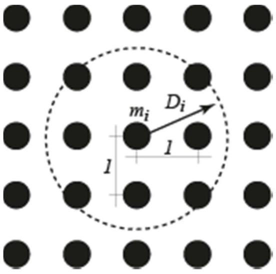


Figure 1: Definition of the *neighbourhood* of the i^{th} agent

The definition of string stability with respect to disturbances presented in (Knorn et al., 2014) will be used.

Definition 4: Consider a system described by $\dot{x} = f(x, d)$ with states $x \in R^{4N}$ and disturbances $d \in R^{2N}$, $f \in R^{4N}$ satisfying $f(x^*, 0) = 0$, where $m$is the number of springs. The equilibrium x^* is l_2 *weakly string stable with respect to disturbances* $d(t)$, if given any $\epsilon > 0$, there exists $\delta_1(\epsilon) > 0$ and $\delta_2(\epsilon) > 0$ (independent of N) such that:

$$|x(0) - x^*| < \delta_1(\epsilon) \; and \; \| d(.) \|_2 < \delta_2(\epsilon) \qquad (2)$$

implies

$$\|x(t) - x^*\|_\infty = sup_{t \geq 0}|x(t) - x^*| < \epsilon \; \forall N \geq 1 \quad (3)$$

Where $\| d(.) \|_2 = \sqrt{\int_0^\infty |d(t)|^2 dt}$

3. BACKGROUND AND METHODOLOGY

In this section the methodology used in the paper is detailed through a simple example consisting in two agents interconnected by means of physical components, namely a spring and a damper.

In the sequel it is assumed that the control signal has the form $u = u_{IDA} + v$, where u_{IDA} is an IDA-PBC law designed for the unperturbed system, i.e. (1) with $d = 0$, and v is an extra control input obtained from a DO.

The methodology employed can be summarized as follows: First and IDA-PBC strategy in the BG domain, using the virtual prototyping method (Junco, 2004), is employed to define the control law in absence of disturbance, i.e. $d = 0$. Then, the closed loop system equations in the pH framework are obtained from the BG domain using the methodology developed in (Donaire and Junco, 2009). Finally, the previous closed loop system is robustified using the output of a DO in the pH framework (Nacusse, Donaire and Junco, 2018).

3.1. IDA-PBC in the BG domain

The design of the control law u follows an energy based approach completely designed in the BG domain (Junco, 2004) where: first, the closed-loop specifications are expressed by a so-called Target Bond Graph (TBG), see Figure 2, representing the equivalent closed-loop behavior of the network. Then, in order to obtain the control law, the controlled sources –which provide the manipulated variables in the BG model of the plant– are prototyped (meaning that their behavior is expressed through BG components) in such a way that their power-interconnection with the rest of the plant BG –which is called a Virtual BG (VBG)– matches the TBG. Finally, the control law is obtained from the VBG by simply reading the outputs of the prototyped sources with the help of the causal assignment in the VBG is expressed in (4) for the i^{th} vehicle (an analogous law can be derived for the j^{th} vehicle).

In the vector BG of Figure 2 the corresponding *effort* and *flow* of each bond are vectors of two components each.

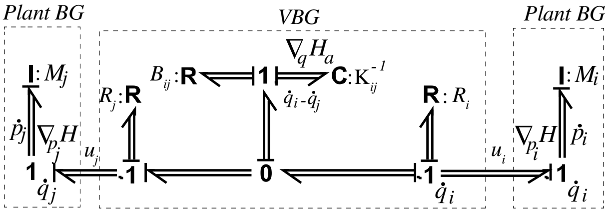


Figure 2: VBG of the interconnection between agents.

$$u_i = -\underbrace{K_{\mathrm{ij}}\left(q_i - q_{\mathrm{j}} - L_{ij}\right)}_{\frac{\partial H_a}{\partial \tilde{q}_{ij}},} - B_{\mathrm{ij}} \underbrace{\left(\dot{q}_i - \dot{q}_{\mathrm{j}}\right)}_{\frac{\partial H}{\partial p_i} - \frac{\partial H}{\partial p_j}} - R_i \dot{q}_i \qquad (4)$$

where, $K_{ij} = diag\left(k_{ijx,}k_{ijy,}\right)$, $B_{ij} = diag\left(b_{ijx,}b_{ijy}\right)$ and $L_{ij} = [L_{xij} \quad L_{yij}]^T$ is the natural length of the spring and represents the desired distance to be kept between the two vehicles, B_{ij}, R_i and R_j are design parameters to be chosen. Note that, besides the virtual spring-damper interconnection between the two vehicles, dissipation has been assigned to each of them through the elements with coefficients $R_{i,j}$.

Remark 1: notice that, the first term of (4) is associated with the gradient of the added potential energy due to the action of the spring, i.e. $\frac{\partial H_a}{\partial \tilde{q}_{ij}} = K_{ij}\tilde{q}_{ij}$, where $\tilde{q}_{ij} = \left(q_i - q_j - L_{ij}\right)$ and $H_a\left(q_i, q_j\right) = 1/2\tilde{q}_{ij}{}^T K_{ij}\tilde{q}_{ij}$. Without loss of generality a linear constitutive relationship for the spring and the damper has been chosen. A nonlinear constitutive relationship, particularly in the spring, could provide some advantages in the performance of the closed loop. For example, a nonlinear relation may augment the force exponentially when two vehicles are too close.

Remark 2: notice that, if the vehicles move along a straight line, i.e. the workspace $W \subset \mathbb{R}$, then the vector TBG of Figure 2 is reduced to a single bond TBG.

3.1.1. Obtaining the pH system from the BG

The related pH system can be obtained directly from the BG of Figure 2 via following the procedure detailed in (Donaire and Junco, 2009). In particular, in a BG model with all the storage elements in integral causality, as the one shown in Figure 2, the procedure can be summarized as follows:

1. Compute the total energy of the system using the constitutive relationships of the storage elements.

$$H_{dij}\left(\tilde{q}_{ij}, p_i, p_j\right) = \frac{1}{2} p_i^T M_i^{-1} p_i + \frac{1}{2} p_j^T M_j^{-1} p_j + \frac{1}{2}\tilde{q}_{ij}^T K_{ij}\tilde{q}_{ij} \tag{5}$$

The flow or effort variables entering to the storage elements are the time derivatives $\dot{x}$ of the states which in this example are $\dot{x} = [\dot{q} \quad \dot{p}]^T$, while the outputs of the storage elements are the gradient components of the storage elements $\nabla H_{dij} = \left[\left(K_{ij}\left(q_i - q_j - L_{ij}\right)\right)^T \quad \left(M_i^{-1} p_i\right)^T \quad \left(M_j^{-1} p_j\right)^T\right]^T$.

2. Compute the structure and dissipation matrixes J and R using the gains of the causal paths between the storage elements, and between the storage and the dissipation elements, respectively.

$$\begin{bmatrix} \dot{\tilde{q}}_{ij} \\ \dot{p}_i \\ \dot{p}_j \end{bmatrix} = \left[\underbrace{\begin{bmatrix} 0_2 & I_2 & -I_2 \\ -I_2 & 0_2 & 0_2 \\ I_2 & 0_2 & 0_2 \end{bmatrix}}_{J} - \underbrace{\begin{bmatrix} 0_2 & 0_2 & 0_2 \\ 0_2 & \multicolumn{2}{c}{\multirow{2}{*}{R_d}} \\ 0_2 & & \end{bmatrix}}_{R}\right] \nabla H_{dij} \tag{6}$$

Where R_d is the $4x4$ matrix

$$R_d = \begin{bmatrix} -\left(B_{ij} + R_i\right) & B_{ij} \\ B_{ij} & -\left(B_{ij} + R_j\right) \end{bmatrix}$$

Remark 3: The stability properties of the equilibrium point, $\left(\bar{p}_i, \bar{p}_j, \tilde{q}_{ij}\right) = \left([0,0]^T, [0,0]^T, [0,0]^T\right)$, of the closed loop system, defined in the TBG of Figure 2, can be analyzed using the energy function (5) as a Lyapunov function candidate and the *LaSalle* invariance principle.

3.2. DBG and DO in the pH framework

This section defines a closed-loop DBG from a behavioral BG model of the desired closed-loop. The output of the closed-loop DBG is a residual signal that indicates the discrepancy between the desired closed-loop dynamics and real one. Then a DO is defined in the pH framework and its output, i.e. the control input v, is used next to design the outer control loop in order to compensate or attenuate the effect of the perturbation.

Thus the perturbed closed-loop systems results from replacing $u = u_{IDA} + v$ on the plant (1), i.e. with $d \neq 0$, and replacing (4) into (1) results in:

$$\begin{bmatrix} \dot{\tilde{q}}_{ij} \\ \dot{p}_i \\ \dot{p}_j \end{bmatrix} = \begin{bmatrix} 0_2 & I_2 & -I_2 \\ -I_2 & \multicolumn{2}{c}{\multirow{2}{*}{$-R_d$}} \\ I_2 & & \end{bmatrix} \nabla H_{dij} + \begin{bmatrix} 0_2 \\ v_i \\ v_j \end{bmatrix} + \begin{bmatrix} 0_2 \\ d_i \\ d_j \end{bmatrix} \tag{7}$$

3.2.1. Closed-Loop DBG

The closed-loop DBG is constructed from a behavioral BG model of the desired closed-loop model injecting the plant measurements through modulated sources. The residual signal is then obtained by measuring the power co-variables of the modulated sources, and is an indication of the discrepancy between the desired and real perturbed closed-loop dynamics.

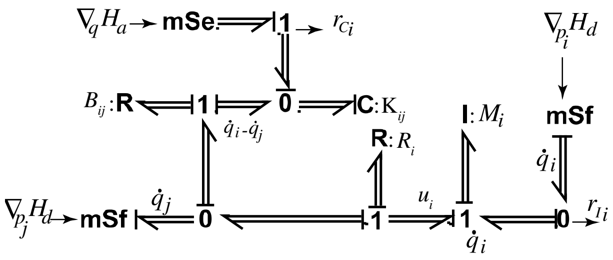


Figure 3: Closed-loop DBG of the interconnection between agents for the i^{th} point mass.

In Figure 3 closed-loop DBG of the interconnection between agents for the i^{th} point mass is shown where the residual signals r_{I_i} and r_{C_i} can be obtained. Notice that, if the masses of the agents are known, then $r_{C_i} = 0$, since the injected effort on the 0-junction is calculated by the control input, i.e. the first term of (4), and $\tilde{q}_{ij}$ is a state of the controller. In addition, the residual signal r_{I_i} can be computed reading the effort on the associated 0-junction as in (9).

Thus replacing the $\dot{p}_i$ of (7) into (9) yields (10), where the residual signal is driven by the perturbations.

$$r_{C_i}(x) = \nabla_{p_i} H_{dij} - \nabla_{p_j} H_{dij} - \dot{\tilde{q}}_{ij} \tag{8}$$

$$r_{I_i}(x) = \dot{p}_i + K_{ij}\tilde{q}_{ij} + B_{ij}\left(\dot{q}_i - \dot{q}_j\right) + R_i\dot{q}_i \qquad (9)$$

$$r_{I_i}(x) = v_i + d_i \qquad (10)$$

The dynamics of the DO for the disturbance is defined using the residual signal as follows.

$$\dot{z}_i = G_i^{-1}\, r_{I_i}(x) \qquad (11)$$

Where $G_i > 0, G_i = G_i^T \in R^2$ is a diagonal matrix. Or, expressed in term of the desired closed-loop pH system replacing (9) into (11), yields (12).

$$\dot{z}_i = G_i^{-1}\left[\dot{p}_i + K_{ij}\tilde{q}_{ij} + B_{ij}\left(\dot{q}_i - \dot{q}_j\right) + R_i\dot{q}_i\right] \qquad (12)$$

To show that z_i is the disturbance estimations replace $\dot{p}_i$ from (7) in (12), with $v_i = -z_i$, obtaining.

$$\dot{z}_i = -G_i^{-1}z_i + G_i^{-1}d_i \qquad (13)$$

Then, defining the perturbation error as $e_{d_i} = z_i - d_i$, for $i = 1,2$. and replacing e_{d_i} in (13), the dynamics of e_{d_i} are.

$$\dot{e}_{d_i} = -G_i^{-1}e_{d_i} - \dot{d}_i \qquad (14)$$

The perturbation-error dynamics are driven by $\dot{d}_i$, the time derivative of the perturbations. It is straightforward to prove that this error tends to zero exponentially for constant perturbations, i.e. $\dot{d}_i = 0$, and remains bounded if $\|\dot{d}_i\| < \alpha_i$. Notice that the choice of the constant matrix G_i fixes the rate of convergence of the DO.

Remark: the DO defined in (12) depends on the time derivatives of the states, i.e. $\dot{p}_i$. In real applications these variables cannot be always measured via sensors, thus it is needed to compute them with the consequent error due to noise in the measurements. To solve this problem, an internal extra variable of the DO can be defined, see (Mohammadi, Marquez and Tavakoli, 2017) for further details about this procedure. In this example, i.e. two masses connected through the VBG of Figure 2 integrating (12) allows to express the control input $v_i = -z_i$ in terms of the closed loop variable as in (15).

$$\begin{aligned} v_i = &-G_i^{-1}M_i\dot{q}_i - G_i^{-1}B_{ij}\left(q_i - q_j\right) - G_i^{-1}R_iq_i \\ &- G_i^{-1}\int_0^t K_{ij}\tilde{q}_{ij}\, d\tau \end{aligned} \qquad (15)$$

3.2.2. DO in the pH framework

The previous ideas, elaborated above on the Bond Graph domain for the control of just one vehicle of the platoon, is extended to the whole system and theoretically developed in the pHs set-up, for further details on this approach refer to (Nacusse, Donaire and Junco, 2018) . Figure 4 depicts the block diagram representation of the connection between the plant and the so-called Diagnostic pH system (D-pH), where the measurements injected into the D-pH block are identified as the gradient of the Hamiltonian or stored energy.

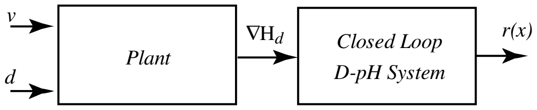


Figure 4: Interconnection between plant and Closed-loop D-pH System.

Figure 5 shows an internal representation of the D-pH system, where it is assumed that ∇H_d is bijective, i.e. exists $h_d\left(\nabla H_d(x)\right) = x$. Notice that, x is the state variable driven by the dynamics of the perturbed system. Where $J_d(x)$ and $R_d(x)$ are the desired interconnection and dissipation matrices.

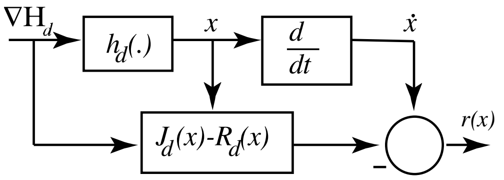


Figure 5: Internal block diagram of the closed-loop D-pH.

The system (16) represents the closed-loop system with the control input $u = u_{IDA} + v$, being u_{IDA} the collection of all interconnection control laws of the form (4).

$$\begin{aligned} \begin{bmatrix} \dot{\tilde{q}} \\ \dot{p} \end{bmatrix} &= \begin{bmatrix} 0_m & S \\ -S^T & -R_d \end{bmatrix} \begin{bmatrix} \nabla_{\tilde{q}} H_d \\ \nabla_p H_d \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} (v + d) \\ y &= \nabla_p H \end{aligned} \qquad (16)$$

With $H_d(\tilde{q}, p) = \frac{1}{2}p^T\mathcal{M}^{-1}p + \frac{1}{2}\tilde{q}^TK\tilde{q}$

Where $p \in R^{2N}$ and $\tilde{q} \in R^m$ are column vectors that collect all the generalized momenta of the vehicles masses and the states of the springs, respectively. S is a $mx2N$ matrix with most of its elements equal to zero and that contains only a 1 and a -1 on each row representing the interconnection structure between agents; R_d is the dissipation structure matrix, which is assumed positive definite by design; $\mathcal{M} = diag(M_1, \ldots, M_N)$ and $K = diag(K_1, \ldots, K_m)$.

Thus, the outputs of the D-pH are $r(x) = [r_q(x), r_p(x)]^T$:

$$r_q(x) = \dot{\tilde{q}} - S\nabla_p H_d \qquad (17)$$

$$r_p(x) = \dot{p} + S^T \nabla_{\tilde{q}} H_d + R_d \nabla_p H_d \qquad (18)$$

The same procedure described above in the BG domain can be applied to (18) to obtain the output of the DO.

$$\dot{z} = G^{-1}\, r_p(x) \qquad (19)$$

$$\dot{z} = -G^{-1}\, z + G^{-1} d \qquad (20)$$

where $G = diag(G_1, \dots, G_N)$ is the gain of the DO.

4. MAIN RESULT

In this section the properties of a network of N interconnected agents, with an extra control law v which depends on the output of the DO, are studied. The network of N interconnected agents is represented by the system (16) where each agent can be coupled to more than one neighbor depending on the network configuration.

Proposition:

System (16) with the disturbance estimation z (19) and control input (20) has the following properties:

$$v = -z - \left(\frac{1}{4}\right) G G^T \nabla_p H_d \qquad (21)$$

1- It can be expressed as a pH system as:

$$\begin{bmatrix} \dot{\tilde{q}} \\ \dot{p} \\ \dot{z} \end{bmatrix} = [J_c - R_c] \nabla Q + \beta\, d \qquad (22)$$

$$With \quad J_c = \begin{bmatrix} 0 & S & 0 \\ -S^T & 0 & -\frac{1}{2}G \\ 0 & \frac{1}{2}G^T & 0 \end{bmatrix}, R_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & R_d^* & \frac{1}{2}G \\ 0 & \frac{1}{2}G^T & I \end{bmatrix},$$

$R_d^* = R_d + \left(\frac{1}{4}\right) G G^T$, $\beta = [0 \quad I \quad G^{-1}]^T and$ $Q(p, \tilde{q}, z) = H_d(p, \tilde{q}) + \frac{1}{2} z^T G^{-1} z$: where the 0 and I are the zero and identity matrices with appropriate dimensions.

2- If the disturbance d is constant, that is $d(t) = d_c$ and $d_d(t) = 0$, then the equilibrium $(p^*, \tilde{q}^*, z^*) = (0, 0, d_c)$ of the closed loop is asymptotically stable with Lyapunov function (23):

$$Q_2 = H_d(p, \tilde{q}) + \frac{1}{2}(z - d_c)^T G^{-1} (z - d_c) \qquad (23)$$

3- The closed loop system (22) is l_2 weakly string stable with respect to the dynamic disturbances $d(t)$.

Notice that the term $(1/4) G G^T \nabla_p H_d$ in (21) is a damping that always can be injected.

Proof:

1- To prove the first claim consider $Q(p, \tilde{q}, z) = H_d(p, \tilde{q}) + \frac{1}{2} z^T G^{-1} z$, then writing the dynamics of the states $[p, \tilde{q}]$ and substituting the input by the control law (21), yields the closed loop dynamics (24):

$$\begin{bmatrix} \dot{\tilde{q}} \\ \dot{p} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & S & 0 \\ -S^T & -R_d^* & -G \\ 0 & 0 & -I \end{bmatrix} \begin{bmatrix} \nabla_{\tilde{q}} H_d \\ \nabla_p H_d \\ G^{-1} z \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ -G^{-1} \end{bmatrix} d \qquad (24)$$

Finally decompose the matrix that multiplies ∇Q in (24) into its symmetric and skew-symmetric component to obtain the dynamics (22).

2- To prove that $(p^*, \tilde{q}^*, z^*) = (0, 0, d_c)$ is an asymptotically stable equilibrium point of the system define Q_2 as in (23), then the closed loop system can be expressed as:

$$\begin{bmatrix} \dot{\tilde{q}} \\ \dot{p} \\ \dot{z} \end{bmatrix} = [J_c - R_c] \nabla Q_2 \qquad (25)$$

Use Q_2 as a candidate Lyapunov function, and compute its time derivative, which result as follows

$$\dot{Q}_2 = \nabla {Q_2}^T \begin{bmatrix} \dot{\tilde{q}} \\ \dot{p} \\ \dot{z} \end{bmatrix} \qquad (26)$$

$$\dot{Q}_2 = \nabla {Q_2}^T \begin{bmatrix} 0 & S & 0 \\ -S^T & -R_d^* & -G \\ 0 & 0 & -I \end{bmatrix} \nabla Q_2 \qquad (27)$$

$$\dot{Q}_2 = -\nabla {Q_2}^T \begin{bmatrix} 0 & 0 & 0 \\ 0 & R_d^* & \frac{1}{2}G \\ 0 & \frac{1}{2}G^T & I \end{bmatrix} \nabla Q_2 \qquad (28)$$

$$\dot{Q}_2 = \begin{bmatrix} \nabla_p H_d \\ G^{-1}(z - d_c) \end{bmatrix}^T \underbrace{\begin{bmatrix} R_d^* & \frac{1}{2}G \\ \frac{1}{2}G^T & I \end{bmatrix}}_{R_c^*} \begin{bmatrix} \nabla_p H_d \\ G^{-1}(z - d_c) \end{bmatrix} \qquad (29)$$

Applying Schur's complements in $R_c^* > 0 \Leftrightarrow R_d^* > 0$ which implies that $\dot{Q}_2 \leq 0$. Thus, the equilibrium point is asymptotically stable via the application of the *La Salle* Invariance Principle, which ensures that the trajectories of the state converge to the largest invariant set (Khalil, 2002).

3- The procedure used to prove *Theorem 4* in (Knorn et al., 2014) it is used here to prove this claim. Using $H_d(p,\tilde{q})$ as candidate of Lyapunov function and following the procedure of the proof of the claim 2, then the derivative of $H_d(p,\tilde{q})$ along the trajectories can be written as:

$$\dot{\mathrm{H}}_\mathrm{d} \leq -\nabla \mathrm{H_d}^\mathrm{T} \begin{bmatrix} 0 & 0 & 0 \\ 0 & R_d^* & \frac{1}{2}\mathrm{G} \\ 0 & \frac{1}{2}\mathrm{G}^\mathrm{T} & I \end{bmatrix} \nabla \mathrm{H_d} + \nabla \mathrm{H_d}^\mathrm{T} d_d \tag{30}$$

$$\dot{\mathrm{H}}_\mathrm{d} \leq -\chi^T \begin{bmatrix} R_d^* & \frac{1}{2}\mathrm{G}^\mathrm{T} \\ \frac{1}{2}G^T & I \end{bmatrix} \chi + \chi^T \delta_d \tag{31}$$

where $\chi = \begin{bmatrix} \nabla_p H_d & G^{-1} z \end{bmatrix}$ and $\delta_d = \begin{bmatrix} I \\ G^{-1} \end{bmatrix} d_d$ then:

$$\dot{\mathrm{H}}_\mathrm{d} \leq -\lambda_{min}(R_d^*)|\chi|^2 + \chi^T \delta_d \tag{32}$$

$$\dot{\mathrm{H}}_\mathrm{d} \leq -\frac{1}{2}\lambda_{min}(R_d^*)|\chi|^2 + \frac{1}{2\lambda_{min}(R_d^*)}|\delta_d|^2 \tag{33}$$

$$\dot{\mathrm{H}}_\mathrm{d} \leq \frac{1}{2\lambda_{min}(R_d^*)}|\delta_d|^2 \tag{34}$$

Then, integrating both terms of (34) along time:

$$\mathrm{H_d}(t) \leq \mathrm{H_d}(0) + \frac{1}{2\lambda_{min}(R_d^*)}\|\delta_\mathrm{d}\|_2{}^2 \tag{35}$$

Replacing $H_d(0)$ and operating yields

$$\begin{aligned} H_d(t) \leq & \frac{1}{2}\lambda_{min}(M^{-1})|p(0)| + \frac{1}{2}\lambda_{min}(K)|\tilde{q}(0)| \\ & + \frac{1}{2}\lambda_{min}(G^{-1})|z(0)| + \frac{1}{2\,\lambda_{min}(R_d^*)}\|\delta_d\|_2{}^2 \end{aligned} \tag{36}$$

Which means that $\mathrm{H_d}(p,\tilde{\mathrm{q}},z,t)$ is bounded for all agents if $|p(0)|$, $|\tilde{q}(0)|$, $|z(0)|$, and $\|\delta_d\|^2$ do not increase with number of agents N. As $H_d(p,q,z,t)$ is monotically increasing, then an upper bound of $\mathrm{H_d}(p,\tilde{\mathrm{q}},z,t)$ implies that the states $(p,\tilde{q},z)$ are also bounded. Therefore, the system is l_2 weakly string stable with respect to the dynamic disturbances $d_d(t)$.

5. APPLICATION EXAMPLES

This section presents some simulations results to show the performance of the control laws obtained above in two different configurations among agents. First a *partially coupled network* is studied, see Figure 6a, in which each agent has coupling index $k = 2$, i.e. each agent is connected to only two other agents, and then a *fully coupled network*, see Figure 6b, configuration in which the agents are connected to all the surrounding agents with distance $D_i \leq 2$.

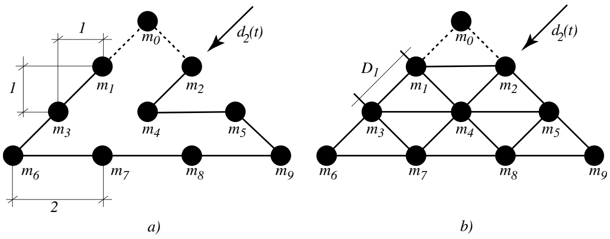


Figure 6: Interconnection and distance, in meters, among agents. a) *partially coupled network* with coupling index $k = 2$. b) *fully coupled network.*

Figure 6 shows the desired triangle formation where the black dots represent the agents and the connections among them are represented by lines. The dashed lines represent the unidirectional coupling of *agent 1* and *agent 2* with the leader, while the solid lines represent the bi-directional coupling among agents, i.e. in the BG domain these lines are represented with the VBG shown in Figure 2.

The simulations were performed using 20sim environment (20Sim, 2013) and the scenario is as follows: at time $t = 0sec$ the agents are gathered at the origin and then they move to the desired triangle formation. At time $t = 8sec$ the leader moves 1 meter in the Y direction. Finally, at time $t = 20sec$ a disturbance, which is a logarithmic sine sweep of the form, $d(t) = 50\sin(\omega(t)t)$ (see 20Sim reference manual for further details), affects the agent 2 as indicated in Figure 6.

The parameters used in the simulations are: $m_i = 1Kg$, for $i = 0\ to\ 9$, $B_i = 0\,I_2 Kg/sec$, $B_{ij} = 10I_2 Kg/sec$, $K_{ij} = 10I_2 Kg/sec^2$, $L_{ij} = I_2 m$ and $G_i = 100I_2$, where I_2 is the $2x2$ identity matrix.

In Figure 7 the position of the leader, the disturbance affecting the mass 2 and the disturbance error are depicted for both configurations. Notice that the disturbance error can be reduced even more by increasing the value of G_i.

Figure 8 and Figure 9 show the distance between the leader and each agent for the *partially coupled network* and the *fully coupled network* configuration respectively, with and without the action of the DO. The distance between the leader and each agent, defined in the desired formation configuration of Figure 6, is $Dist\_m_i = |q_0 - q_i|$ for $i = 0\ to\ 9$. As can be seen in Figure 8a and Figure 9a for time $t < 20\,sec$ and in Figure 8b and Figure 9b the $Dist\_m_i$ of each agent reaches the desired distance. Notice, the improvement due to the application of the control input v in Figure 8b and Figure 9b for time $t > 20\ sec$.

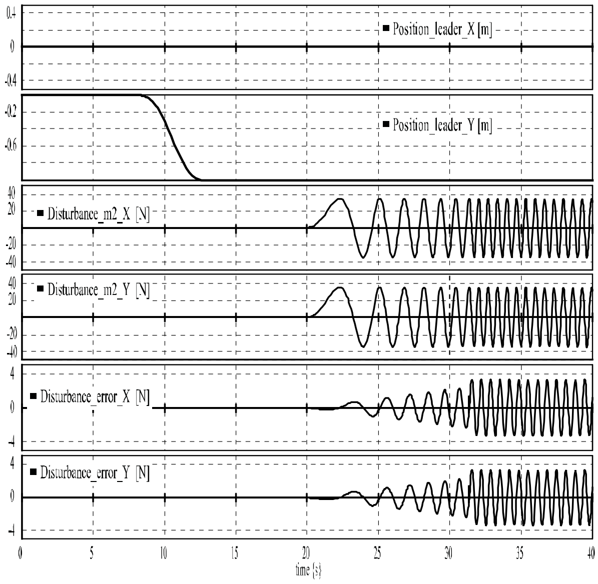


Figure 7: From top to bottom: Cartesian position XY of the leader; Cartesian disturbance XY acting on m_2; Cartesian disturbance errror XY obtained from the DO.

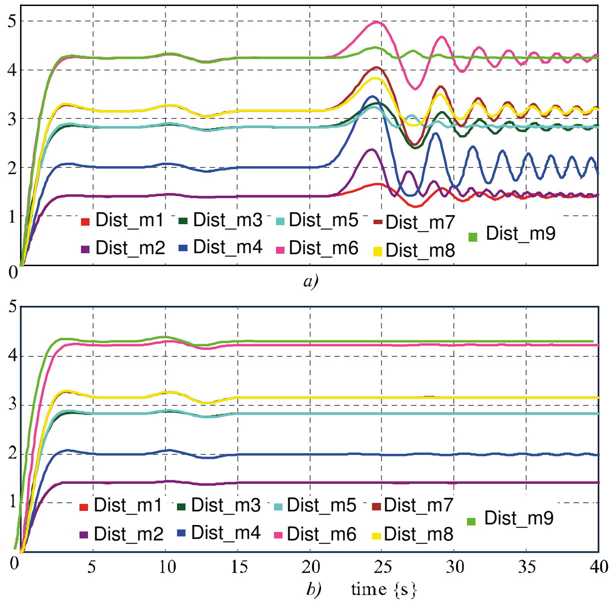


Figure 8: Distance, in meters, between the leader and each agent. a) without DO compensator, b) with DO compensator.

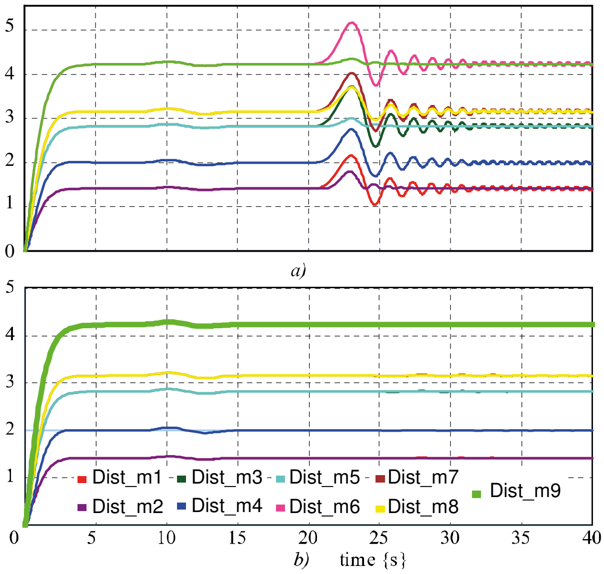


Figure 9: Distance, in meters, between the leader and each agent. a) without DO compensator, b) with DO compensator.

6. CONCLUSIONS

This work tackles the problem of formation control for a group of holonomic vehicles using the Bond Graphs formalism. The control laws for the agents are physically inspired and designed in the BG domain. Later these control laws are robustified by adding an extra control action based in a DO definition. The main properties of the resulting closed-loop are: constant disturbance rejection and l_2 weakly string stable with respect to disturbances.

ACKNOWLEDGMENTS

The authors wish to thanks SeCyT-UNR (the Secretary for Science and Technology of the National University of Rosario) for their financial support through project PID-UNR 1ING502 and the ANPCyT (Argentine National Agency for Scientific and Technological Promotion), under project PICT 2017-3644.

APPENDIX

The perturbed TBG for *partially coupled network* and the *fully coupled network* interconnection are shown in Figure 10 and Figure 11, where the interconnection between the 1-junctions in done through the VBG of Figure 2. The matrixes S and R of the system (16), are not deduced here due to space constraint, but these can be computed following the procedure detailed in Section 3.1.1.

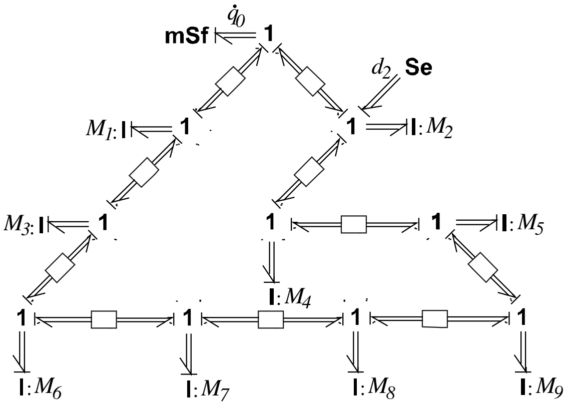


Figure 10: Perturbed TBG of 10 agents in triangle formation for *partially coupled network* of Figure 6a.

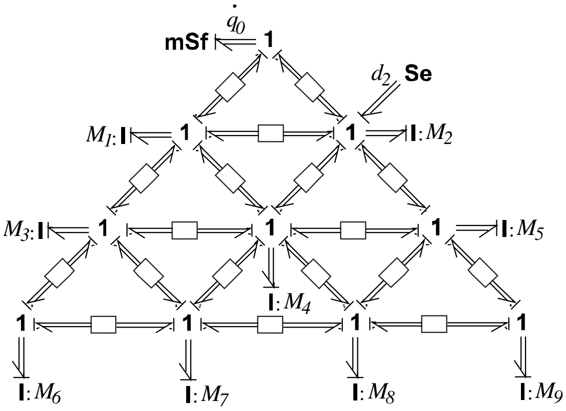


Figure 11: Perturbed TBG of 10 agents in triangle formation for *partially coupled network* of Figure 6b.

REFERENCES

20Sim (2013) *Controllab Products, B. V.*, Available: http://www.20sim.com.

Arcak, M. (2007) 'Passivity as a Design Tool for Group Coordination', *IEEE Transactions on Automatic Control*, vol. 52, no. 8, Aug, pp. 1380-1390.

Barooah, P., Mehta, P.G. and Hespanha, J.P. (2009) 'Mistuning-Based Control Design to Improve Closed-Loop Stability Margin of Vehicular Platoons', *IEEE Transactions on Automatic Control*, vol. 54, no. 9, Sep., pp. 2100-2113.

Donaire, A. and Junco, S. (2009) 'Derivation of input-state-output port-hamiltonian systems from bond graphs', *Simulation Modelling Practice and Theory*, vol. 17, no. 1, pp. 137-151.

Donaire, A. and Junco, S. (2009) 'Energy shaping, interconnection and damping assignment, and integral control in the bond graph domain', *Simulation Modelling Practice and Theory*, vol. 17, no. 1, pp. 152-174.

Feddema, J.T., Lewis, C. and Schoenwald, D.A. (2002) 'Decentralized control of cooperative robotic vehicles: theory and application', *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, Oct, pp. 852-864.

Galceran, E. and Carreras, M. (2013) 'Planning coverage paths on bathymetric maps for in-detail inspection of the ocean floor', 2013 IEEE International Conference on Robotics and Automation, 4159-4164.

Gao, W., Jiang, Z., Lewis, F.L. and Wang, Y. (2018) 'Leader-to-Formation Stability of Multiagent Systems: An Adaptive Optimal Control Approach', *IEEE Transactions on Automatic Control*, vol. 63, no. 10, Oct, pp. 3581-3587.

Hameed, I.A. (2014) 'Intelligent Coverage Path Planning for Agricultural Robots and Autonomous Machines on Three-Dimensional Terrain', *Journal of Intelligent Robotic Systems*, vol. 74, no. 3, Jun, pp. 965-983.

Ji, W., Li, J.L., Zhao, D.A. and Jun, Y. (2014) 'Obstacle Avoidance Path Planning for Harvesting Robot Manipulator Based on MAKLINK Graph and Improved Ant Colony Algorithm', Advances in Measurements and Information Technologies, 1063-1067.

Junco, S. (2004) 'Virtual Prototyping of Bond Graphs Models for Controller Synthesis through Energy and Power Shaping', International Conference on Integrated Modeling and Analysis in Applied Control and Automation (IMAACA 2004).

Karnopp, D.C., Margolis, D.L. and Rosenberg, R.C. (2006) *System Dynamics: Modeling and Simulation of Mechatronic Systems*, New York, NY, USA: John Wiley \& Sons, Inc.

Khalil, H.K. (2002) *Nonlinear Systems*, Prentice Hall PTR.

Knorn, S., Donaire, A., Agüero, J.C. and Middleton, R.H. (2014) 'Passivity-based control for multi-vehicle systems subject to string constraints', *Automatica*, vol. 50, no. 12, pp. 3224-3230.

Knorn, S. and Middleton, R.H. (2013) 'Two-dimensional analysis of string stability of nonlinear vehicle strings', 52nd IEEE Conference on Decision and Control, 5864-5869.

Middleton, R.H. and Braslavsky, J.H. (2010) 'String Instability in Classes of Linear Time Invariant Formation Control With Limited Communication Range', *IEEE Transactions on Automatic Control*, vol. 55, no. 7, July, pp. 1519-1530.

Mohammadi, A., Marquez, H.J. and Tavakoli, M. (2017) 'Nonlinear Disturbance Observers: Design and Applications to Euler-Lagrange Systems', *IEEE Control Systems*, vol. 37, Aug, pp. 50-72, Available: ISSN: 1066-033X.

Nacusse, M., Donaire, A. and Junco, S. (2018) 'Robustifying Passive Closed-Lopp Port-Hamiltonian Systems Using Observer Based Control', International Conference on Integrated Modeling and Analysis in Applied Control and Automation (IMAACA 2018).

Nacusse, M. and Junco, S. (2011) 'Passive Fault Tolerant Ccontrol: a Bond Graph Approach', International Conference on Integrated Modeling and Analysis in Applied Control and Automation (IMAACA 2011).

Nacusse, M. and Junco, S. (2015) 'Bond-graph-based controller design for the quadruple-tank process', *International Journal of Simulation and Process Modelling.*, vol. 10, pp. 179-191.

Ollero, A., Marron, P.J., Bernard, M., Lepley, J., la Civita, M., de Andres, E. and van Hoesel, L. (2007) 'AWARE: Platform for Autonomous self-deploying and operation of Wireless sensor-actuator networks cooperating with unmanned AeRial vehiclEs', 2007 IEEE International Workshop on Safety, Security and Rescue Robotics, 1-6.

Ortega, R. and García-Canseco, E. (2004) 'Interconnection and Damping Assignment Passivity-Based Control: A Survey', *European Journal of Control*, vol. 10, no. 5, pp. 432-450.

Radke, A. and Gao, Z. (2006) 'A survey of state and disturbance observers for practitioners', 2006 American Control Conference, 6 pp.-.

Samantaray, A.K., Medjaher, K., Bouamama, B.O., Staroswiecki, M. and Dauphin-Tanguy, G. (2006) 'Diagnostic bond graphs for online fault detection and isolation', *Simulation Modelling Practice and Theory*, vol. 14, no. 3, pp. 237-262.

Seiler, P., Pant, A. and Hedrick, K. (2004) 'Disturbance propagation in vehicle strings', *IEEE Transactions on Automatic Control*, vol. 49, no. 10, Oct, pp. 1835-1842.

Soni, A. and Hu, H. (2018) 'Formation Control for a Fleet of Autonomous Ground Vehicles: A Survey', *Robotics*, vol. 7, no. 4.

Swaroop, D. and Hedrick, J.K. (1996) 'String stability of interconnected systems', *IEEE Transactions on Automatic Control*, vol. 41, no. 3, March, pp. 349-357.

Swaroop, D. and Hedrick, J. (1999) 'Constant Spacing Strategies for Platooning in Automated Highway Systems.', *ASME. J. Dyn. Sys., Meas., Control.*, vol. 121, no. 3, pp. :462-470.

Tuna, G., Gungor, V.C. and Gulez, K. (2014) 'An autonomous wireless sensor network deployment system using mobile robots for human existence detection in case of disasters', *Ad Hoc Networks*, vol. 13, pp. 54-68.

Tuna, G., Güngör, V.Ç. and Potirakis, S.M. (2015) 'Wireless sensor network-based communication for cooperative simultaneous localization and mapping', *Computers \& Electrical Engineering*, vol. 41, pp. 407-425.

Xiao, L., Gao, F. and Wang, J. (2009) 'On scalability of platoon of automated vehicles for leader-predecessor information framework', 2009 IEEE Intelligent Vehicles Symposium, 1103-1108.

Yuming, L., Ruchun, W., Zhenli, Z. and Junlin, Z. (2011) 'Dynamic coverage path planning and obstacle avoidance for cleaning robot based on behavior', 2011 International Conference on Electric Information and Control Engineering, 4952-4956.

Zheng, Y., Eben Li, S., Wang, J., Cao, D. and Li, K. (2016) 'Stability and Scalability of Homogeneous Vehicular Platoon: Study on the Influence of Information Flow Topologies', *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 1, Jan, pp. 14-26.

FULL MODELLING AND SLIDING MODE CONTROL FOR A QUADROTOR UAV IN VISUAL SERVOING TASK

Choukri Bensalah[a], **Nacer K. M'Sirdi**[b], **Aziz Naamane**[c]

[b], [c]Aix Marseille University, Université de Toulon, CNRS, LIS UMR 7020, SASV, Marseille, France
[a]Laboratoire d'Automatique de Tlemcen (LAT), University of Tlemcen, Algeria

[b]nacer.msirdi@lis-lab.fr, [c]aziz.naamane@lis-lab.fr , [a]choukri.bensalah@lis-lab.fr

ABSTRACT

In this paper the Control of an UAV is designed by an Image Based Visual Servoing (IBVS). The trajectories are generated by the IBVS to track a target. The dynamic model describing the Quadrotor behaviour is established taking into account all internal and external aerodynamic forces and moments, for simulation. Furthermore, actuators and sensors dynamics are also considered. To overcome all the nonlinearities, as well as the strong coupling in 3D position and Euler angles of Quadrotor system, a robust Sliding Mode Control (SMC) is designed. Unlike several literature works in this topic, the SMC control uses only an approximated model of the Quadrotor. In other words, SMC does not use the dynamic inversion of Quadrotor model. The Quadrotor is endowed a virtual camera to perform visual tracking, in order to evaluate the robustness of our controller.

Keywords : Full Quadrotor modelling, SMC Control, nonlinear control and compensation, IBVS, Virtual Camera.

1. INTRODUCTION

Unmanned Aerial Vehicles (UAV) are popular due to the multiple application possibilities (rescue, surveillance, inspection, mapping, bridges, buildings supervision, and cinema). Their ability to take off and land vertically, to perform stationary flight as well as their manoeuvrability and controllability give them a key position as mobile robots Guerrero and Lozano [2012]Rabhi et al. [2011]Lozano [2013]. The four rotors helicopter (see figure 1) exhibits a nonlinear behaviour which is subject to aerodynamic forces, moments and unknown disturbances Mistler et al. [2001]Mistler et al. [2002]. It is multi-variable, non linearly coupled and has inherent uncertainties in both high and low frequencies Mederreg et al. [2004]. This work was supported by SASV of LiS from AMU

The required application objective is its capability to follow desired trajectories and autonomous motions. The system must have some desired features like robustness to uncertainties, perturbations and parameters variations Mokhtari et al. [2006].

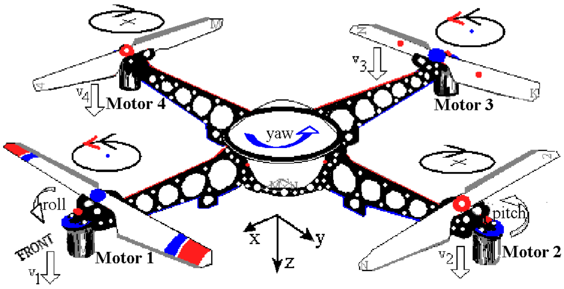


Figure 1: Mechanical structure of the Quadrotor.

During the past two decades, the Quadrotor is considered as suitable platform to evaluate and compare the performances of designed controllers. In his thesis Bouabdallah [2007], the author has developed several existing classical controllers; Lyapunov based control, PID, LQR, Backstepping, SMC, Adaptive optimal control and others obtained by combining the approaches. To overcome the under-actuated problem the adopted control consists of acting directly on the three Euler angles and the altitude using the four inputs. Two non-actuated variables are controlled by Virtual Inputs related to both pitch and roll angles. In Mokhtari et al. [2006], the authors use a feedback linearization leading to a linear extended system, based on the approach of Mistler et al. [2001]. Other strategies have been proposed using neural network Das et al. [2008], fuzzy logic approach Cosmin and Macnab [2006], dynamic inversion mechanism Das et al. [2009] or H_∞ robust control Raffo et al. [2011].

The most of the literature works, use for the control the same dynamic equations as the simulated model. However, it is not possible to know exactly this dynamic. Furthermore, to prove the robustness of controllers, the authors introduce disturbances during the simulation or, they assume uncertainties in some (but not all) parameters, neither errors on all the measured outputs Becker et al. [2012].

In B. Wang [2017], Islam et al. [2015], Emelyanov [2007] for example, the modelling error and disturbances uncertainties are considered by assuming knowledge of theirs bounds and model structure.

Sliding mode control approach, which is purely nonlinear, is dedicated to a wide spectrum of systems, especially for Variable Structure Systems Emelyanov [2007]. The outstanding feature of this controller is its robustness with regard to model error, parameters' uncertainties, external disturbances and noisy measurements, the reasons of which the SMC is adopted in our work. In addition to that, SMC is selected to be able to deal with arisen effects during the motors control loop M'Sirdi and Nadjar-Gauthier [2002].

From the most popular methods, the rotors speed is used as input reference for motor control by a linear loop Chan and Woo [2015]. In our case, a new contribution consists of using directly the thrust force as input reference for motor control. This means that the forces are controlled directly, which compensate errors due to non linearity of actuation, as what is done for pneumatic or hydraulic actuators Manamanni et al. [2001], M'Sirdi et al. [1997].

On the other side, the performance of SMC could be affected due to the succession of two sliding mode controllers in cascade (inner and outer control loops). This problem is avoided by considering two different convergence rate. To evaluate the effectiveness of the designed SMC, the Quadrotor performs trajectories tracking generated by visual servoing controller. This last controller provides a potential technique to fulfill motions of UAV thanks to visual measurement extracted from a captured image of an inboard camera. In the literature of UAV controlled by visual servoing task, there are two operating principles Hamel and Mahony [2006], Azrad et al. [2010];

- position-based visual servoing (PBVS) and
- image-based visual servoing (IBVS).

The implementation of IBVS is easier than PBVS since it is not necessary to know and reconstruct the 3D model of the target Chaumette and Hutchinson [2006].

This is one of the motivations to select IBVS method through this paper. Furthermore, the dynamic and the projection principle of IBVS controller are based on the virtual camera concept. Usually, the velocity trajectories generated by IBVS are compared with the velocities, mostly provided from Inertial Measurement Unit (IMU) sensor. The formulation background used is the Task Function approach Samson and Espiau [1990]. However, the IMU sensor exhibits bias and noise in the measurements which can cause a non-zero steady state error in image information Xie et al. [2017]. To overcome the problem, it is possible to obtain the velocity measurement only from the optical flow using the image information Mahony et al. [2017], Herisse et al. [2012] or to combine optical flow and IMU measurements Grabe et al. [2013].

This paper is organized as follows. We start with a full modelling of the Quadrotor in which all effects internal and external, forces and moments are considered. To simulate IMU measurements, we consider a dynamics for sensors taking into account a measurement bias and noise. In Section III, we explain different steps to design SMC. In Section IV, IBVS controller based on a virtual camera is detailed. SMC under IBVS controller for tracking task is simulated in Section V. Finally, Section VI reports some conclusions of the paper.

2. QUADROTOR UAV MODELLING

2.1. System description

The four rotors helicopter, shown in Fig. 1, is propelled by the four forces F_i $(i \in \{1,2,3,4\})$. The UAV is moved by varying the rotor speeds. The impair rotors (1, 3) turn in the same direction, which is in opposite directions of the pair ones (2, 4). This eliminates the anti-torque.

A variation of the rotor speeds altogether with the same quantity creates the lift forces which will affect the altitude z enabling vertical take-off or landing. The velocity speeding up or slowing down the diagonal motors creates the moment which produce a yaw or pitch motion. Yaw angle y is obtained by speeding up or slowing down the clockwise motors regard to the others. Pitch angle j axis allows the Quadrotor to move towards the longitudinal direction x. Roll angle f allows the Quadrotor to move towards the lateral direction y.

2.2. Actuators model

The thrust forces F_i are generated by 4 DC motors M$_i$ $(i \in \{1,2,3,4\})$ as depicted in Fig. 1. These forces are assumed to be proportional to the square of the angular motors speeds, denoted with ω_i and is given by:

$$F_i = b\omega_i^2 \tag{1}$$

where $b = \frac{1}{2}\rho\Lambda C_T r^2$ with ρ is the air density, r and Λ are the radius and the section of the propeller, respectively. The term C_T is the aerodynamic thrust coefficient. $\omega_{1,2,3,4}$ are the angular speeds of the four rotors. The aerodynamic drag torques δ_i produced at each actuator are opposed to the motor torque and proportional to the propeller angular speed.

$$\delta_i = \frac{1}{2}\rho\Lambda C_D r^2 \omega_i^2 = d\omega_i^2 \tag{2}$$

where C_D is the aerodynamic drag coefficient.

Therefore, these forces create different torques around the pitch ϕ, roll θ and yaw ψ axis that are respectively given as follows (Guerrero and Lozano [2012]; Austin [2010]):

$$\Gamma_\theta = u_2 = l\,(F_3 - F_1) \tag{3}$$

$$\Gamma_\phi = u_3 = l\,(F_4 - F_2) \tag{4}$$

$$\Gamma_\psi = u_4 = d\,(F_1 - F_2 + F_3 - F_4) \tag{5}$$

where d is a positive coefficient defined in Eq.2 and l is the distance between the generated force position and the gravity center of the Quadrotor.

Recall that the sum of the forces control the vertical motion z. So, the Quadrotor is controlled by varying the speed of these four motors. The control inputs of the

Quadrotor dynamics are thus defined as follows, with $u=(u_1,u_2,u_3,u_4)^T$:

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & -l & 0 & l \\ -l & 0 & l & 0 \\ d & -d & d & -d \end{pmatrix} \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{pmatrix} \tag{6}$$

u_1 denotes the first input force of the Quadrotor body in the z-axis. The inputs u_2 and u_3 represent the roll and the pitch input torques, respectively. The input u_4 represents the yawing control torque. These forces are provided through four Brushless DC motors which are characterized by a high torque and little friction (Austin [2010]). The dynamic model equation of all motors are reported in the next subsection.

2.3. Kinematics and Dynamics

The Quadrotor is described through the body-frame $\mathbf{R}_B(O,x_b,y_b,z_b)$ and earth-frame $\mathbf{R}_E(o,x_e,y_e,z_e)$ as shown in Fig. 1. Let us note $\xi=(x,y,z)^T$ the absolute Cartesian position of the Quadrotor Center of Gravity (CoG) relative to its fixed earth-frame $\mathbf{R}_E$ and, the Euler angles $\eta=(\phi,\theta,\psi)^T$ give its attitude relative to $\mathbf{R}_E$. The rotation matrix $\mathscr{R}:\mathbf{R}_E\to\mathbf{R}_B$ depends on the Euler angles (ϕ,θ,ψ) Mistler et al. [2001]. It is defined as follows:

$$\mathscr{R}(\phi,\theta,\psi) = \begin{pmatrix} c\psi c\theta & s\phi s\theta c\psi - s\psi c\theta & c\phi s\theta c\psi + s\psi s\phi \\ s\psi c\theta & s\phi s\theta s\psi + c\psi c\theta & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{pmatrix} \tag{7}$$

where $c=\cos(.)$ and $s=\sin(.)$ and under the stability limit constraints on the pitch $\phi\in\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$, the roll $\theta\in\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$, pitching and the yaw $\psi\in[-\pi,\pi]$ motions. The linear velocity of the UAV in the earth-frame $\mathbf{R}_E$ is denoted by the vector $v=(\dot{x},\dot{y},\dot{z})^T$ and is expressed as follows:

$$v=\mathscr{R}(\phi,\theta,\psi).v_B \tag{8}$$

where v_B is the linear velocity of the Quadrotor in the body attached frame.
Let us consider m as the total mass of the Quadrotor, g represents the gravity and l the distance from the center of each rotor to the CoG Mederreg et al. [2004].
Consider the vector $\vartheta=(p,q,r)^T$ which denotes the angular velocity in the frame $\mathbf{R}_B$. This vector can be transformed from the body frame $\mathbf{R}_B$ into the inertial one $\mathbf{R}_E$ as follows:

$$\vartheta = \begin{pmatrix} \dot{\phi} - s\theta\dot{\psi} \\ c\phi\dot{\theta} + s\phi c\theta\dot{\psi} \\ c\phi c\theta\dot{\psi} - s\phi\dot{\theta} \end{pmatrix} \tag{9}$$

So, we can deduce the angular velocities in the inertial frame which are given by the following transformation relationship:

$$\vartheta = \begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{pmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = T(\phi,\theta)\,\dot{\eta} \tag{10}$$

where T is the well known velocities' transformation matrix which is invertible. $\dot{\eta}=[\dot{\phi},\dot{\theta},\dot{\psi}]^T$

2.4. Modelling with Newton-Euler formalism

The variations of the propellers rotation speeds produce gyroscopic torques. There are two rotational motions of the Quadrotor body:

$$\mathbf{M}_{gp} = \sum_{i=1}^{4} \Omega \wedge \left(0,0,J_r(-1)^{i+1}\,\omega_i\right)^T \tag{11}$$

$$\mathbf{M}_{gb} = \Omega \wedge J\Omega \tag{12}$$

where Ω is the angular velocities vector in the fixed-frame, J_r is the propeller inertia for each rotor. The inertia matrix J of the Quadrotor body is defined as follows:

$$J = \begin{pmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{pmatrix} \tag{13}$$

Using the Newton-Euler formalism for modelling, the Newton's laws lead to the following motion equations of the Quadrotor:

$$\begin{cases} m\ddot{\xi} = \mathbf{F}_{th} - \mathbf{F}_d + \mathbf{F}_g \\ J\Omega = \mathbf{M} - \mathbf{M}_{gp} - \mathbf{M}_{gb} - \mathbf{M}_a \end{cases} \tag{14}$$

where $\mathbf{F}_{th}=R(\phi,\theta,\psi)(0,0,F)^T$ denotes the total thrust force of the four rotors, $\mathbf{F}_d=diag(\kappa_1,\kappa_2,\kappa_3)\,v_e^T$ is the air drag force which resists to the Quadrotor motion, $\mathbf{F}_g=(0,0,mg)^T$ is the gravity force, $\mathbf{M}=\left(\Gamma_\phi,\Gamma_\theta,\Gamma_\psi\right)^T$ represents the total rolling, pitching and yawing torques. The terms $\mathbf{M}_{gp}$ and $\mathbf{M}_{gb}$ are the gyroscopic torques and $\mathbf{M}_a=diag(\kappa_4,\kappa_5,\kappa_6)\,\vartheta^T$ is the torque resulting from aerodynamic frictions.
By substituting the position vector and the forces with their expressions into Eq. (14), we have the following translation dynamics of the Quadrotor:

$$\begin{cases} \ddot{X} = & -\frac{\kappa_1}{m}\dot{X} + \frac{1}{m}(c\phi c\psi s\theta + s\phi s\psi).u_1 \\ \ddot{Y} = & -\frac{\kappa_2}{m}\dot{Y} + \frac{1}{m}(c\phi s\psi s\theta - s\phi c\psi).u_1 \\ \ddot{Z} = & \frac{1}{m}c\phi c\theta u_1 - g - \frac{\kappa_3}{m}\dot{Z} \end{cases} \tag{15}$$

From the second part of Eq. (14), and while substituting each moment by its expression, we deduce the following rotational dynamics of the rotorcraft:

$$\begin{cases} \dot{p} = & \frac{J_{yy}-J_{zz}}{J_{xx}}.q.r - \frac{J_r}{J_{xx}}\omega_r.q - \frac{\kappa_4}{J_{xx}}.p + \frac{1}{J_{xx}}.u_2 \\ \dot{q} = & \frac{J_{zz}-J_{xx}}{J_{yy}}.p.r + \frac{J_r}{J_{yy}}\omega_r.p - \frac{\kappa_5}{J_{yy}}q + \frac{1}{J_{yy}}.u_3 \\ \dot{r} = & \frac{J_{xx}-J_{yy}}{J_{zz}}.p.q - \frac{\kappa_6}{J_{zz}}r + \frac{1}{J_{zz}}.u_4 \end{cases} \tag{16}$$

According to the established equations (10), (15) and (16), $x=\left(X,\dot{X},Y,\dot{Y},Z,\dot{Z},\phi,p,\theta,q,\psi,r\right)$ is retained as the state-space vector of the nonlinear model of the

Quadrotor rewritten as the following form $\dot{x} = f(x,u)$:

$$\dot{x} = \begin{cases} \dot{x}_1 = & x_2 \\ \dot{x}_2 = & a_9 x_2 + \frac{1}{m}(c(x_7)c(x_9)s(x_{11}) + s(x_7)s(x_1 1))u_1 \\ \dot{x}_3 = & x_4 \\ \dot{x}_4 = & a_{10} x_4 + \frac{1}{m}(c(x_7)s(x_9)s(x_{11}) - s(x_7)c(x_{11}))u_1 \\ \dot{x}_5 = & x_6 \\ \dot{x}_6 = & a_{11} x_6 + \frac{c(x_7)c(x_9)}{m} u_1 - g \\ \dot{x}_7 = & x_8 + x_{10}\, s(x_7) \tan(x_9) + x_{12}\, c(x_7) \tan(x_9) \\ \dot{x}_8 = & a_1 x_{10} x_{12} + a_2 x_8 + a_3 \omega_r x_{10} + b_1 u_2 \\ \dot{x}_9 = & x_{10}\, c(x_7) - x_{12}\, s(x_7) \\ \dot{x}_{10} = & a_4 x_8 x_{12} + a_5 x_{10} + a_6 \omega_r x_8 + b_2 u_3 \\ \dot{x}_{11} = & x_{10}\, s(x_7)\, sec(x_9) + x_{12}\, c(x_7)\, sec(x_9) \\ \dot{x}_{12} = & a_7 x_8 x_{10} + a_8 x_{12} + b_3 u_4 \end{cases} \tag{17}$$

where $a_1 = (J_y - J_z)/J_x$, $a_2 = -\kappa_4/J_x$, $a_3 = -J_r/J_x$, $a_4 = (I_z - I_x)/J_y$, $a_5 = -\kappa_5/J_y$, $a_6 = -J_r/J_y$, $a_7 = (I_y - I_x)/J_z$, $a_8 = -\kappa_6/J_z$, $a_9 = -\kappa_1/m$, $a_{10} = -\kappa_2/m$, $a_{11} = -\kappa_3/m$, $b_1 = -l/J_x$, $b_2 = -l/J_y$, $b_3 = -l/J_z$,
Note that $\kappa_{1,2,\ldots,6}$ are the aerodynamic friction and translational drag coefficients, $\omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4$ is the overall residual rotor angular speed.

2.5. Actuators Dynamics

The created aerodynamical torques and forces, given in Eq. (1) to Eq. (5), respectively, are provided through four Brushless DC motors which are characterize by a high torque and little friction (Austin [2010]). We consider that these motors have the same behavior of conventional DC motor at the steady state regime. Hence, the armature voltage of the i^{th} Brushless DC motor is defined as follows:

$$v_i = \frac{R_{mot}}{k_{mot}} J_r \dot{\omega}_i + k_{mot} \omega_i + d R_{mot} \omega_i^2 \tag{18}$$

where R_{mot} and k_{mot} denote the internal resistance and torque coefficient of the Brushless motors, respectively, d is the drag propellers' coefficient.
Since that the drag coefficient d is very small, this dynamic can be approximated as a first order lag transfer function where the characteristic parameters can be identified by experimental trials as shown in (Becker et al. [2012]).
So, the Quadrotor is controlled by varying the force generated by each motor, the other terms, by varying the motors speed.

2.6. Sensors Dynamics

The Quadrotor states are measured using an Inertial Measurement Unit (IMU) which contains accelerometers and gyroscope sensors (Guerrero and Lozano [2012], Becker et al. [2012], Austin [2010]). These give us measurements of the translational and rotational velocities. The translation and rotation outputs measurements along x, y, and z axes can be described by Eq. (19) and Eq. (20) respectively.

$$y^{acc} = \alpha^{acc} \nu_B + \beta^{acc} + \gamma^{acc} \tag{19}$$

$$y^{gyro} = \alpha^{gyro} \vartheta + \beta^{gyro} + \gamma^{gyro} \tag{20}$$

where $y^{acc} = (y_X^{acc}, y_Y^{acc}, y_Z^{acc})^T$ are the sensor outputs, $\alpha^{acc} = diag(\alpha_X^{acc}, \alpha_Y^{acc}, \alpha_Z^{acc})$ denote the accelerometer gains, $\nu_B = \mathscr{R}^{-1}(\phi, \theta, \psi)\, \nu$ presents the linear velocities in the body-frame, $\beta^{acc} = (\beta_X^{acc}, \beta_Y^{acc}, \beta_Z^{acc})^T$ are the sensor bias and $\gamma^{acc} = (\gamma_X^{acc}, \gamma_Y^{acc}, \gamma_Z^{acc})^T$ present the zero mean white noises.
where $y^{gyro} = \left(y_X^{gyro}, y_Y^{gyro}, y_Z^{gyro}\right)^T$ are the sensor outputs' voltage, $\alpha^{gyro} = diag\left(\alpha_X^{gyro}, \alpha_Y^{gyro}, \alpha_Z^{gyro}\right)$ are the gyroscope gains, $\vartheta = (p, q, r)^T$ denotes the angular velocities in the body-frame, $\beta^{gyro} = \left(\beta_X^{gyro}, \beta_Y^{gyro}, \beta_Z^{gyro}\right)^T$ are the sensor bias and $\gamma^{gyro} = \left(\gamma_X^{gyro}, \gamma_Y^{gyro}, \gamma_Z^{gyro}\right)^T$ present the zero mean white noises.

3. CONTROL STRATEGY

The Quadrotor control strategy can be split in four blocks or steps in the control loop. The first step (motors block) consists in controlling each motor separately to produce the propelling forces. Unlike the commonly used methods which control the angular speed of motors, here, we use the generated thrust F_i as the output force to be controlled from corresponding voltage input v_i of the motors (see Fig (3)).
In another hand, the Quadrotor contains four inputs u_i and six output position variables to be controlled, thus it is not possible to control all states separately at the same time. Generally, to overcome this problem, the procedure consists of controlling the cartesian positions x, y, z and the drone orientations angle ψ. The remaining two angles ϕ and θ are internal states which are not directly controlled. Their reference signals are deduced from the outputs in order to control the longitudinal and lateral directions, respectively x, y. Then, ϕ and θ, are controlled by an inner loop form the second step. Then x and y are controlled in the third step. The control of z and ψ is done directly in the third block.
The trajectories to be followed are generated by using IBVS approach, this defines the last control step. In the Fig. 2 we summarize the proposed control strategy of the whole system. This avoids the use of a joystick (or the human in the loop).

3.1. Actuator controllers

The most common motor controllers in the Quadrotor literature use the speed motors ω_i as reference signal to be controlled from input v_i, then the forces $F_i = \omega_i^2$ are generated. The speed output signal should be squared before being used as the force control signal (in the second step) since the thrust is assumed only proportional to the square of the motor speeds. This induces neglected dynamics in the propelling forces. This can be seen as inherent control perturbation.
Here, we propose that the forces F_i take place of the controlled signal instead of the motors speed ω_i. Then a simple Proportional Integral (PI) controller is widely efficient with a very small time response and zero steady

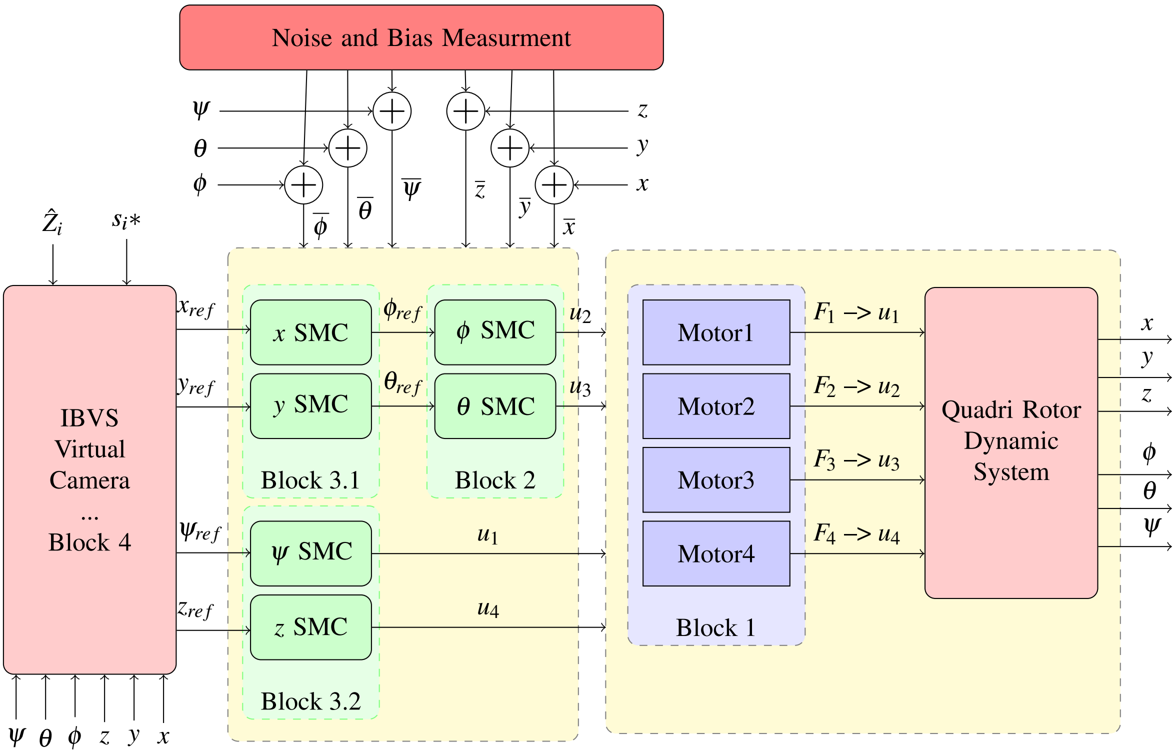


Figure 2: Control strategy of SMC using IBVS for Quadrotor

state error to track the desired propelling forces. The Fig. 3 depicts the scheme of the proposed motor control.

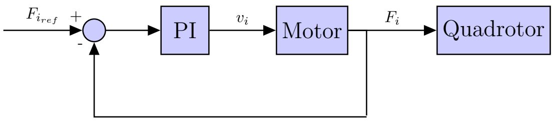


Figure 3: Motor control loop

3.2. Sliding Mode Control

The sliding mode control achieves the desired configuration in two steps M'Sirdi and Nadjar-Gauthier [2002]. The first is to drag all states toward this desired configuration called sliding surface $S(e,t)$ and enforce them to manifest around it Mederreg et al. [2005], Mokhtari et al. [2006]. Generally, the sliding surface of first order is given by:

$$S(e,t) = \dot{e} + \lambda e \tag{21}$$

where $e = x - x_d$ and λ is positive constant coefficient. The stabilizing control law leading to $S(e,t) = 0$ is deduced by using the Lyapunov function defined as follows:

$$V(e) = \frac{1}{2} S(e,t)^2 \tag{22}$$

As previously indicated, the control law is formed by two terms; the equivalent control law and the switching control law:

$$u = u_{eq} + u_s \tag{23}$$

The equivalent part is a continuous control law deduced from $\frac{\partial S(e,t)}{\partial t} = \dot{S}(e,t) = 0$ (using an available approximate model) and the second control part u_s has a discontinuous feature defined as in M'Sirdi and Nadjar-Gauthier [2002]:

$$u_s = -K\, sign(S(e,t)) \tag{24}$$

where K is a positive constant and *sign* is the sign function.

To design SMC for Quadrotor, we use an approximate model, rather than the defined one in Eq.17 for simulation, which parameters are not well known (parametric uncertainty).

$$\dot{\bar{x}} = \bar{f}(\bar{x},u) : \begin{cases} \dot{\bar{x}}_1 = \bar{x}_2 \\ \dot{\bar{x}}_2 = \bar{a}_9 \bar{x}_2 + \frac{1}{m} u_x u_1 \\ \dot{\bar{x}}_3 = \bar{x}_4 \\ \dot{\bar{x}}_4 = \bar{a}_{10} \bar{x}_4 + \frac{1}{m} u_y u_1 \\ \dot{\bar{x}}_5 = \bar{x}_6 \\ \dot{\bar{x}}_6 = \bar{a}_{11} x_6 + \frac{c(\bar{x}_7)c(\bar{x}_9)}{m} u_1 - g \\ \dot{\bar{x}}_7 = \bar{x}_8 \\ \dot{\bar{x}}_8 = \bar{a}_1 \bar{x}_{10} \bar{x}_{12} + \bar{a}_2 \bar{x}_8 + \bar{a}_3 \omega_r \bar{x}_{10} + \bar{b}_1 u_2 \\ \dot{\bar{x}}_9 = \bar{x}_{10} \\ \dot{\bar{x}}_{10} = \bar{a}_4 \bar{x}_8 \bar{x}_{12} + \bar{a}_5 \bar{x}_{10} + \bar{a}_6 \omega_r \bar{x}_8 + \bar{b}_2 u_3 \\ \dot{\bar{x}}_{11} = \bar{x}_{10} \\ \dot{\bar{x}}_{12} = \bar{a}_7 \bar{x}_8 \bar{x}_{10} + \bar{a}_8 \bar{x}_{12} + \bar{b}_3 u_4 \end{cases} \tag{25}$$

where $\bar{x}_i$ are biased measurements perturbed by noise, $(\bar{a}_i, \bar{b}_i)$ are the approximated model parameters:

$$(\bar{a}_i, \bar{b}_i) = (a_i, b_i) \pm 25\%(a_i, b_i)$$

SMC in altitude and attitude control uses u_i as control input to track the desired sliding surface. For example, the sliding surface as defined in Eq.21 for the altitude is given as:

$$S_z = \bar{x}_6 - x_{d_6} - \lambda_z(\bar{x}_5 - x_{d_5}) \tag{26}$$

Then the equivalent control u_{eq} is obtained from the condition:

$$\begin{aligned} \dot{S}_z = 0 \quad & \Leftrightarrow \bar{a}_{11}\bar{x}_6 + \tfrac{c(\bar{x}_7)c(\bar{x}_9)}{m} u_1 - g - \dot{x}_{d_6} + \lambda_z(\bar{x}_6 - x_{d_6}) = 0 \\ u_{eq_z} &= \tfrac{m}{c(\bar{x}_7)c(\bar{x}_9)}(g + \dot{x}_{d_6} - a_{11}\bar{x}_6 - \lambda_z(\bar{x}_6 - x_{d_6})) \end{aligned} \tag{27}$$

The VSS (Variable Structure Control) control is given by M'Sirdi and Nadjar-Gauthier [2002]:

$$u_{s_z} = -k_z sign(S_z) \tag{28}$$

The complete control law can be written according to Eq.23:

$$u_1 = \frac{m}{c(\bar{x}_7)c(\bar{x}_9)}(g + \dot{x}_{6_d} - a_{11}\bar{x}_6 - \lambda_z(\bar{x}_6 - x_{6_d})) - k_z sign(S_z)$$

The same steps are followed to extract other control laws:

$$\begin{aligned} u_2 = \quad & \tfrac{1}{\bar{b}_1}(\dot{x}_{8_d} - \bar{a}_1\bar{x}_{10}\bar{x}_{12} - \bar{a}_2\bar{x}_8 - \bar{a}_3\omega_r\bar{x}_{10} - \lambda_\phi(\bar{x}_8 - x_{8_d})) \\ & -k_\phi sign(S_\phi) \end{aligned}$$

$$\begin{aligned} u_3 = \quad & \tfrac{1}{\bar{b}_2}(\dot{x}_{10_d} - \bar{a}_4\bar{x}_8\bar{x}_{12} - \bar{a}_5\bar{x}_{10} - \bar{a}_6\omega_r\bar{x}_8 - \lambda_\theta(\bar{x}_{10} - x_{10_d})) \\ & -k_\theta sign(S_\theta) \end{aligned}$$

$$u_4 = \frac{1}{\bar{b}_3}(\dot{x}_{12_d} - \bar{a}_7\bar{x}_8\bar{x}_{10} - \bar{a}_8\bar{x}_{12} - \lambda_\psi(\bar{x}_{12} - x_{12_d})) - k_\psi sign(S_\psi)$$

The x, y directions are controlled by the virtual control law u_x and u_y, respectively.

$$u_x = \frac{m}{u_1}(\dot{x}_{2_d} - \bar{a}_9\bar{x}_2 - \lambda_x) - k_x sign(S_x)$$

$$u_y = \frac{m}{u_1}(\dot{y}_{4_d} - \bar{a}_{10}\bar{x}_4 - \lambda_y) - k_y sign(S_y)$$

Unlike x_d, y_d, z_d, ψ_d which are derived from the visual servoing task, ϕ_d and θ_d are obtained from both virtual control laws u_x, u_y as follows:

$$\begin{bmatrix} \phi_d \\ \theta_d \end{bmatrix} = \begin{bmatrix} \sin(\psi_d) & \cos(\psi_d) \\ -\cos(\psi_d) & \sin(\psi_d) \end{bmatrix}^{-1} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \tag{29}$$

4. IBVS CONTROLLER

IBVS is a controller essentially based on visual data extracted from a camera. This approach has been emerged form Task Oriented Feedback Approach (see Samson and Espiau [1990] in the aim of controlling robot manipulators. This is called Eye-in-hand or Eye-to-hand, depending to the location of the camera with respect to robot Hutchinson et al. [1996]. In Vision Servoing, the control strategy is based on a cost function minimization Chaumette and Hutchinson [2006].
As task function cost we can take:

$$e_i(t) = s_i(t) - s_i^* \tag{30}$$

where $s_i(t)$ and s_i^* are the image measurement and desired configuration in the image plane, respectively. These measurements are defined in pixels $s_i = (p_{x_i}, p_{y_i})^T$ and they are used to determine the corresponding point in image frame, denoted by $p_i = (x_i, y_i)$. Relationship between both points is defined by applying the inverse of the camera intrinsic matrix:

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} f_{xx} & f_{xx}\alpha & x_0 \\ 0 & f_{yy} & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{x_i} \\ p_{y_i} \\ 1 \end{bmatrix} \tag{31}$$

In the Eye-in-hand approach adopted in this work, the relationship between the projection measurement in the image frame p_i and the target point defined in the camera frame, denoted by $P_{c_i} = (X_{c_i}, Y_{c_i}, Z_{c_i})$ is derived from the Pinhole camera model:

$$\begin{cases} x_i = \frac{X_{c_i}}{Z_{c_i}} \\ y_i = \frac{Y_{c_i}}{Z_{c_i}} \end{cases} \tag{32}$$

When the camera is moving with $V_c = [\upsilon_c, \omega_c]^T = [\upsilon_x, \upsilon_y, \upsilon_z, \omega_x, \omega_y, \omega_z]^T$, the dynamics of P_{c_i} are given by applying the following formula Chaumette and Hutchinson [2006]:

$$\dot{P}_{c_i} = \upsilon_{c_i} + \omega_{c_i} \times P_{c_i} \tag{33}$$

with × is the cross product. The dynamics of the point in image plane is taken from the derivative of Eq.32:

$$\begin{cases} x_i = \frac{\dot{X}_{c_i}}{Z_{c_i}} - \frac{X_{c_i}\dot{Z}_{c_i}}{Z_{c_i}^2} \\ y_i = \frac{\dot{Y}_{c_i}}{Z_{c_i}} - \frac{Y_{c_i}\dot{Z}_{c_i}}{Z_{c_i}^2} \end{cases} \tag{34}$$

Using Eq.33 and Eq.34, the following results are obtained:

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \mathbb{L} \begin{bmatrix} \upsilon_c \\ \omega_c \end{bmatrix} \tag{35}$$

where $\mathbb{L}$ is the interaction matrix:

$$\mathbb{L} = \begin{bmatrix} \frac{1}{Z_{c_i}} & 0 & -\frac{x_i}{Z_{c_i}} & -x_i y_i & 1 + x_i^2 & -y_i \\ 0 & \frac{1}{Z_{c_i}} & -\frac{y_i}{Z_{c_i}} & -(1 + y_i^2) & x_i y_i & x_i \end{bmatrix} \tag{36}$$

It is clear, to compute all components of the camera velocity, we need more than three points with different depth. If the feature points to be tracked are co-planar, we need at least four points.

4.1. Trajectory generation based on visual tracking

The key component of visual servoing controller is the interaction matrix. The common approach considered in IBVS control problem is based on the choice of an exponential decreasing of the cost function:

$$\dot{e}_i = -\delta e_i(t) \tag{37}$$

where the decreasing rate δ is a positive constant. Under the assumption that the desired configuration is constant and using the Eq.35, the derivative of the error is:

$$\dot{s}(t) = -\delta e(t) = \mathbb{L} V_c \tag{38}$$

This leads to the camera velocity:

$$V_c = -\delta \mathbb{L}^{+} e(t) \tag{39}$$

where $\mathbb{L}^{+} \in \mathbb{R}^{6 \times 2k}$ is a pseudo-inverse matrix with $k \geq 4$ as the image number of points to be tracked. If only one camera is used, is it not possible to define exactly the depth Z_{c_i}. Consequently, the interaction matrix is not well defined. To overcome this problem, this matrix can be estimated or approximated by fixing z_c to its value of the desired configuration.

4.2. IBVS using a virtual camera

It is expected that the IBVS controller generates six desired velocities based on the observed object. The trajectories are defined in the camera frame and should be transformed to the Quadrotor frame. Under assumption of any translation and rotation between virtual camera and Quadrotor frames, the generated trajectories are directly used by Quadrotor Controller.

However, since the pitch and roll angles in Quadrotor are implicitly used to drive the motion in x and y direction, ω_x and ω_y they are not addressed for tracking task. This might cause disappearance of the target from the view field of the camera. The proposed solution is to use saturation in pitch and roll variations. The final process to obtain Quadrotor references is defined as:

$$[\dot{X}_{ref}, \dot{Y}_{ref}, \dot{Z}_{ref}] = \mathscr{R}[\upsilon_x, \upsilon_y, \upsilon_z] \tag{40}$$

and for rotation:

$$[\dot{\phi}_{ref}, \dot{\theta}_{ref}, \dot{\psi}_{ref}] = T(\phi, \theta)\,[\omega_x, \omega_y, \omega_z] \tag{41}$$

5. SIMULATION RESULTS

In order to verify the performance of the control strategy, the simulations was developed under Matlab/Simulink software. Various simulations are conducted to achieve the final task. The Quadrotor and motors parameters used in all our simulation are reported in Table.1

The parameters of the sensor dynamic applied to all outputs are chosen as follows: $\alpha^{acc} = \alpha^{gyro} = 1$, $\beta^{acc} = \beta^{gyro} = 10^{-3}$. As for the noise, we have used the Simulink Band-Limited White Noise block generator with noise power equal to 10^{-4}.

For the visual servoing task, intrinsic parameters of the virtual camera are inspired from a real camera parameters; $(f_{xx}, f_{yy}) = (657.4, 657.8)$, $\alpha = 10^{-4}$, $(x_0, y_0) = (303, 243)$. During the simulation, the depth Z_{c_i} defined in interaction matrix was fixed to be equal to $z_{fd} = 0.8$.

First, the trajectory generation based on visual servoing and SMC control for Quadrotor were tested separately (see figures 4). Under the assumption the camera frame superposes the Quadrotor frame, the desired target in image plane corresponds to the desired final pose $([x_{fd}, y_{fd}, z_{fd}d, \phi_{fd}, \theta_{fd}, \psi_{fd}] = [0, 0, 0.8, 0, 0, \frac{pi}{4}])$. The evolution of the virtual camera frame and the generated trajectories using IBVS without Quadrotor is shown in Fig.4. The considered target is a plane with four points.

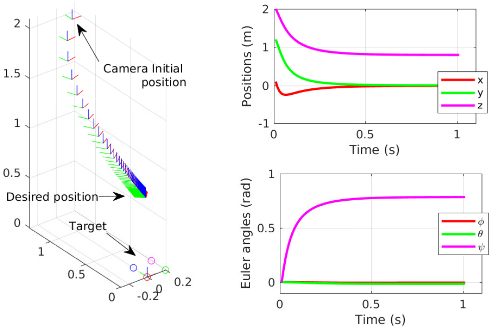


Figure 4: Virtual camera frame evolution in IBVS generation trajectories without Quadrotor

Now, we apply these trajectories to Quadrotor using SMC. For all the simulation tests, the SMC control is based on an approximate model (with less dynamics) and with uncertainties in all the a priori estimated parameters. After many tests, the SMC gain parameters are selected to be $k_x = k_y = 0.9$, $k_z = k_\phi = k_\theta = 0.5$ and $k_\psi = 0.01$.

The visual servoing trajectory generation is supposed launched from the following initial pose $[x_0, y_0, z_0, \phi_0, \theta_0, \psi_0] = [0.3, -0.3, 1.2, 0, 0, 0]$. However, this pose is different to take-off pose of Quadrotor. Then, before launching the visual servoing task, the Quadrotor performs the path between both poses. In simulation, this step takes the first $20s$.

Simulation results with a sliding mode controller by performing a visual tracking are shown in Fig.5 and Fig.6 which shows the positions and the Euler anglers, respec-

Parameter	Value	Parameter	Value
m	0.486	g	0.486
l	0.25	b	2.9842e-5
d	3.2320e-7	J_x	2.8385e-5
J_y	2.8385e-5	J_z	2.8385e-5
κ_1	5.5670e-4	κ_2	5.5670e-4
κ_3	6.3540e-4	κ_4	5.5670e-4
κ_5	5.5670e-4	κ_6	6.3540e-4
R_{mot}	6.3540e-4	J_r	2.8385e-5
k_{mot}	$20J_r$		

Table 1: Quadrotor parameters values

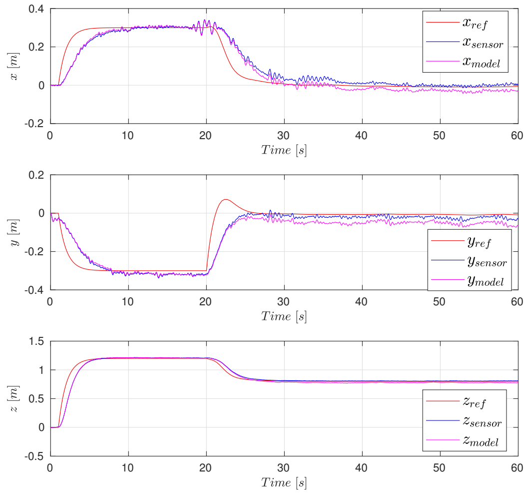


Figure 5: Time evolution of reference trajectories, outputs issue from sensors and model outputs in x,y and z directions

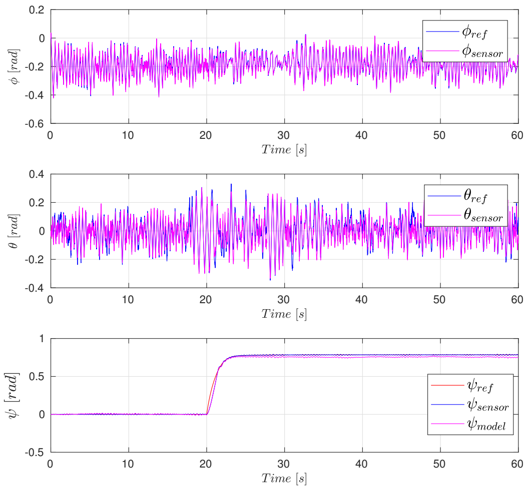


Figure 6: Time evolution of reference trajectories, outputs issue from sensors for ϕ and θ angles. In addition, output model for ψ

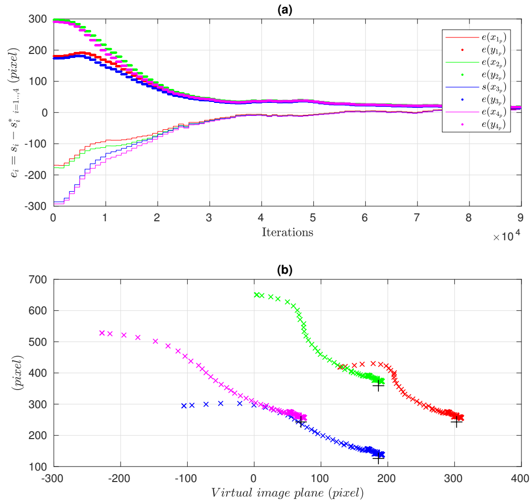


Figure 7: Tracking errors evolution of all points in image pixel measurement (a). Tracking evolution in image plane (b)

tively. The effect of the sensors dynamic and perturbation on the outputs of Quadrotor can be seen clearly.

Despite, the proposed approach shows a considerable ability to reject the perturbations. Since the yaw angle and the altitude are controlled directly, they achieve the desired reference in shorter time with less oscillations with respect to x and y directions. This is mainly due to the strong coupling effects between the blocks in the control scheme of Fig.2.

In addition, the peak shown is time evolution in y direction is generated by convergence rate coefficient δ in the visual servoing task. To reduce this effect, we can use a weighting matrix rather than using a scalar coefficient to damp more or less the convergence rate along this direction. Thus, instead of using δ, we use $\Delta_{6\times6} = diag\{\delta_1, ..., \delta_6\}$. However, if δ_2 is selected to be so small, this can produce a slight static error in this direction.

A well known drawback of SMC is the chattering effects, which appears in roll and pitch time evolution. The convergence criteria can be also evaluated based on the visual data. Fig.7.a shows the error evolution of each coordinate of all points and Fig.7.b is the evolution of each point in the image plane of the virtual camera. Both figures show an error equal to ± 10 *pixels* which is considered to be within an acceptable variation range. Basically, this error is caused by the measurement bias of the sensors.

Through the last simulation shown in Fig.8, we try to illustrate the effectiveness of our approach when the target is moving. Here, the center of target performs a circular motion in (x,y) plan. Despite that the Quadrotor follows the circular motion, there is a small delay in time response. This is due to neglected term of $\frac{\partial s(t)^*}{\partial t}$ in task function which becomes different to zero.

6. CONCLUSION

This paper proposes a robust control strategy for a Quadrotor UAV system endowed with a virtual camera.

To simulate all effects that can influence the evolution of the Quadrotor in real environment, we have developed a dynamic model taking into account physical parts and aerodynamics phenomena, as well as different hardware components such as the actuators with their controller (force control) and sensors dynamics and perturbations.

In simulations, we remark that bias of sensors cause gaps between model outputs and sensors measurements. To overcome the nonlinear effects and for robustness, a sliding mode control has been developed and tested with use of the IBVS. Here, the SMC uses an approximate dynamic (with less dynamics) in the control model and uncertainties in all the control model parameters.

To verify the robustness of the proposed controller, the

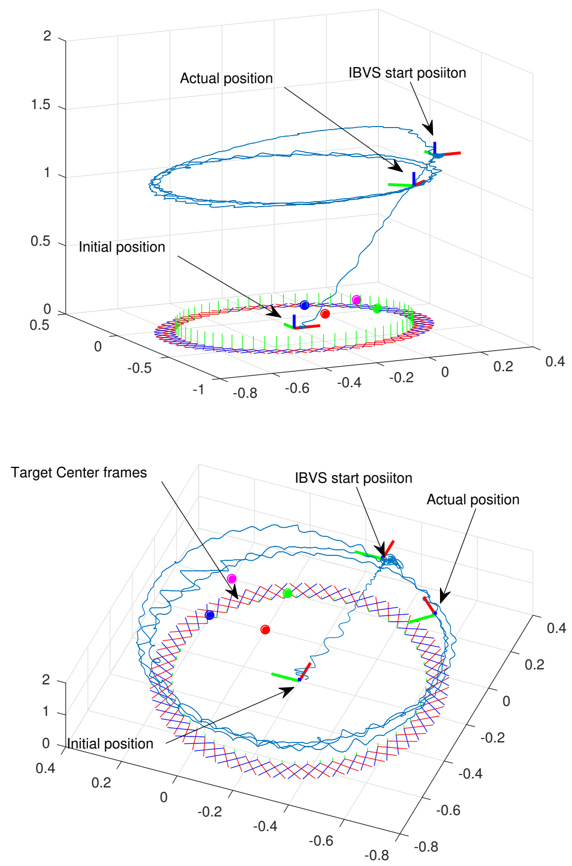


Figure 8: Tracking behavior for circular trajectory

Quadrotor tracks trajectories generated by visual servoing task using a virtual camera. The obtained simulation results show the efficiency of the proposed controller to track the generated trajectories in spite of the complexity and the nonlinearities of the whole system.
The proposed control approach is also tested with a moving target for evaluation of the tracking capabilities. As prospects, we hope to reduce the shattering effect by adopting a smooth sign function or by using sliding mode control with a higher order. As for the engendered error in image plane, we hope to combine sensor measurements with the optical flow from image information to well estimate the Quadrotor movements.

REFERENCES

R. Austin. *Unmanned Aircraft Systems: UAVs design, development and deployment*. 1st edition, John Wiley and Sons Ltd, London, 2010.

S. Azrad, F. Kendoul, and K. Nonami. Visual servoing of quadrotor micro-air vehicle using color-based tracking algorithm. *J. Syst. Des. Dyn.*, 4(2), 2010.

Y Zhang B. Wang, L. Mu. Adaptive robust tracking control of quadrotor helicopter with parametric uncertainty and external disturbance. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017.

M. Becker, B. Coronel, R. Sampaio, S. Bouabdallah, V. De Perrot, and R. Siegwart. In flight collision avoidance for a mini-uav robot based on onboard sensors. *J. of the Brazilian Society of Mechanical Sciences and Engineering*, 34(3):294–307, 2012.

S. Bouabdallah. *Design and control of quadrotors with application to autonomous flying*. PhD thesis, École Polytechnique fédérale de Lausanne, 2007.

H.L. Chan and K.T. Woo. Design and control of small quadcopter system with motor closed loop speed control. *International Journal of Mechanical Engineering and Robotics Research*, 4(3), 2015.

F. Chaumette and S. Hutchinson. Visual servo control part i: basic approaches. *IEEE Robot. Autom. Mag*, 13 (4), 2006.

C. Cosmin and J. B. Macnab. A new robust adaptive-fuzzy control method addaplied to quadrotor helicopter stabilization. In *Fuzzy Information Processing Society, Annual Meeting of the North American, NAFIPS*, pages 454–458, 2006.

A. Das, F. Lewis, and S. Subbarao. Dynamic neural network based robust backstepping control approach for quadrotors. In *Proc. of the AIAA Guidance, Navigation and Control Conference and Exibit , Hawaii, USA*, 2008.

A. Das, K. Subbarao, and F. Lewis. Dynamic inversion with zero-dynamics stabilisation for quadrotor control. *IET Control Theory Application*, 3(3):303–314, 2009.

S.V. Emelyanov. Theory of variable-structure control systems: inception and initial development. *Computational Mathematics and Modeling*, 18(4), 2007.

V. Grabe, H.H. Bulthoff, and P.R. Giordano. A comparison of scale estimation schemes for a quadrotor uav based on optical flow and imu measurements. In *IEEE - RSJ Int. Conf. on Intelligent Robots and Systems, IROS*, 2013.

A. Guerrero and R. Lozano. *Flight formation control*. John Wiley and Sons Inc, New York, 2012.

T. Hamel and R. Mahony. Visual servoing of an underactuated dynamic rigid-body system: an image-based approach. *IEEE Trans. Robot. Autom*, 18(2), 2006.

B. Herisse, T. Hamel, R. Mahony, and F.X. Russotto. Landing a vtol unmanned aerial vehicle on a moving platform using optical flow. *IEEE Trans. Robot*, 28(1), 2012.

S. Hutchinson, G.D. Hager, and P.I. Corke. A tutorial on visual servo control. *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*, 12(5), 1996.

S. Islam, P. X. Liu, and A. El Saddik. Robust control of four-rotor unmanned aerial vehicle with disturbance uncertainty. *IEEE Transactions on Industrial Electronics*, 62(3):1563–1571, March 2015. ISSN 0278-0046. doi: 10.1109/TIE.2014.2365441.

R. Lozano. *Unmanned aerial vehicles: Embedded control*. John Wiley and sons, 2013.

R. Mahony, P. Corke, and T. Hamel. Dynamic image-based visual servo control using centroid and optic flow features. *J. Dyn.Syst-T. ASME*, 130, 2017.

N. Manamanni, M. Djemai, T. Boukhobza, and N.K. M'Sirdi. Nonlinear sliding observer based control for a pneumatic robot leg. *International Journal of Robotics and Automation*, 16:100–112, 01 2001.

L. Mederreg, F. Diaz, and N.K. M'Sirdi. Nonlinear backstepping control with observer design for 4 rotors helicopter. In *Proceedings of AVCS 2004*, Genova, 2004.

L. Mederreg, F. Diaz, and N.K. M'Sirdi. Dynamic feedback control for a quadrotor unmanned aerial vehicle. In *SSD 2005*, Sousse, Tunisia, 2005.

V. Mistler, A. Benallegue, and N.K. M'Sirdi. Exact linearization and noninteracting control of a 4-rotors helicopter via dynamic feedback. In *ROMAN 10th IEEE Int. Workshop on Robot-Human Interactive Communication, Bordeaux*, pages 586–593, Bordeaux and Paris, 2001. ISBN 0-7803-7222-0. doi: 10.1109/ROMAN.2001.981968.

V. Mistler, A. Benallegue, and N.K. M'Sirdi. Linéarisation exacte et découplage entrées-sorties, comparaison entre l'hélicoptère standard et l'hélicoptère 4 rotors. In *Proceedings of the CIFA 2002*, Nantes, 2002.

A. Mokhtari, N.K. M'Sirdi, K. Meghriche, and A Belaidi. Feedback linearization and linear observer for a quadrotor unmanned aerial vehicle. *Advanced Robotics*, 20(1):71–91, 2006. doi: DOI:10.1163/156855306775275495. URL https://doi.org/10.1163/156855306775275495.

N.K M'Sirdi and N. Nadjar-Gauthier. *Application of Sliding Mode Control to Robotic Systems*, chapter 13, pages 351–387. Control Engineering Series. Marcel Dekker edited by Wilfrid Perruquetti and Jean Pierre Barbot, New York, 2002. URL https://books.google.fr/books?isbn=0203910850.

N.K. M'Sirdi, P. Fraisse, P. Dauchez, and N. Manamani. Sliding mode control for a hydraulic underwater manipulator. In *Syroco'97*, 1997.

A. Rabhi, M. Chadli, and C. Pegard. Robust fuzzy control for stabilization of a quadrotor. In *International Conference on Advanced Robotics*, pages 471–475, 2011.

G.V. Raffo, M. G Ortega, and F. R. Rubio. Nonlinear hinfinity controller for the quad-rotor helicopter with input coupling*. *IFAC Proceedings Volumes*, 44 (1):13834 – 13839, 2011. ISSN 1474-6670. doi: https://doi.org/10.3182/20110828-6-IT-1002.02453. URL http://www.sciencedirect.com/science/article/pii/S1474667016458477. 18th IFAC World Congress.

C. Samson and B. Espiau. Application of the task-function approach to sensor-based control of robot manipulators. *IFAC Proceedings Volumes*, 23 (8, Part 5):269 – 274, 1990. ISSN 1474-6670. doi: https://doi.org/10.1016/S1474-6670(17)51746-2. URL http://www.sciencedirect.com/science/article/pii/S1474667017517462. 11th IFAC World Congress on Automatic Control, Tallinn, 1990 - Volume 5, Tallinn, Finland.

H. Xie, K.H. Low, and Z. He. Adaptive visual servoing of unmanned aerial vehicles in gps-denied environments. *IEEE/ASME Trans.Mechatron*, 22(6), 2017.

ACKNOWLEDGMENT

This work is supported by the SASV research group and its funding. The research project was initiated and driven by N.K. M'Sirdi several years ago in the LRV (Robotics Laboratory of university of Versailles Saint Quentin). It is now, for SASV in a collaboration with the LAT of Tlemcen and other partners.

AUTHORS BIOGRAPHY

Choukri Bensalah is a post Doc at the LIS research laboratory of the University of Aix Marseille University, (LIS UMR CNRS 7020). His is assistant professor at University of Tlemcen, Algeria, he is member of "Laboratoire d'automatique de Tlemcen" since 2016. He was Graduated in "chaotic systems" from Tlemcen University (2007) and received the Ph.D from the Robotics Lab of the University Carlos III of Madrid (2014). His general research interests include modelling and simulation of dynamic systems, robotics and vision applications and control theory.

Nacer K. M'Sirdi is professor at Polytech Marseille and Aix Marseille University (AMU). He got the Phd in Electronics at ENSERG-INPG (1983) and the Doctorat d'Etat in adaptive signal processing at the ENSIEG-INPG, in 1988 (LAG). He was assistant professor, in University of Paris 6 in 1987 and Professor at University of Versailles in 1993. From 2005 up to now, he is a research member of the LIS (UMR CNRS 7020). In 2009 he has created the VSAS research project on Variable Structure Automatic Systems (SASV), for research in automatic control and optimization of VSS Systems with commutations.

Aziz Naamane is an Associate Professor at the Aix Marseille University, he is a member of Laboratoire d'Informatique et des Systmes (LIS UMR CNRS 7020). His main research activities deal with discrete event modelling and simulation, optimal control, diagnosis, vehicle dynamics and renewable energy. He is Member (co advisor) of the SASV (Variable Structure Automatic Systems) research Group and the HyRES Lab. His research field is now focused on Prototyping, Design and control of embedded power electric systems.

PATH TRACKING FOR THE CONVOY OF AUTONOMOUS VEHICLES BASED ON A NON-LINEAR PREDICTIVE CONTROL

M. Mahmoud Mohamed Ahmed(a), **Aziz Naamane**(b), **Nacer K. M'Sirdi**(c)

(a), (b), (c)Aix Marseille University, Université de Toulon, CNRS, LIS UMR 7020, SASV, Marseille, France

(a)mohamed-mahmoud.mohamed-ahmed@lis-lab.fr, (b)aziz.naamane@lis-lab.fr, (c)nacer.msirdi@lis-lab.fr

ABSTRACT

In this paper, a nonlinear predictive control of a platoon of several vehicles is proposed by using non-linear robotic form model of the vehicles. The model used represents the longitudinal, lateral and yaw movement for each vehicle in the fleet. this control approach allows controlling the fleet, uses the available information, ensures a safe distance between vehicles to avoid collisions and follows the path of the leader. The robustness of the control will be studied in order to assess the different errors occurring in the estimated parameters values.

1. INTRODUCTION

The explosion in the number of vehicles put into circulation each year in the world poses problems for road infrastructures today. There is also air pollution and the safety of people. Today, peri-urban networks are affected by recurrent congestion phenomena, due to the increasing number of urban-urban journeys. How to increase the capacity of the infrastructures, while improving the safety and the comfort of the motorists? Solutions can then be considered: better use of available space by automating vehicles at low speeds or streamline all travel. The first strategy led to the behavioral study of inter-vehicular distances Ali et al. [2015], Nouveliere et al. [2002]. In the field of road transport, the constraints related to the safety and the capacity of the traffic lanes make the knowledge of inter-vehicle distances and possibly their control necessary.

The vehicle fleet is a very efficient means of transportation for passengers, merchandise and increased traffic capacity. For example, a convoy of trucks carries goods, with a single driver Ali [2015]. Other benefits such as reducing fuel consumption and minimizing manpower. The convoy is composed of a vehicle in the head and other cars are followers. The leader vehicle can be autonomous or driven by a driver, the other vehicles follow the leader with a safety distance to avoid collisions between vehicles. Two spacing approaches for the safety distance between vehicles have been proposed in the literature; an established distance and a distance proportionally with the speed Swaroop [1994] . Nouveliere et al. [2002]. For a longitudinal displacement, the distances are constant. For overall control of longitudinal and lateral movements, the distance between vehicles can be proportional to speed and depends on the reference path for lateral deviation.

Several control approaches have been proposed in the literature for vehicle fleets, in Ali et al. [2015] a linear dual integrator dynamic model is used after an exact linearization for a vehicle convoy. The longitudinal movement is controlled with a linear control to ensure a safe distance between the vehicles. The lateral movement is controlled by acting on the vehicle orientation angle with respect to the desired trajectory. Longitudinal and lateral control are independent. Another control approach has been proposed in Xiang and Bräunl [2010] which represents a distribution algorithm based on the relative error of the previous vehicle, position, the vehicle model used for this approach is the kinematic one.

In order to obtain precise data, sensors are placed on board of each vehicle of the fleet, for this mission. The local strategy is based on data or information that are shared between close neighbors. In the literature, most Leader-Follower control approaches belong to this category Avanzini et al. [2010], Avanzini [2010]. The Leader vehicle can move autonomously to follow a desired path. It serves as a target or reference for the vehicle following it. Each vehicle in the convoy group plays the role of Leader for the vehicle following it (the follower). The driven vehicle is dependent of the data of its predecessor, the control architecture here is unidirectional. The global architecture uses the information of all the vehicles of the convoy as the state of the leader and neighboring vehicles, for example, the control referenced on the previous vehicle and the leader. This control approach is divided into two categories, using either a centralized or decentralized architectures. For the centralized architecture, the control law applied to each vehicle in the fleet is based on the data of all vehicles in the convoy Yazbeck [2014]. On the other hand, the decentralized architecture is based on the data of a part of the convoy, to minimize the numbers of the sensors used. A review on modelling and control strategies has been presented in M'Sirdi [2018].

Several convoy project are realized in the literature that are based on these approaches of control, we can mention the AutoNet2030 project for a self-driving vehicle cooperation system and a manual drive based on the decentralized approach to make their decision. The control laws are based on the information of the neighbors. And the SARTE project funded by the European Union in 2012 with the aim of driving a convoy of vehicles with high speed on a motorway without modifying the infrastructure. The control law applied on each vehicle of the convoy is based on the decentralized global approach, such that the leader information and the neighbors used to build this control Avanzini [2010]. Another important project is the one called Chauffeur, which deals with the conveying of trucks. The leading vehicle was controlled manually by a driver and the other vehicles (trucks) automatically follow the truck ahead.

In this work, we propose a coupled longitudinal and lateral control of a fleet of autonomous vehicles using non-linear predictive control. The model used for this control approach represents the non-linear robotic form model of a vehicle . The model represents the longitudinal and lateral movement of the fleet and the movement of the yaw. The longitudinal movement of the fleet is controlled by the driving/braking wheels torque and the lateral movement is controlled by the steering angle. In this control approach, the model of the fleet is not linearized. The kinematic model will be considered for moving the fleet in the reference frame. The lateral control of the fleet is coupled with the longitudinal movement according to the speed, as the lateral movement is controlled by imposing a lateral acceleration and this desired acceleration is calculated according to the longitudinal speed and the reference trajectory for each vehicle of the convoy. The overall control of the convoy makes it possible to follow a desired trajectory for the convoy and to ensure a safe distance between the vehicles of the fleet to avoid collisions.

2. MODELING

The dynamic model is considered in this part to control the fleet by the efforts that are applied for each movement of the convoy.

2.1. Dynamic model

Several methods of modeling can be found in the literature to determine the model of a vehicle. These different methods lead to sets of equations that represent the dynamic motion of the vehicle DeSantis [1995] Jaballah [2011] Rabhi [2005]. The dynamic model used of a vehicle was determined using the robotic formalism Chebly [2017]. The vehicle is represented in the figure 1 with the following variables in (G, x, y) the vehicle reference frame. G is the gravity center.

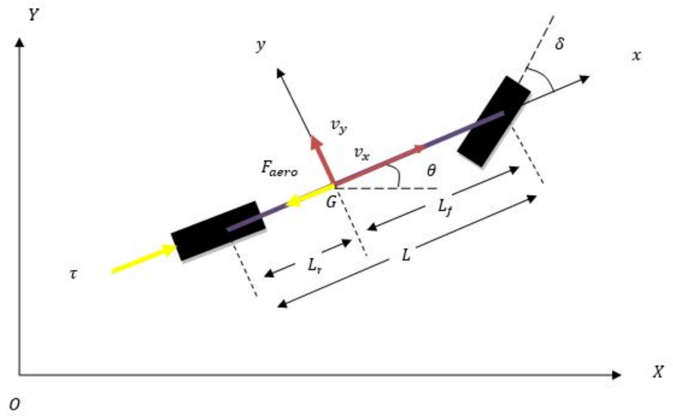


Figure 1: The Vehicle Description

L_f is distance from the front wheel to G .
L_r: is the distance from the rear wheel center to G.
m, I_z: the mass and Inertia Moment of the vehicles.
m_w, I_w :the mass and the rotational inertia of the wheel.
$\dot{x}, v_x$: longitudinal vehicle velocity along x axis.
$\dot{y}, v_y$: lateral velocity (axis y).
θ: yaw angle and $\dot{\theta}$: yaw rate.
$a_x = \ddot{x} - \dot{y}\dot{\theta}$: longitudinal acceleration.
$a_y = \ddot{y} + \dot{x}\dot{\theta}$: lateral acceleration.
$C_{\alpha f}, C_{\alpha r}$: are respectively the cornering stiffness of the front and the rear wheels.
τ: driving/braking wheels torque.
δ : steering wheel angle.
$F_{aero} = \frac{1}{2}\rho c s \dot{x}^2$: aerodynamic force, where ρ, s and c : are the air density, the vehicle frontal surface and the aerodynamic constant.
R_t :Radius of the tire and E: Vehicle's track.
We define m_e, L_3 and I_3 as follows :
$m_e = m + 4\frac{I_w}{R_t^2}$, $L_3 = 2m_w(L_r - L_f)$ and $I_3 = I_z + m_w E^2$.

The generalized coordinates $q \in R^3$ are defined as : $q_i = [x_i, y_i, \theta_i]^T$. The dynamic model of a vehicle is presented as follows:

$$M_i(q_i).\ddot{q}_i + H_i(\dot{q}_i, q_i) = U_i \tag{1}$$

Where the inertia Matrix $M_i(q_i)$ is:

$$M_i = \begin{pmatrix} m_{e_i} & 0 & 0 \\ 0 & m_i & -L_{3i} \\ 0 & -L_{3i} & I_{3i} \end{pmatrix}$$

And the vector $H_i(\dot{q}_i, q_i)$ is equal to

$$H_i(\dot{q}_i, q_i) = \begin{pmatrix} -m_i\dot{q}_{2i}\dot{q}_{3i} + L_{3i}\dot{q}_{3i}^2 + \delta_i(2C_{\alpha fi}\delta_i - 2C_{\alpha fi}\frac{\dot{q}_{1i}(\dot{q}_{2i}+L_{fi}\dot{q}_{3i})}{\dot{q}_{1i}^2-(\dot{q}_{3i}E_i/2)^2}) + F_{aero_i} \\ m_i\dot{q}_{1i}\dot{q}_{3i} + 2C_{\alpha fi}\frac{\dot{q}_{1i}(\dot{q}_{2i}+L_{fi}\dot{q}_{3i})}{\dot{q}_{1i}^2-(\dot{q}_{3i}E_i/2)^2} + 2C_{\alpha ri}\frac{\dot{q}_{1i}(\dot{q}_{2i}-L_{ri}\dot{q}_{3i})}{\dot{q}_{1i}^2-(\dot{q}_{3i}E_i/2)^2} \\ 2L_{fi}C_{\alpha fi}\frac{\dot{q}_{1i}(\dot{q}_{2i}+L_{fi}\dot{q}_{3i})}{\dot{q}_{1i}^2-(\dot{q}_{3i}E_i/2)^2}) - 2L_{ri}C_{\alpha ri}\frac{\dot{q}_{1i}(\dot{q}_{2i}-L_{ri}\dot{q}_{3i})}{\dot{q}_{1i}^2-(\dot{q}_{3i}E_i/2)^2} - L_{3i}\dot{q}_{1i}\dot{q}_{3i} \end{pmatrix}$$

And the input vector $U_i = (u_{1i}, u_{2i}, u_{3i})^T$:

$$U_i = \begin{pmatrix} \frac{\tau_i}{R_{t_i}} \\ (2C_{\alpha fi} - 2\frac{I_{wi}}{R_{t_i}^2}\ddot{q}_{1i})\delta_i \\ L_{fi}u_{2i} - (\frac{E_i}{2}C_{\alpha fi}\frac{E_i\dot{q}_{3i}(\dot{q}_{2i}+L_{fi}\dot{q}_{3i})}{\dot{q}_{1i}^2-(\dot{q}_{3i}E_i/2)^2})\delta_i \end{pmatrix}$$

The inputs of the system are the control of the torque and the steering wheel angle.

2.2. Kinematic Equations

The transformation matrix of the velocity, from the absolute vehicle frame (G, x, y) to the velocity in the reference frame R(0,X,Y) is defined by:

$$\begin{bmatrix} \dot{X}_i \\ \dot{Y}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 \\ \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\theta}_i \end{bmatrix} \quad (2)$$

Such as we get the kinematics of the i^{th} vehicle:

$$\begin{cases} \dot{X}_i = \dot{x}_i \cos\theta_i - \dot{y}_i \sin\theta_i \\ \dot{Y}_i = \dot{x}_i \sin\theta_i + \dot{y}_i \cos\theta_i \end{cases} \quad (3)$$

2.3. Convoy Motion

The movement of the fleet in a path of reference is presented in the Fig. 2, the convoy moves in this trajectory with a distance that separates every two vehicles. The curvilinear inter-distance error is calculated as a function of the travel distance for each two neighboring vehicles in the curvature of the reference path. Let M the center of gravity of the vehicle (i), the curvilinear error between the vehicle (i) and the vehicle (i-1) is defined as follows:

$$e_{s_i} = S_{i-1} - S_i - l_{d_i} \quad (4)$$

S_i : represents the curvilinear abscissa of the vehicle (i) at the center of gravity, is calculated as follows:

$$\dot{S}_i = \sqrt{\dot{x}_i^2 + \dot{y}_i^2} \quad (5)$$

By replacing equation (5) in equation (4), the curvilinear error will be defined as follows:

$$e_{s_i} = \int_0^t (\dot{x}_{i-1}^2 + \dot{y}_{i-1}^2)^{\frac{1}{2}} dt - \int_0^t (\dot{x}_i^2 + \dot{y}_i^2)^{\frac{1}{2}} dt - l_{d_i} \quad (6)$$

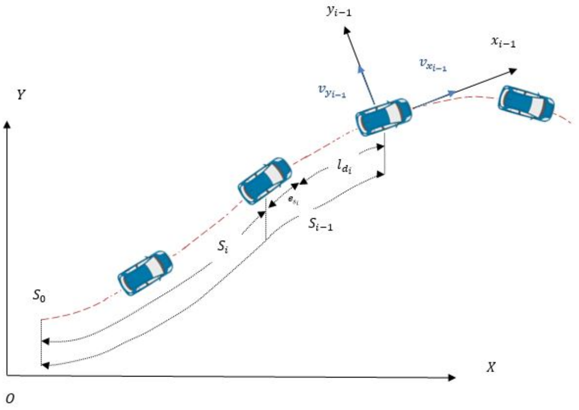


Figure 2: Geometric description of the convoy motion

3. CONTROL

3.1. Vehicle State Space Model

We have as a state vector, the position and speed of each vehicle:

$$z_i = \begin{pmatrix} z_{1i} \\ z_{2i} \end{pmatrix} \quad (7)$$

$$\dot{z}_{2i} = M^{-1}(z_{1i})(-H(z_{1i}, z_{2i}) + U_i) \quad (8)$$

with positions: $z_{1i} = [x_i, y_i, \theta_i]^T$
and velocities: $z_{2i} = [\dot{x}_i, \dot{y}_i, \dot{\theta}_i]^T$
The dynamic model of a vehicle i of the convoy is represented in canonical forms :

$$\begin{cases} \dot{z}_{1i} = z_{2i} \\ \dot{z}_{2i} = f(z_{1i}, z_{2i}) + g(z_{1i}) U_i \end{cases} \quad (9)$$

For our model we have

$$f(z_{1i}, z_{2i}) = -M^{-1}(z_{1i}) H(z_{1i}, z_{2i})$$

and $\quad g(z_{1i}) = M^{-1}(z_{1i})$

3.2. The objectives

The aims of the control are to :
-Control the vehicles to follow the trajectories of the leader by ensuring a safe distance between the vehicles to avoid collisions,
-Use the available information to calculate the law of the control and ensures local stability for each vehicle and global one for the fleet,
-Ensure robustness of control over errors in model parameter estimates with the presence of a non-linear model

3.3. Longitudinal and lateral control

Tracking accuracy can be improved by using non-linear predictive control based on knowledge of the reference trajectory Hedjar et al. [2005] Merabet and Gu [2008]. This control approach is based on the optimization of the cost function with the objective of controlling the fleet to follow the trajectory of the leader with a safety distance between the vehicles to avoid the collision Song et al. [2017].

$$J_i = \frac{1}{2} \int_0^h e_i(t+T)^T Q_i e_i(t+T) dT + \frac{1}{2} U_i^T R_i U_i \quad (10)$$

With: h represents the horizon of the prediction, T is the time of the prediction . $e(t+T)$: the tracking errors at the next step. For the model defined in the equation (9), we have that $e_i = (e_{1i}, e_{2i})$ with e_i represents the position errors, e_2 speed errors and $Q = \begin{pmatrix} Q_1 & 0 \\ 0 & T^2 Q_2 \end{pmatrix}$ With Q_1, Q_2, R are weighting matrices.
The aim of the longitudinal control of the fleet is to impose a longitudinal speed on the fleet and to ensure a safety distance between each two neighboring vehicles Fig. 3.
To simplify the writing we define the error of the fleet as defined in the dynamic model as $e_i = (e_{1i}, e_{2i}, e_{3i})$ with e_1 the longitudinal error of the position, e_2 the error of the lateral position and e_3 the error of the position of the yaw.
We define the curvilinear spacing error between the vehicles of the convoy:

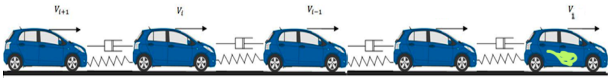


Figure 3: The longitudinal movement of the convoy

$$e_{1i}(t+T) = S_i(t+T) - S_{i-1}(t+T) + l_d(t+T)$$

With l_d : the safety distance. $S_i(t+T)$: represents the curvilinear abscissa of the vehicle (i) at the next step.
The error of the longitudinal velocity is defined as follows:

$$\dot{e}_{1i}(t+T) = \dot{x}_i(t+T) - \dot{x}_{i-1}(t+T)$$

The lateral movement of the fleet is controlled by the steering angle of vehicles. The error of the lateral acceleration is defined as follows:

$$\ddot{e}_{2i}(t+T) = ay_i(t+T) - ay_{di}(t+T) \qquad (11)$$

The reference lateral acceleration is calculated as follows: $ay_{di} = \dot{x}_i^2/r_i$. Such as r_i represents the radius of the leader's trajectory. We have that $a_{yi} = \ddot{y}_i + \dot{x}_i\dot{\theta}_i$. replacing the two previous expressions in the equation (11), We can write the lateral error in the following form :

$$\ddot{e}_{2i}(t+T) = \ddot{y}_i(t+T) - \ddot{y}_{di}(t+T)$$

With : $\ddot{y}_{di} = \dot{x}_i^2/r_i - \dot{x}_i\dot{\theta}_i$.
The prediction of the tracking error (longitudinal and lateral) can be made using the Taylor approximation and based on the model defined in (9), such as e_i represents the position error:

$$e_i(t+T) = e_i(t) + T\dot{e}_i + \frac{T^2}{2!}(f(z_i) - \ddot{z}_{id}) + \frac{T^2}{2!}g(z_i)U_i$$

$$\dot{e}_i(t+T) = \dot{e}_i + T(f(z_i) - \ddot{z}_{id}) + Tg(z_i)U_i$$

The minimization of the cost function is obtained such that: $\partial J_i/\partial U_i = 0$

$$\begin{aligned} U_i = & -g(z_i)^{-1}(\frac{h^5}{20}(Q_{1i}+4Q_{2i}) \\ & + g(z_{1i})^{-T}Rg(z_{1i})^{-1})^{-1}(\frac{h^3}{6}Q_{1i}e_i + \dots \\ & + \frac{h^4}{8}(Q_{1i}+2Q_{2i})\dot{e}_i + \frac{h^5}{20}(Q_{1i}+4Q_{2i})(f(z_{1i},z_{2i}) - \ddot{z}_{i-1}) \end{aligned} \qquad (12)$$

U_i controls the longitudinal movement of the fleet by the torque (u_{1i}) of each vehicle and the lateral movement by the steering angle (u_{2i}). In our case, the longitudinal and lateral control are coupled by the longitudinal velocity. The steering angle is used to calculate the third control (u_{3i}) (yaw movement) to calculate the yaw rate and present the movement of the fleet in the reference frame (0,X,Y) by the transformation matrix.

3.3.1. Convergence Analysis

The stability study for each vehicle in the convoy is based on the vehicle error and the lateral error with respect to the leader's trajectory. We define the parameters: $K_{1i} = \frac{h^3}{6}Q_{1i}, K_{2i} = \frac{h^4}{8}(Q_{1i}+2Q_{2i})$ and $K_{3i} = \frac{h^5}{20}(Q_{1i}+4Q_{2i})$ For the stability study according to the errors (longitudinal and lateral), we neglect the weighting on the control. Let the candidate Lyapunov function :

$$V_i = \frac{1}{2}\dot{e}_i^T\dot{e}_i + \frac{1}{2}e_i^T\frac{K_1}{K_3}e_i \qquad (13)$$

Deriving this function we find:

$$\dot{V}_i = \dot{e}_i^T\ddot{e}_i + \dot{e}_i^T\frac{K_1}{K_3}e_i \qquad (14)$$

Replacing $\ddot{e}$ (the acceleration error) with its expression ($\ddot{e}_i = \ddot{z}_i - \ddot{z}_{i-1}$) :

$$\dot{V}_i = \dot{e}_i^T(f(z_{1i},z_{2i}) + g(z_{1i})U_i - \ddot{z}_{i-1}) + \dot{e}_i^T\frac{K_1}{K_3}e_i \quad (15)$$

$$\begin{aligned} & = \dot{e}_i^T(f(z_{1i},z_{2i}) - \ddot{z}_{i-1} - \frac{K_1}{k_3}e + \dots \\ & - \frac{K_2}{K_3}\dot{e}_i - f(z_{1i},z_{2i}) + \ddot{z}_{i-1}) + \dot{e}_i^T\frac{K_1}{K_3}e_i \end{aligned} \quad (16)$$

It is clear that the stability condition is verified when the gains of the weighting matrices are positive such that : $\dot{V}_i = -\dot{e}_i^T\frac{K_2}{K_3}\dot{e}_i < 0$. The choice of K_1, K_2 and K_3 depends on the weighting matrices and the horizon of the prediction that is around ms. By increasing the gains of the matrices Q_1 and Q_2, the stability is still checked and ensured.

4. SIMULATIONS

To validate this result we used the parameters of a vehicle of the Scanner Studio. 10 vehicles are simulated in Matlab Simulink using both dynamic and kinematic models. The simulation achieved to validate the control law in both directions of longitudinal and lateral motions and to check the stability and accuracy of trajectory tracking. The leader has been controlled using a chosen reference speed and a desired trajectory. The other vehicles using the predecessor's information to calculate their control and follow the path of the leader and ensure distances between each neighboring pair.
The longitudinal velocity is limited to $v_x < 50km/h$, and the imposed lateral acceleration has been bounded by two values ; $a_{ymin} < a_y < a_{ymax}$ and as a function of the longitudinal velocity and the radius of the reference trajectory of the leader and the convoy. The inter-vehicle distance is limited between $l_{dmin} < l_d < l_{dmax}$. The displacement of the fleet in the fixed reference is presented using the following kinematic model:

$$X_i = \int_0^t (\dot{x}_i\cos\theta_i - \dot{y}_i\sin\theta_i)dt \qquad (17)$$

Parameter	Value	Parameter	Value
m	1500 kg	m_w	$23.2kg$
I_z	$1652.7kg.m^2$	I_w	$2kg.m^2$
$C\alpha_f$	$67689N/rad$	$C\alpha r$	$69253N/rad$
Lr	$1.441m$	Lf	$1.099m$
s	$2m^2$	E	$1.5m$
c	0.3	ρ	1.3

Table 1: Vehicle parameters values SCANeR-Studio

$$Y_i = \int_0^t (\dot{x}_i \sin\theta_i + \dot{y}_i \cos\theta_i) dt \tag{18}$$

The two previous equations are used to calculate the positions of the fleet in the reference frame R(O,X,Y).
The Fig. 4 shows the movement of the fleet in the leader's trajectory. The convoy follows the path of the leader, the lateral error is almost negligible; that is to say, no angular deviation between the fleet and the desired trajectory. We can see that even with this control approach that uses the information from its predecessor, the fleet is still on the same path and the accumulation of tracking error is almost negligible too. The path tracking accuracy of the convoy Fig. 4 shows the robustness of the non-linear predictive control for tracking the trajectory of a fleet of 10 vehicles that takes into account the non-linear dynamics of each vehicle in the convoy. This control approach makes it possible to control the movements of the fleet, based on the available data, longitudinal and lateral movements.

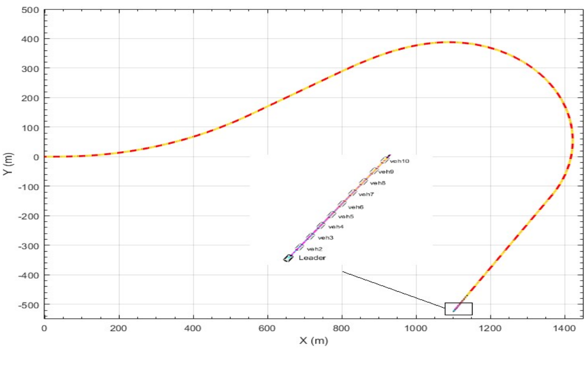


Figure 4: Trajectory of the convoy

The safety distance between the fleet is shown in the Fig 5. We can see a deviation of this distance between 2 and 4.5 m then it stabilizes for a value of 3.5 m. This safety distance was chosen for speed around 43 km/h. Generally, the convoy moves at low speed. The safety distance is almost the same for convoy vehicles. For a convoy that moves with a high speed, the distance must be higher because the risk of collision increases with a high speed and a small distance

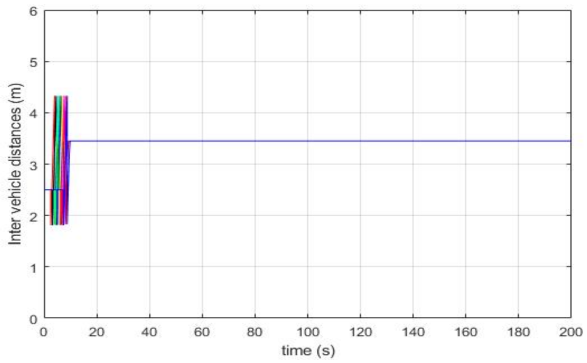


Figure 5: Inter vehicle distance

The Fig. 6 shows the different steering angles for the fleet vehicles. These angles are calculated using the second term of the global control (u_{2i}) such as:

$$\delta_i = u_{2_i}/(2C_{\alpha fi} - 2\frac{I_{wi}}{R_{ti}^2}\ddot{x}_i) \tag{19}$$

This control approach makes it possible to control the longitudinal and lateral movements of the fleet. The lateral movement of the fleet is based on the trajectory of the leader as shown in the Fig. 4 and the longitudinal velocity. That is, both controls are coupled by lateral acceleration. We can clearly see a lateral movement or a lateral deviation by carrying the x-axis of the longitudinal movement from the t = 5 s. The steering angle is almost constant between the interval t ∈ [10, 55 s] with a value of 0.03 rad. This value is always dependent on the speed of the fleet and the desired trajectory.

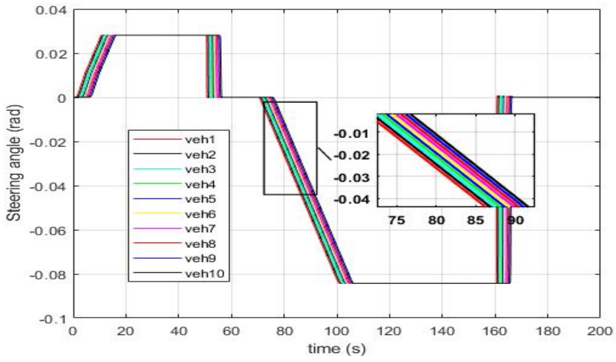


Figure 6: Steering angle

The Longitudinal speed of the fleet is presented in the Fig. 7. This speed has been imposed for the leader. By the law of the control and with the predictive control it is clear that the vehicle speeds of the convoy converge quickly to the speed of the leader. The speed of the convoy compared to the speed of the leader which is propagated in the convoy is almost negligible, which shows the best precision and performance given by the predictive control for a convoy of 10 vehicles.

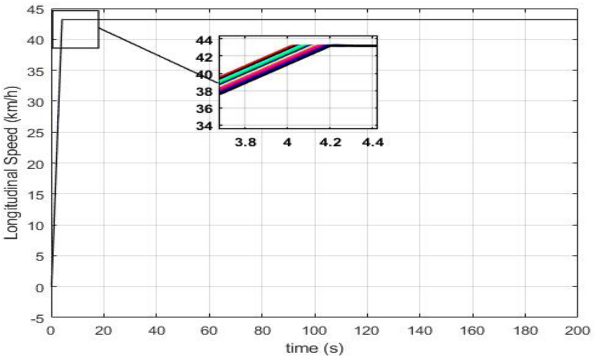


Figure 7: velocity v_x of the convoy

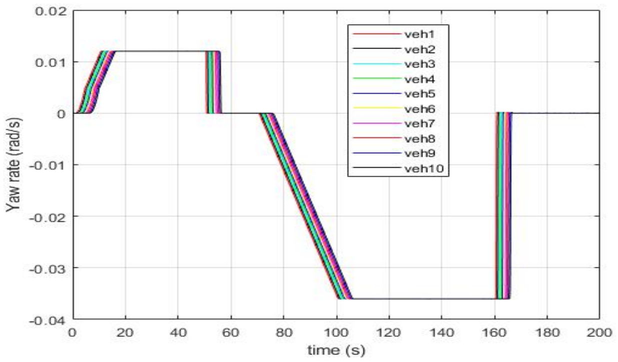


Figure 8: Yaw rate of the convoy

The lateral acceleration of the fleet is presented in Fig. 9. In our case, we took into account the speed of the yaw as $a_{yi} = \ddot{y}_i + \dot{\theta}_i \dot{x}_i$. This acceleration is proportional to the longitudinal velocity and the radius of the leader's trajectory. We see clearly at t = 5s a presence of the lateral movement to wait for a value 0.15 m/s^2. This acceleration is positive for t ∈ [5, 55 s] and allows vehicles to be oriented for a positive lateral deviation along the y-axis. For t ∈ [55, 75 s] the lateral acceleration is zero, that proves , that the fleet remains in the same direction (longitudinal direction), then between [75, 170 s], we can see a decceleration. Fig. 10 and Fig. 8 represents the lateral velocities of the fleet and the yaw rate which are proportional to the lateral acceleration.

To test the robustness of the control on the parameters of the model; we assumed that the parameters are not well estimated, that is, 20% errors of $f \Rightarrow \Delta f = f - \hat{f} = 20\% f$ and 20 of g $\Rightarrow \Delta g = g - \hat{g} = 20\% g$. The results Fig. 11 and 12 show that the fleet is still following the leader's trajectory and the safety distance remains the same. The lateral deviation from the reference trajectory is still negligible, which proves the robustness of the control compared to the estimation errors on the model parameters.

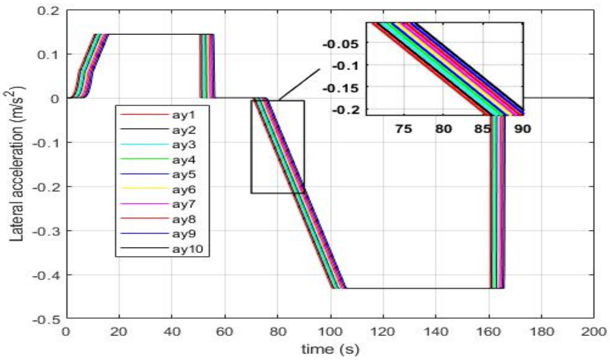


Figure 9: Lateral acceleration a_y of the convoy

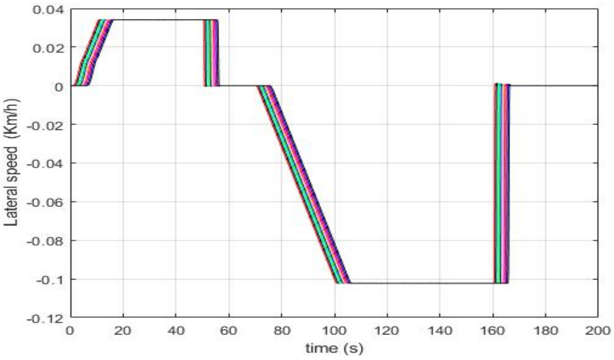


Figure 10: velocity v_y of the convoy

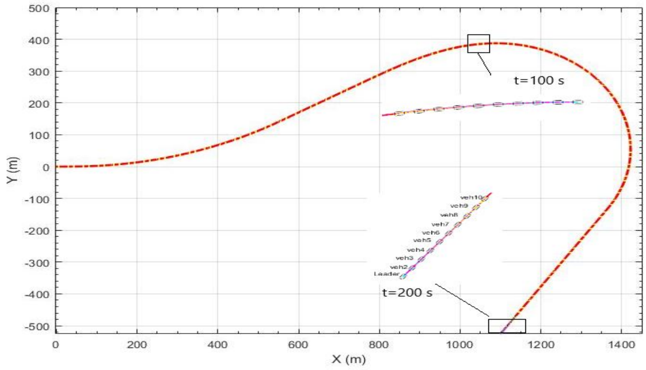


Figure 11: Trajectory of the convoy +20% of estimation errors

5. CONCLUSIONS

In this paper, we proposed a coupled longitudinal and lateral control for a convoy of autonomous vehicles. This approach uses nonlinear predictive control for tracking trajectory. The proposed approach allows to control the fleet by the available information and to follow the reference trajectory of the leader. Dynamic and kinematic modeling was presented to control and represent the movement of the fleet in the reference frame. This

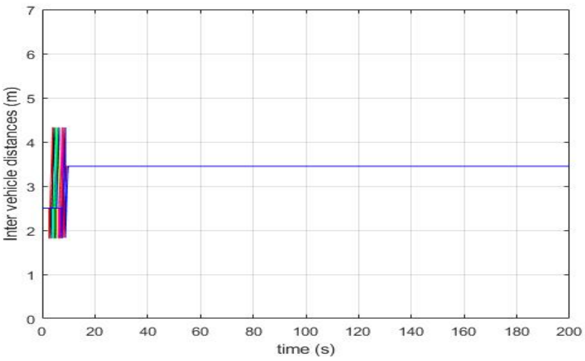


Figure 12: Inter vehicle distance $+20\%$ of estimation errors

nonlinear control approach has shown a precision performance with respect to the trajectory tracking for the lateral movement of the fleet and robustness when the parameters are not well estimated. The control law makes it possible to ensure a safe distance between the vehicles to avoid collisions by the longitudinal control, such that the fleet moves with the same speed of the leader. Accumulation of fleet tracking error is negligible when using this control approach.

REFERENCES

A. Ali. *Modélisation et commande d'un convoi de véhicules urbains*. PhD thesis, Université Nantes Angers Le Mans, 2015.

Alan Ali, Gaetan Garcia, and Philippe Martinet. Urban platooning using a flatbed tow truck model. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 374–379. IEEE, 2015.

Pierre Avanzini. *Modélisation et commande d'un convoi de véhicules urbains par vision*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II, 2010.

Pierre Avanzini, Benoit Thuilot, and Philippe Martinet. Accurate platoon control of urban vehicles, based solely on monocular vision. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 6077–6082. IEEE, 2010.

Alia Chebly. *Trajectory planning and tracking for autonomous vehicles navigation*. PhD thesis, Université de Technologie de Compiègne, 2017.

RM DeSantis. Path-tracking for car-like robots with single and double steering. *IEEE Transactions on vehicular technology*, 44(2):366–377, 1995.

Ramdane Hedjar, Redouane Toumi, Patrick Boucher, and Didier Dumur. Finite horizon nonlinear predictive control by the taylor approximation: application to robot tracking trajectory. *International Journal of Applied Mathematics and Computer Science*, 15:527–540, 2005.

Belgacem Jaballah. *Observateurs robustes pour le diagnostic et la dynamique des véhicules*. PhD thesis, Université Paul Cézanne-Aix-Marseille III, 2011.

A Merabet and J Gu. Robust nonlinear predictive control based on state estimation for robot manipulator. *International Journal of Applied Mathematics and Mechanics*, 5(1):48–64, 2008.

Nacer K. M'Sirdi. Vehicle platooning: an overview on modelling and control approaches. In *International Conference on Applied Smart Systems (ICASS'18)*. Medea University, 2018.

L Nouveliere, J Sainte Marie, and S. Mammar an N K M'Sirdi. Controle longitudinal de véhicules parcommande sous optimale. In *CIFA 2002*, pages 906–911. Nantes Juillet, 2002.

Abdelhamid Rabhi. *Estimation de la dynamique du véhicule en interaction avec son environnement*. PhD thesis, Versailles-St Quentin en Yvelines, 2005.

Linhuan Song, Hongyan Guo, Fei Wang, Jun Liu, and Hong Chen. Model predictive control oriented shared steering control for intelligent vehicles. In *Control And Decision Conference (CCDC), 2017 29th Chinese*, pages 7568–7573. IEEE, 2017.

D. Swaroop. *String Stability of Interconnected Systems : An application to platooning in AHS*. PhD thesis, University of California at Berkeley, 1994.

Ji Xiang and Thomas Bräunl. String formations of multiple vehicles via pursuit strategy. *IET control theory & applications*, 4(6):1027–1038, 2010.

Jano Yazbeck. *Accrochage immatériel sûr et précis de véhicules automatiques*. PhD thesis, Université de Lorraine, 2014.

DEFENDER-ATTACKER-TARGET GAME:
FIRST-ORDER DEFENDER AND ATTACKER DYNAMICS

Vladimir Turetsky[(a)], **Valery Y. Glizer**[(b)]

[(a), (b)] Ort Braude College of Engineering

[(a)]turetsky1@braude.ac.il, [(b)]valery48@braude.ac.il

ABSTRACT

Based on the solution of a linear-quadratic differential game with a terminal attacker's constraint, obtained in the previous paper, the practically important case of first-order players' dynamics is treated. The game space decomposition is constructed. The fulfillment of the saddle point inequalities is demonstrated. The feedback realization of the optimal strategies is presented.

Keywords: pursuit-evasion differential game, zero-sum linear-quadratic game, terminal constraint, first-order dynamics

1. INTRODUCTION

A defender-attacker-target problem is a widely discussed topic in the control and guidance literature (see e.g., (Rubinsky and Gutman 2014; Garcia, Casbeer, and Pachter 2017) and others). In the previous paper of the authors (Turetsky and Glizer 2019), one can find a detailed literature review on different approaches for the modeling and solution of this problem.

In (Turetsky and Glizer 2019), a linear-quadratic differential game with an attacker's terminal constraint was considered. It was assumed that the player's controllers are described by linear differential equations of an arbitrary order. In this game, the objective of the defender (pursuer) is to capture the attacker (evader), i.e. to nullify the miss distance (the closest separation between the vehicles), while the evader tries to avoid the capture. However, in a practical situation, avoiding the capture is not the main aim of the evader. Its actual aim is to hit a prescribed static object (target), while avoiding the capture is an auxiliary aim (see, e.g., (Lipman and Shinar 1995) and references therein). Since the evader tries not only to escape the pursuer, but also to hit the target, it should be able to reach the target after the interception moment. This is described by a terminal state inequality constraint.

The general solution of the corresponding linear-quadratic differential game with the terminal evader's constraint was obtained in the previous paper of the authors (Turetsky and Glizer 2019). It was shown that subject to a condition on the evader's penalty coefficient in the cost functional, the game space is decomposed into three non-intersecting regions of different saddle point solutions.

In the literature, different types of missiles' dynamics can be found. The zero-order evader's dynamics ("ideal" evader) is traditionally interpreted as a worst case for the pursuer. The special case, where the controller dynamics of both the pursuer and the evader is zero-order, was considered in (Rubinsky and Gutman 2014; Glizer and Turetsky 2015). In (Lipman and Shinar 1995), the evader is ideal, whereas the pursuer has the first-order dynamics. However, real life controllers cannot transfer the control command into the missile acceleration instantaneously. Therefore, the zero-order dynamics model has rather theoretic implementation. In this paper, the special case where both players have first-order dynamics is elaborated. The first-order dynamics of the players models an intrinsic controller property: a time lag between the control command and the lateral acceleration. Thus, this case represents a realistic model of missiles engagement as emphasized by Shinar (1981), which makes it very important from the practical point of view. The first-order approximation of the pursuer's and the evader's dynamics was exploited in numerous papers on vehicles guidance and control (see, e.g., (Shinar 1981; Shinar, Glizer, and Turetsky 2013) and references therein).

2. PREVIOUS RESULTS

In this section, the results of Turetsky and Glizer (2019) on the solution of the game with arbitrary order of the players' controllers are briefly outlined.

2.1. Original Pursuit-Evasion Game

The engagement between the defender (pursuer) and the attacker (evader) is considered. In Fig. 1, the schematic engagement geometry is depicted. The X - axis is the initial line of sight. The Y -axis is normal to the X - axis. The origin of the coordinate system is collocated with the target position, which is also the initial position of the pursuer. The points (x_p, y_p) and (x_e, y_e) are current coordinates of the pursuer and the evader, respectively; V_p, V_e are their velocities; a_p, a_e are their lateral accelerations; φ_p, φ_e are the respective angles between the velocity vectors and the X -axis.

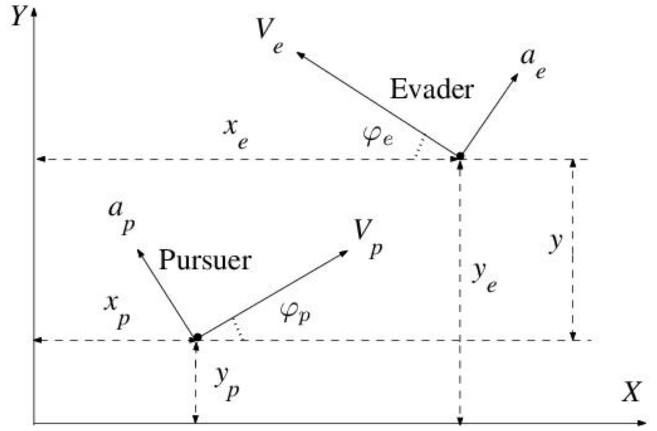


Figure 1: Interception Geometry

The controller dynamics of the pursuer and the evader are described by the equations

$$\dot{\bar{x}}_i = \bar{A}_i \bar{x}_i + \bar{b}_i u_i,\ \bar{x}_i(0) = [0]_{n_i \times 1},\ i = p,e, \tag{1}$$

$$a_i = \bar{c}_i^T \bar{x}_i + \bar{d}_i u_i,\ i = p,e, \tag{2}$$

where $\bar{x}_i$ is the state vector consisting of n_i internal variables, u_i is the scalar control, $i = p,e$; a_p and a_e are the lateral accelerations of the pursuer and the evader, respectively, $[0]_{k\times m}$ denotes a zero $(k \times m)$-matrix. In the equations (1) – (2), $\bar{A}_i$ is a given constant matrix, $\bar{b}_i$ and $\bar{c}_i$ are given constant vectors, $\bar{d}_i$ is a given scalar.

The system dynamics for $t \in [0, t_f]$ is described by the linear differential equations of motion of the pursuer and the evader:

$$\dot{X}_i = A_i X_i + B_i u_i,\ i = p,e \tag{3}$$

where the state vector is $X_i = [y_i, \dot{y}_i, \bar{x}_i^T]^T$,

$$A_i = \begin{bmatrix} 0 & 1 & [0]_{1\times n_i} \\ 0 & 0 & \bar{c}_i^T \\ [0]_{n_i \times 1} & [0]_{n_i \times 1} & \bar{A}_i \end{bmatrix}, \tag{4}$$

$$B_i = \begin{bmatrix} 0 \\ \bar{d}_i \\ \bar{b}_i \end{bmatrix}, i = p,e. \tag{5}$$

The initial condition is

$$X_i(0) = [0, V_i \varphi_i(0), [0]_{1\times n_i}]^T,\ i = p,e. \tag{6}$$

The objective of the pursuer is to minimize the cost functional

$$J = (y_e(t_f) - y_p(t_f))^2 + \alpha \int_0^{t_f} u_p^2(t)dt - \beta \int_0^{t_f} u_e^2(t)dt, \tag{7}$$

where $\alpha, \beta > 0$ are penalties for the players' controls. The evader's first objective is to maximize (7). The second objective is to be capable to reach the target at $t = t_f + t_c$, i.e. to satisfy the constraint

$$| D_e \Phi_e(t_f + t_c, t_f) X_e(t_f) | \le \mu_e a_e^{\max}, \tag{8}$$

where $t_c = \nu t_f$, $\nu = V_p / V_e$,

$$D_e = \left[1, [0]_{1\times(n_e+1)}\right], \tag{9}$$

$$\mu_e = \int_{t_f}^{t_f+t_c} \left| D_e \Phi_e(t_f + t_c, t) B_e \right| dt, \tag{10}$$

$\Phi_e(t,\tau)$ is the transition matrix of the homogeneous system, corresponding to (3) for $i = e$. For $t \in [t_f, t_f + t_c]$, it is assumed that

$$| u_e(t) | \le a_e^{\max}. \tag{11}$$

The pursuit-evasion differential game for the system (3) with the cost functional (7) and the evader's terminal constraint (8) is called the Original Pursuit-Evasion Game (OPEG).

2.2. Reduced Game

The relative motion between the evader and the pursuer in the direction normal to the initial line-of-sight (the Y axis direction) is described by the system

$$\dot{X}_{ep} = A_{ep} X_{ep} + B_{ep} u_p + C_{ep} u_e, \tag{12}$$

where $X_{ep} = [y_e - y_p, \dot{y}_e - \dot{y}_p, x_p^T, x_e^T]^T$,

$$A_{ep} = \begin{bmatrix} 0 & 1 & [0]_{1\times n_p} & [0]_{1\times n_e} \\ 0 & 0 & -c_p^T & c_e^T \\ [0]_{n_p\times 1} & [0]_{n_p\times 1} & A_p & [0]_{n_p\times n_e} \\ [0]_{n_e\times 1} & [0]_{n_e\times 1} & [0]_{n_e\times n_p} & A_e \end{bmatrix},$$

$$B_{ep} = \begin{bmatrix} 0 \\ -\bar{d}_p \\ \bar{b}_p \\ [0]_{n_e\times 1} \end{bmatrix}, C_{ep} = \begin{bmatrix} 0 \\ \bar{d}_e \\ [0]_{n_p\times 1} \\ \bar{b}_e \end{bmatrix}.$$

Let $D_{ep} = \left[1, [0]_{1\times(n_p+n_e+1)}\right]$, $\Phi_{ep}(t_f, t)$ be the transition matrix of the homogeneous system, corresponding to (12). New scalar state variables

$$z(t) = D_{ep} \Phi_{ep}(t_f, t) X_{ep}(t), \tag{13}$$

$$w(t) = D_e\Phi_e(t_f + t_c, t)X_e(t), \tag{14}$$

satisfy the differential equations

$$\dot{z} = h_p(t)u_p + h_e(t)u_e,\ z(0) = z_0, \tag{15}$$

$$\dot{w} = g_e(t)u_e,\ w(0) = w_0, \tag{16}$$

where

$$h_p(t) = D_{ep}\Phi_{ep}(t_f, t)B_{ep},$$
$$h_e(t) = D_{ep}\Phi_{ep}(t_f, t)C_{ep}, \tag{17}$$

$$g_e(t) = D_e\Phi_e(t_f + t_c, t)B_e, \tag{18}$$

$$z_0 = t_f(V_e\varphi_e^0 - V_p\varphi_p^0),\ w_0 = (t_f + t_c)V_e\varphi_e^0. \tag{19}$$

Note that $z(t_f) = y_e(t_f) - y_p(t_f)$, and the cost functional (7) can be rewritten as

$$J = |z(t_f)|^2 + \alpha\int_0^{t_f} u_p^2(t)dt - \beta\int_0^{t_f} u_e^2(t)dt. \tag{20}$$

Due to (14), the constraint (8) becomes

$$|w(t_f)| \le \mu_e a_e^{\max}. \tag{21}$$

Thus, the OPEG is reduced to the pursuit-evasion differential game for the system (15) – (16) with the cost functional (20) and the terminal evader's constrain (21). This game is called the Reduced Pursuit-Evasion Game (RPEG).

2.3. Saddle Points in Reduced Game

In this section, we obtain the pairs of strategies $(u_p^*(\cdot), u_e^*(\cdot))$, constituting the saddle point in the Reduced Game, i.e., satisfying for all admissible strategies $u_p(\cdot)$, $u_e(\cdot)$ the saddle point inequality

$$J(u_p^*(\cdot), u_e(\cdot)) \le J(u_p^*(\cdot), u_e^*(\cdot)) \le J(u_p(\cdot), u_e^*(\cdot)). \tag{22}$$

Let us define the values

$$s = 1 + \frac{1}{\alpha}\int_0^{t_f} h_p^2(t)dt - \frac{1}{\beta}\int_0^{t_f} h_e^2(t)dt > 0, \tag{23}$$

$$a = \frac{1}{\beta s}\int_0^{t_f} h_e(t)g_e(t)dt. \tag{24}$$

$$v_p = \frac{1}{\alpha}\int_0^{t_f} h_p^2(t)dt, \tag{25}$$

$$G_2 = \frac{1}{\beta}\int_0^{t_f} h_e(t)g_e(t)dt,\ G_3 = \frac{1}{\beta}\int_0^{t_f} g_e^2(t)dt, \tag{26}$$

$$d = \frac{v_p G_2^2}{G_1((s - v_p)G_3 + G_2^2)},$$

(27)

the matrix

$$G = \begin{bmatrix} s(0) & G_2 \\ -G_2 & G_3 \end{bmatrix}, \tag{28}$$

the vectors

$$b^+ = \begin{bmatrix} z_0 \\ w_0 - \mu_e a_e^{\max} \end{bmatrix}, b^- = \begin{bmatrix} z_0 \\ w_0 + \mu_e a_e^{\max}. \end{bmatrix}, \tag{29}$$

$$\omega_f^+ = (z_f^+, v_f^+)^T = G^{-1}b^+, \tag{30}$$

$$\omega_f^- = (z_f^-, v_f^-)^T = G^{-1}b^-, \tag{31}$$

and the sets

$$\Omega = \{(z_0, w_0) : |w_0 + az_0| < \mu_e a_e^{\max}\}, \tag{32}$$

$$\Omega^+ = \{(z_0, w_0) \notin \Omega : w_0 + az_0 \ge d\mu_e a_e^{\max}\}, \tag{33}$$

$$\Omega^- = \{(z_0, w_0) \notin \Omega : w_0 + az_0 \le -d\mu_e a_e^{\max}\}. \tag{34}$$

In what follows, we assume that the condition

$$\beta > \int_0^{t_f} h_e^2(t)dt \tag{35}$$

holds.

Remark 1. *Subject to the condition (35),*

$$\Omega^+ = \{(z_0, w_0) : w_0 + az_0 \ge \mu_e a_e^{\max}\}, \tag{36}$$

$$\Omega^- = \{(z_0, w_0) : w_0 + az_0 \le -\mu_e a_e^{\max}\}. \tag{37}$$

In this case, the planar sets Ω, Ω^+ and Ω^- do not intersect each other, and $\Omega \cup \Omega^+ \cup \Omega^-$ coincides with the entire (z_0, w_0)-plane.

The saddle point solutions of Reduced Game are defined separately for the cases $(z_0, w_0) \in \Omega$, $(z_0, w_0) \in \Omega^+$ and $(z_0, w_0) \in \Omega^-$.

Theorem 1. *If the condition (35) holds and* $(z_0, w_0) \in \Omega$, *the pair*

$$u_p^0(t) = -\frac{h_p(t)z_0}{\alpha s},\ u_e^0(t) = \frac{h_e(t)z_0}{\beta s}, \tag{38}$$

is an open-loop saddle point in the Reduced Game.

Remark 2. *If* $(z_0, w_0) \in \Omega$, *then, the solution* $w(t)$ *generated by* $u_e^*(\cdot)$ *in (38), satisfies the inequality (21) strictly, i.e.,* $|w(t_f)| < \mu_e a_e^{\max}$.

Theorem 2. *Let (35) hold and* $(z_0, w_0) \in \Omega^+$. *Then, the pair*

$$\begin{aligned} u_p^+(t) &= -\frac{1}{\alpha} h_p(t) z_f^+, \\ u_e^+(t) &= \frac{1}{\beta}\left[h_e(t) z_f^+ - g_e(t) v_f^+\right] \end{aligned} \tag{39}$$

is an open-loop saddle point in the Reduced Game.

Remark 3. *If* $(z_0, w_0) \in \Omega^+$, *then the optimal trajectory* $(z(t), w(t))$ *generated by the pair* $(u_p^+(\cdot), u_e^+(\cdot))$ *satisfies the terminal conditions*

$$z(t_f) = z_f^+, \; w(t_f) = \mu_e a_e^{\max}, \tag{40}$$

i.e., the terminal condition (21) is satisfied as an equality with the sign "+ ".

Theorem 3. *Let (35) hold and* $(z_0, w_0) \in \Omega^-$. *Then, the pair*

$$\begin{aligned} u_p^-(t) &= -\frac{1}{\alpha} h_p(t) z_f^-, \\ u_e^-(t) &= \frac{1}{\beta}\left[h_e(t) z_f^- - g_e(t) v_f^-\right] \end{aligned} \tag{41}$$

is an open-loop saddle point in the Reduced Game.

Remark 4. *If* $(z_0, w_0) \in \Omega^-$, *then, the optimal trajectory* $(z(t), w(t))$ *generated by the pair* $(u_p^-(\cdot), u_e^-(\cdot))$ *satisfies the terminal conditions*

$$z(t_f) = z_f^-, \; w(t_f) = -\mu_e a_e^{\max}, \tag{42}$$

i.e., the terminal condition (21) is satisfied as an equality with the sign "− ".

3. SPECIAL CASE: FIRST-ORDER PURSUER AGAINST FIRST-ORDER EVADER

In this section, the theory of the previous section is applied to the particular case of (1) – (2) which is of a practical interest. This example illustrates some important features of the game solution.

3.1. Original Pursuit-Evasion Game

If both the pursuer and the evader have the first-order dynamics controller, then in the system (1) – (2), $n_p = 1$, $\bar{A}_p = -1/\tau_p$, $\bar{b}_p = 1/\tau_p$, $\bar{d}_p = 0$, $n_e = 1$, $\bar{A}_e = -1/\tau_e$, $\bar{b}_e = 1/\tau_e$, $\bar{d}_e = 0$, where τ_p and τ_e are the time constants of the pursuer's and the evader's controllers. The pursuer's and the evader's controls are the lateral acceleration commands.

In the OPEG, the controlled system is given by (3), where $x = (y_p, \dot{y}_p, a_p, y_e, \dot{y}_e, a_e)^T$,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1/\tau_p & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1/\tau_e \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 1/\tau_p \\ 0 \\ 0 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1/\tau_e \end{bmatrix}.$$

The cost functional (7) becomes

$$J = (x_4(t_f) - x_1(t_f))^2 + \alpha \int_0^{t_f} u_p^2(t)dt - \beta \int_0^{t_f} u_e^2(t)dt.$$

The matrix Φ_e is

$$\Phi_e(t_f + t_c, t) =$$

$$\begin{bmatrix} 1 & t_f + t_c - t & -\tau_e^2 \psi((t_f + t_c - t)/\tau_e) \\ 0 & 1 & \tau_e[\exp(-(t_f + t_c - t)/\tau_e) - 1] \\ 0 & 0 & \exp(-(t_f + t_c - t)/\tau_e) \end{bmatrix},$$

where $\psi(t) = \exp(-t) + t - 1 \geq 0$.

Thus, the terminal inequality constraint (8) becomes

$$| x_4(t_f) + t_c x_5(t_f) - \tau_e^2 \psi(t_c/\tau_e) x_6(t_f) | \leq \mu_e a_e^{\max},$$

where

$$\begin{aligned} \mu_e &= \tau_e \int_{t_f}^{t_f + t_c} \psi((t_f + t_c - t)/\tau_e)dt = \\ &\tau_e^2(1 - \sigma + \sigma^2/2 - \exp(-\sigma)), \sigma = t_c/\tau_e. \end{aligned} \tag{43}$$

3.2. Reduced Game

The scalar variables (13) and (14) become

$$z(t) = y_e - y_p + (t_f - t)(\dot{y}_e - \dot{y}_p) -$$

$$\tau_p^2 \psi((t_f - t)/\tau_p) a_p + \tau_e^2 \psi((t_f - t)/\tau_e) a_e,$$

$$w(t) = y_e + (t_f + t_c - t)\dot{y}_e +$$

$$\tau_e^2 \psi((t_f + t_c - t)/\tau_e) a_e.$$

The coefficient functions (17) in the differential equations (15) – (16) become

$$
\begin{aligned}
& h_p(t) = -\tau_p \psi((t_f - t)/\tau_p), \\
& h_e(t) = \tau_e \psi((t_f - t)/\tau_e),
\end{aligned} \tag{44}
$$

$$
g_e(t) = \tau_e \psi((t_f + t_c - t)/\tau_e). \tag{45}
$$

The differential equations (15) – (16) become

$$
\begin{aligned}
& \dot{z} = -\tau_p \psi((t_f - t)/\tau_p) u_p + \\
& \tau_e \psi((t_f - t)/\tau_e) u_e,
\end{aligned} \tag{46}
$$

$$
\dot{w} = \tau_e \psi((t_f + t_c - t)/\tau_e) u_e. \tag{47}
$$

The Reduced Game (RG) is formulated for the system (46) – (47) with the cost functional (20) and the terminal inequality constraint (21) where μ_e is given by (43).

3.3. Saddle Point Solutions

In this case, the solvability condition (35) reads

$$
\beta > \beta^* = \tau_e^2 \int_0^{t_f} \psi^2((t_f - t)/\tau_e) dt. \tag{48}
$$

In Fig. 2, the game space decomposition into the sets Ω, Ω^+ and Ω^- is shown for $t_f = 1$ s, $\nu = 0.9$, $a_e^{\max} = 100$ m/s^2, $\alpha = 0.05$, $\beta = 0.3$, $\tau_p = 0.2$ s, $\tau_e = 0.1$ s. For these parameters, $\beta^* = 0.2438$, and the solvability condition (48) is valid.

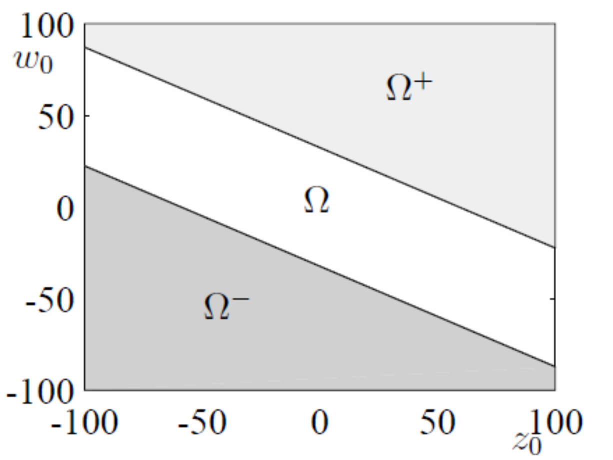


Figure 2: Game space decomposition

3.3.1. Solution for $(z_0, w_0) \in \Omega$

In this case, the optimal controls (38) are calculated by substituting $h_p(t)$ and $h_e(t)$ from (44) into the value of s. The solvability condition (35) yields $s > 0$.

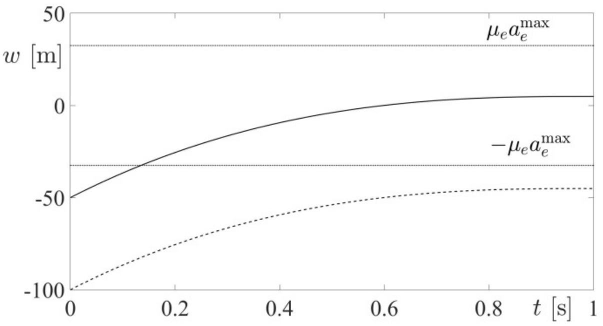


Figure 3: Solution for $(z_0, w_0) \in \Omega$

In this example, the terminal constraint (21) is $|w(t_f)| \le 32.5$. In Fig. 3, two optimal w-trajectories are shown for different initial conditions. If the game starts from $(z_0 = 100, w_0 = -50) \in \Omega$, then $w(t_f) = 4.895$ m satisfies the terminal inequality constraint (the trajectory is shown by the solid line). If the initial position is $(z_0 = 100, w_0 = -100) \notin \Omega$, $w(t_f) = -45.105$ m and the terminal constraint is violated (dashed-line trajectory). The straight lines $w = \pm \mu_e a_e^{\max} = \pm 32.5$ m depict the boundaries of the evader's constraint.

3.3.2. Solution for $(z_0, w_0) \notin \Omega$

We continue using the same parameters as in the previous subsection. In this example, $a = 0.92$, $\mu_e a_e^{\max} = 32.5$, the matrix (28) is

$$
G = \begin{bmatrix} 3.72 & 2.04 \\ -2.04 & 5.91 \end{bmatrix}.
$$

For $(z_0, w_0) = (100, 50) \in \Omega^+$,

$$
b^+ = \begin{bmatrix} 100 \\ 172.5 \end{bmatrix}, \omega_f^+ = \begin{bmatrix} z_f^+ \\ v_f^+ \end{bmatrix} = \begin{bmatrix} 21.22 \\ 10.29 \end{bmatrix}.
$$

For $(z_0, w_0) = (-100, -20) \in \Omega^-$,

$$
b^- = \begin{bmatrix} -100 \\ 12.5 \end{bmatrix}, \omega_f^- = \begin{bmatrix} z_f^- \\ v_f^- \end{bmatrix} = \begin{bmatrix} -23.56 \\ -6.02 \end{bmatrix}.
$$

In Figs. 4 – 5, the optimal trajectories, generated by the saddle-point pairs $(u_p^+(\cdot), u_e^+(\cdot))$ and $(u_p^-(\cdot), u_e^-(\cdot))$, are shown ($z(t)$ and $w(t)$ in Figs. 4 and 5, respectively). It is seen that under the controls $(u_p^+(\cdot), u_e^+(\cdot))$, $z(t_f) = z_f^+ = 21.22$ m and $w(t_f) = \mu_e a_e^{\max} = 32.5$ m, i.e., the terminal equality conditions (40) are satisfied. Correspondingly, under the controls $(u_p^-(\cdot),u_e^-(\cdot))$, $z(t_f)=z_f^-=-23.56$ m and $w(t_f)=-\mu_e a_e^{\max}=-32.5$ m, i.e., the terminal equality conditions (42) are satisfied.

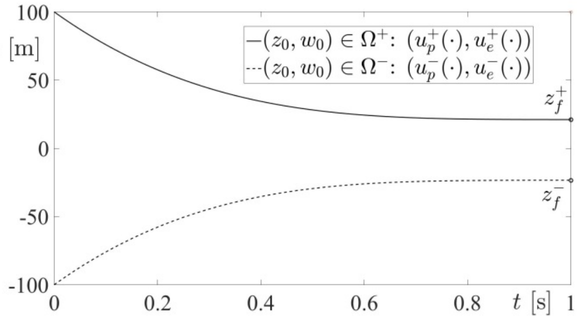


Figure 4: Optimal z -trajectories

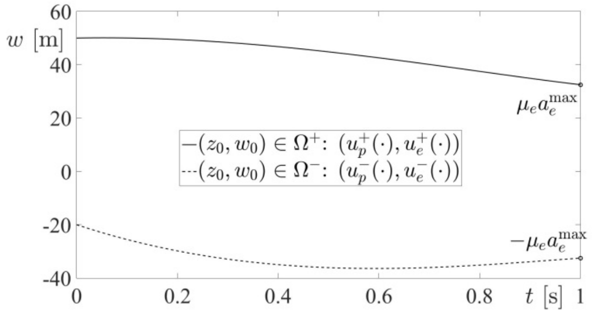


Figure 5: Optimal w -trajectories

The optimal controls $u_p(t)$ and $u_e(t)$ are depicted in Figs. 5 and 6, respectively.

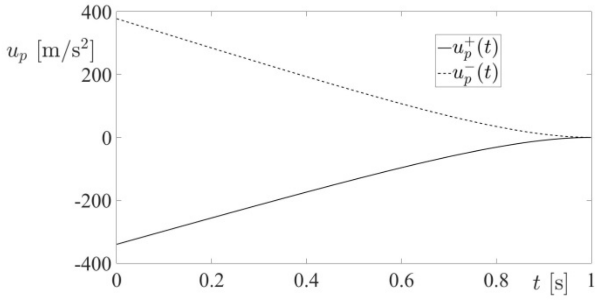


Figure 6: Optimal controls $u_p(t)$

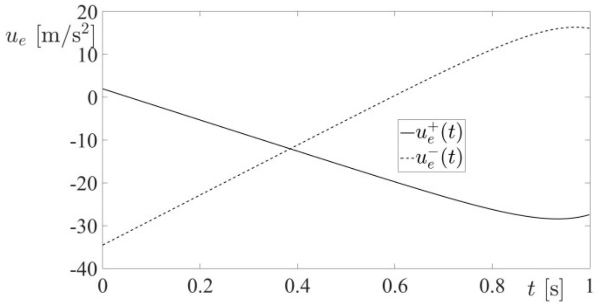


Figure 7: Optimal controls $u_e(t)$

Due to Theorem 2, the pair $(u_p^+(\cdot),u_e^+(\cdot))$ given by (39) constitutes the saddle point in the Reduced Game if and only if $(z_0,w_0)\in\Omega^+=\{w_0+0.92z_0\ge 32.5\}$ (see Fig. 2). Similarly, for $(z_0,w_0)\in\Omega^-=\{w_0+0.92z_0\le -32.5\}$ (see Fig. 2), the saddle point in the Reduced Game is $(u_p^-(\cdot),u_e^-(\cdot))$ given by (39).

Table 1: Results for $(z_0,w_0)\notin\Omega$

$(z_0,w_0)\in\Omega^+$		$(z_0,w_0)\in\Omega^-$	
Controls	Result	Controls	Result
$(u_p^+(\cdot),u_e^+(\cdot))$	1939.2	$(u_p^-(\cdot),u_e^-(\cdot))$	2488.2
$(u_p^+(\cdot),u_e^-(\cdot))$	418.8	$(u_p^-(\cdot),u_e^+(\cdot))$	1463.1
$(u_p^-(\cdot),u_e^+(\cdot))$	2347.7	$(u_p^+(\cdot),u_e^-(\cdot))$	2836.7

Let us chose the initial position $(z_0,w_0)=(100,50)\in\Omega^+$ and calculate the cost functional (20) for three pairs of control functions: for $(u_p^+(\cdot),u_e^+(\cdot))$, for $(u_p^+(\cdot),u_e^-(\cdot))$ and for $(u_p^-(\cdot),u_e^+(\cdot))$. For $(z_0,w_0)=(-100,-20)\in\Omega^-$ we calculate (20) for $(u_p^-(\cdot),u_e^-(\cdot))$, for $(u_p^-(\cdot),u_e^+(\cdot))$ and for $(u_p^+(\cdot),u_e^-(\cdot))$. The results are presented in Table 1. It is seen that for $(z_0,w_0)\in\Omega^+$, the saddle point inequality (22) with $u_p(\cdot)=u_p^-(\cdot)$, $u_e(\cdot)=u_e^-(\cdot)$ is satisfied for $(u_p^+(\cdot),u_e^+(\cdot))$. For $(z_0,w_0)\in\Omega^-$, the saddle point inequality (22) with $u_p(\cdot)=u_p^+(\cdot)$, $u_e(\cdot)=u_e^+(\cdot)$ is satisfied for $(u_p^-(\cdot),u_e^-(\cdot))$.

3.3.3. Feedback realizations of optimal strategies

The complete solution of the Original Game in the class of feedback strategies is the topic of the future research. However, in this paper, we propose the following feedback realization of the saddle point strategies (38), (39) and (41). This realization is based on implementing the open-loop strategy where a current position $(t,z(t),w(t))$ is used instead of the initial position $(0,z_0,w_0)$. The idea of constructing a feedback control based on an open-loop strategy is well known in the control literature (see e.g., (Gabasov, Gaishun, Kirillova, and Prishchepova 1992)).

For $(z_0,w_0)\in\Omega$, the feedback realization of (38) is

$$u_p^0(t,z) = -\frac{h_p(t)z}{\alpha s(t)},\; u_e^0(t,z) = \frac{h_e(t)z}{\beta s(t)}, \qquad (49)$$

where

$$s(t) = 1 + \frac{1}{\alpha}\int_t^{t_f} h_p^2(t)dt - \frac{1}{\beta}\int_t^{t_f} h_e^2(t)dt. \qquad (50)$$

In order to construct the feedback realization for $(z_0, w_0) \notin \Omega$, let us define the matrix

$$G(t) = \begin{bmatrix} s(t) & G_2(t) \\ \\ -G_2(t) & G_3(t) \end{bmatrix}, \qquad (51)$$

where

$$G_2(t) = \frac{1}{\beta}\int_t^{t_f} h_e(t) g_e(t)dt,$$

$$G_3(t) = \frac{1}{\beta}\int_t^{t_f} g_e^2(t)dt, \qquad (52)$$

and the vectors

$$b^+(z,w) = \begin{bmatrix} z \\ w - \mu_e a_e^{\max} \end{bmatrix},$$

$$b^-(z,w) = \begin{bmatrix} z \\ w + \mu_e a_e^{\max}. \end{bmatrix}, \qquad (53)$$

$$\omega_f^+(t,z,w) =$$

$$(z_f^+(t,z,w), v_f^+(t,z,w))^T = G^{-1}(t) b^+(z,w), \qquad (54)$$

$$\omega_f^-(t,z,w) =$$

$$(z_f^-(t,z,w), v_f^-(t,z,w))^T = G^{-1}(t) b^-(z,w). \qquad (55)$$

Then, for $(z_0, w_0) \in \Omega^+$ and $(z_0, w_0) \in \Omega^-$, (39) and (41) become

$$u_p^+(t,z,w) = -\frac{1}{\alpha} h_p(t) z_f^+(t,z,w),$$

$$u_e^+(t,z,w)) =$$

$$\frac{1}{\beta}\left[h_e(t) z_f^+(t,z,w) - g_e(t) v_f^+(t,z,w)\right], \qquad (56)$$

and

$$u_p^-(t,z,w) = -\frac{1}{\alpha} h_p(t) z_f^-(t,z,w),$$

$$u_e^-(t,z,w)) =$$

$$\frac{1}{\beta}\left[h_e(t) z_f^-(t,z,w) - g_e(t) v_f^-(t,z,w)\right], \qquad (57)$$

respectively.

We remind that in this paper, we do not present a strict theoretical justification of feedback solutions (49), (56) and (57).

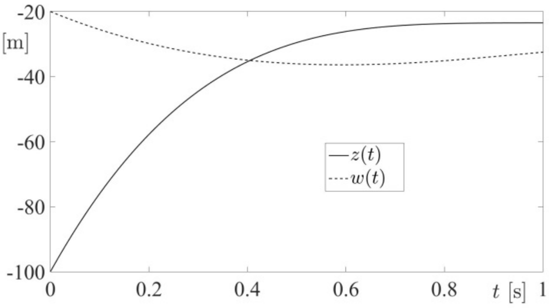


Figure 8: Trajectories generated by (57)

In Fig. 8, the trajectories $z(t)$ and $w(t)$ generated from the position $(z_0, w_0) = (-100, -20) \in \Omega^-$ by the feedback strategies (57), are depicted for the same parameters as in the previous section. These trajectories are close to those generated by the saddle point open-loop strategies (see Figs. 4 and 5). The terminal values are $z(t_f) = -23.484\,\text{m} \approx z_f^- = -23.56\,\text{m}$, $w(t_f) = -32.5\,\text{m} = -\mu a_e^{\max}$.

Now, we examine the behavior of the feedback strategies in the case of noisy state measurements. At each step t_i of the numerical solution, the true values $z_i = z(t_i)$ and $w_i = w(t_i)$ are replaced by $\tilde{z}_i = z_i + \eta_{zi}$ and $\tilde{w}_i = w_i + \eta_{wi}$, where $\eta_{zi} \sim U[-\delta_z, \delta_z]$ and $\eta_{wi} \sim U[-\delta_w, \delta_w]$ are the uniformly distributed measurement errors for z and w respectively. Two cases are distinguished: (I) both the pursuer and the evader obtain the noised state information, and (II) the pursuer obtains the noise information, whereas the evader uses accurate measurements. Let us denote $z_I(t)$, $z_{II}(t)$ and $w_I(t)$, $w_{II}(t)$ the trajectories $z(t)$ and $w(t)$ in the cases (I) and (II), respectively. In Fig. 9, the differences $\Delta_I^z(t) = |z(t) - z_I(t)|$ and $\Delta_{II}^z(t) = |z(t) - z_{II}(t)|$ are shown for $\delta_z = \delta_w = 50$ m. In this simulation, $z_I(t_f) = -23.56$ m, $z_{II}(t_f) = -24.46$ m, yielding $\Delta_I^z(t_f) = 0.46\,\text{m}$, $\Delta_{II}^z(t_f) = 0.98\,\text{m}$. Thus, the terminal error is larger in the case where the evader has an information advantage.

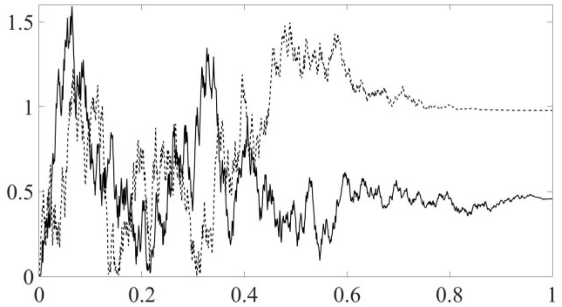


Figure 9: Differences $\Delta_I^z(t)$ and $\Delta_{II}^z(t)$

In Fig. 10, the differences $\Delta_I^w(t) = |w(t) - w_I(t)|$ and $\Delta_{II}^w(t) = |w(t) - w_{II}(t)|$ are shown. It is seen that in the case (I) the difference is large, whereas in the case (II), the difference is close to zero. In this simulation, $w_I(t_f) = -28.21$ m, $w_{II}(t_f) = -32.5$ m, yielding $\Delta_I^w(t_f) = 4.285$ m, $\Delta_{II}^w(t_f) = 0$ m.

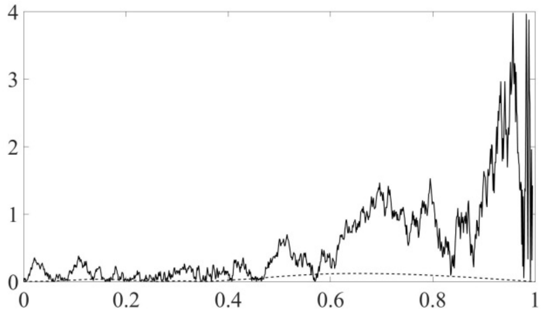


Figure 10: Differences $\Delta_I^w(t)$ and $\Delta_{II}^w(t)$

4. CONCLUSIONS

The practically important special case of a linear-quadratic differential game with a terminal inequality constraint was considered. This game models a pursuit of an evader with two objectives: (i) maximizing the cost functional, and (ii) hitting a stationary target. In this case, both the pursuer and the evader have first-order controller dynamics. The case was treated based on the results presented in the previous paper of the authors and outlined briefly in this paper. In the special case,

- the solvability condition was established;
- the game space decomposition into three non-intersecting sets was constructed;
- the saddle point game solutions were derived;
- the fulfilment of the saddle point inequality was demonstrated;
- the feedback realization of the saddle point open-loop strategies was presented and simulated with state measurement errors.

REFERENCES

Gabasov, R., Gaishun, P.V., Kirillova, F.M., and Prishchepova, S.V., 1992. Optimal feedback for a discrete system with disturbance compensation. II. Synthesis of an optimal regulator, Automation and Remote Control, 53 (4), 500–505

Garcia, E., Casbeer, D.W., and Pachter, M., 2017. Active target defense using first-order missile models. Automatica 78, 139 – 143.

Glizer, V.Y., Turetsky, V., 2015. Linear-quadratic pursuit-evasion game with zero-order players' dynamics and terminal constraint for the evader. In: S.W. Pickl, M. Zsifkovits (eds.), Proceedings of the 16th IFAC Workshop on Control Applications of Optimization CAO 2015, Garmisch-Partenkirchen, 6 - 9 October 2015, IFAC Papers OnLine, vol. 48, pp. 22 – 27.

Lipman, Y., Shinar, J., 1995. A linear pursuit-evasion game with a state constraint for a highly maneuverable evader. In: G.J. Olsder (ed.), New Trends in Dynamic Games and Applications, Annals of the International Society of Dynamic Games, vol. 3, pp. 143 – 164. Birkhauser, Boston.

Rubinsky, S., and Gutman, S. 2014. Three-player pursuit and evasion conflict. Journal of Guidance, Control and Dynamics, 37(1), 98 – 110.

Shinar, J., 1981. Solution techniques for realistic pursuit-evasion games. In: C. Leondes (ed.), Advances in Control and Dynamic Systems, vol. 17, pp. 63 – 124. Academic Press, New York, NY.

Shinar, J., Glizer, V.Y., Turetsky, V., 2013. The effect of pursuer dynamics on the value of linear pursuit-evasion games with bounded controls. In: V. Krivan, G. Zaccour (eds.), Advances in Dynamic Games - Theory, Applications, and Numerical Methods, Annals of the International Society of Dynamic Games, vol. 13, pp. 313 – 350. Birkhauser, Basel.

Turetsky, V., Glizer, V.Y., 2019. Open-loop solution of a defender-attacker-target game: penalty function approach. Journal of Control and Decision, 6(3), 166 – 190.

AUTOMATION AND COGNITION: A METHODOLOGICAL PROCESS FOR HUMAN-CENTRED DESIGN TO MINIMIZE ERRORS

Jean-Marc Mercantini

Aix-Marseille Université, Université de Toulon, CNRS
Laboratoire d'Informatique et Systèmes (LIS - UMR 7020)
Domaine universitaire de Saint Jérôme
Avenue Escadrille Normandie-Niemen
13397 Marseille cedex 20

jean-marc.mercantini@lis-lab.fr

ABSTRACT

Disturbances undergone by a complex system can come as much from its external environment as from the internal elements which constitute it. Complex systems are understood in this study as composed of intelligent machines and humans (IMH), and being in charge to accomplish complex tasks in a collaborative way. Performances of these systems, in terms of robustness, adaptation and resilience, strongly depend on the behaviour of the IMH duo. The works that are the subject of this paper focus on the study of the IMH duo and propose a methodological process using jointly cognitive approaches with formal modelling and simulation to analyse, design and control complex systems. For those systems, human beings are necessarily implied in their global behaviour –including stability–, what crucially calls for a better understanding of their behaviour facing diverse complex situations: normal situations, risky situations, critical and accidental situations. Tools and methods proposed by cognitive Sciences, Cognitive Engineering and Knowledge Engineering allow to take into account the different mechanisms involved in human behaviour to enrich complex system models for a better design and control to munimize human errors.

Keywords: Automation, Cognitive Engineering, Formal modelling, Errors, Complex systems.

1. INTRODUCTION

It may seem curious, a priori, to associate the two terms or concepts: Automation and Cognition. But, from a historical perspective (Mercantini 2015), this association is very pertinent and, maybe even become a concept in its own right, refering to the evolution of the current technological systems (like Robotics and Artificial Intelligence). From the cognitive science dictionary (Tiberghien 2002), "cognition is a function allowing the knowledge realisation and examining the different activities relating to knowledge". Cognition may be also defined (Ganascia 1999) as the ability to integrate multimodal information for generating representations, building associations and elaborating generalizations. The ability to manipulate this knowledge allows the individual to develop a behaviour that depends not only on the environment or the immediate situation.

Originally, the sciences of cognition are based on the study of natural cognition for then evolving toward the study of artificial cognition mobilizing computers to reproduce the mental representations and the functions that allow their treatment. Cognition became an object of scientific study during the twentieth century. Its development is strongly linked to the development of computers used as tools to simulate the cognitive process models, but also used as a metaphor of the brain function where information is received, formatted, processed and stored in memory. This memory is then mobilized to elaborate reasoning and action plans.

In 1956, Cognitive science are emerging from the early development of the cybernetics which is defined by (Wiener 1948) as "the scientific study of control and communication in the animal and the machine". Cybernetics is founded on the key concept of the feedback loop, and its original goal was to provide a unified view of emerging areas of the automatic, the electronic and the mathematical theory of information (Wikipedia 2019).

With cognitive science, the understanding of the outside world changes its viewpoint. It is not external objects that attract attention, but the tool with which they are observed. Cognitive science is concerned with the processes of perception, reasoning, pattern recognition, concept formation, understanding, interpretation, problem solving, control, planning and action. Cognitive engineering and knowledge engineering will propose formal methods, guidelines and norms to design systems in which cognition has a central position.

From the Oxford Handbook of Cognitive Engineering (Lee 2013), Cognitive Engineering is an interdisciplinary approach to the analysis, modelling, and design of engineered systems or workplaces, especially those in which humans and automation jointly operate to achieve system goals. Cognitive engineering characterizes an area of activity (scientific and technical) that is concerned by integrated human-technology systems. It combines knowledge and experience from Cognitive Science, Human Factors, Human-Computer Interaction Design and Systems Engineering (Gersh et al. 2005).

Cognitive Engineering emerged in the early 1980s in response to transformation in the workplace by two major sources (Gersh et al. 2005): (i) computer systems were escaping from the confines of machine rooms and thus design principles were needed to ensure than ordinary people would be able to use them and, (ii) Safety Critical Systems were becoming more complex and increasingly computer controlled; design principles were needed to ensure that teams of skilled technicians could operate them safely and efficiently. Otherwise, this emergence is also linked to the maturation of cognitive science into a discipline whose theories, models and methods are capable of guiding application.

This brief historical review show that Automatics, Automation, Cognition, Cognitive engineering, safety and risk engineering (Mercantini 2015) are closely linked for the design of artefacts that have to be associated with human beings. The combination of Automation with Cognition (and cognitive engineering) leads almost "naturally" to the idea of building new intelligent systems where human beings and artefacts can work together in a coherent organization to face complex tasks and problems. It implies new approaches and new tools to model, to analyse, to control, to predict, to prevent and to protect. The joint consideration of automation and cognition might lead to address automation issues with a more comprehensive and coherent vision, which should lead to the design of new tools marked of consistency.

From a methodological perspective, this paper shows the importance of ontologies to jointly considere automation and cognition with the purpose to minimize human errors within piloting activities of complex systems. Ontologies constitute fundamental tools (i) for structuring a domain (at the conceptual level) as perceived by its actors and (ii) for building computer tools dedicated to assist human actors in solving complex problems in that domain. The Knowledge Oriented Design method (KOD) (Vogel 1988), originally designed to develop Knowledge Based Systems, has been used to elaborate domain or application ontologies.

After describing problems due to complex system piloting, a methodological process is proposed to tackle them with a cognitive perspective, by the use of the KOD method. Results obtained by applying this methodological process to a chosen case is presented and discussed. Finally, we conclude on the suitability of the methodological process proposed to take in account cognition in automation design to minimize errors.

2. ANALYSIS OF PROBLEMS DUE TO COMPLEX SYSTEM PILOTING

2.1. Analysis of the complexity

The generic functional representation of a dynamic system is conventionally represented by a feedback loop (Figure 1). These dynamic systems will be qualified as to be complex because composed of Humans interacting with Intelligent Machines (HIM), and being in charge to accomplish complex tasks in a collaborative way. Performances of these systems, in terms of stability, robustness, adaptation and resilience, strongly depend on the behaviour of the HIM duo. The objectives of these systems can be declined in terms of productivity, reliability, availability, security, quality, but also protection of the environment, risk, or any other objectives more specific to the nature of the piloted process, which can itself be partially or fully automated.

The piloting systems, depending on the nature of the process and the expected performances, can be classified according to different levels of complexity (Table 1). Level 0 and 1 correspond to "classical" commands of the analog or digital type without taking into account the human factor. The levels from 2 to 5 correspond to piloting systems where human supervisors are cooperatively associated to intelligent systems for process control and monitoring and problem solving assistance (CCM or DCCM, in table 1). Human supervisors constitute a homogeneous team (HoHST) when they are trained to work together to perform complex tasks related to the process. They constitute a heterogeneous team (HeHST) when they are coming from diverse origins (cultural, professional, social, academic, etc.) and have not been trained to work together. They may even have opposite objectives and opposite decisions to pilot the process, like it is often the case in crisis situation. In both cases (HoHST et HeHST), human errors have to be taken in account.

Controlled processes may also be classified according to their level of complexity (Table 2). Levels 0, 1 and 2 correspond to processes consisting of more or less complex artificial machines, from a simple machine to an automated industrial plant, without taking in account human operator teams. From level 3 to 5, human operator teams are considered within the automated industrial plant to form a complex system. Level 5 corresponds to complex large-scale systems, that is to say a complete territory that may consist of several industrial systems, an ecosystem, a population and intervention teams. The human component may correspond to operator teams who work in contact with the machines (HoHOT) or to a heterogeneous set of operators in the case of co-activities or dysfunctional or accidental situations, including the intervention teams (HeHOT = HoHOT + external agents). In all cases, the controlled process may be in a "normal functioning state" or in an "abnormal functioning state" that is to say, it may be a faulty process or within a risky or accidental situation.

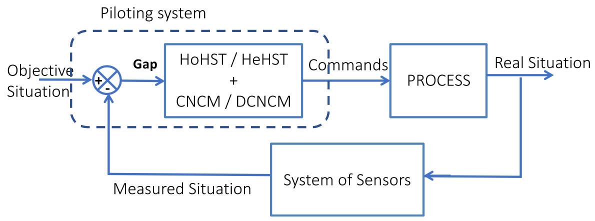


Figure 1: functional representation of dynamic systems

Table 1: Complexities of the piloting systems

Piloting Systems
Complexity Level : Nature
Level 0 : Analog Control
Level 1 : Computer Numerical Control
Level 2 : CCM + HoHST
Level 3 : CCM + HeHST
Level 4 : DCCM + HoHST
Level 5 : DCCM + HeHST
CCM : Computer Control and Monitoring
DCNCM : Distributed Computer Control and Monitoring
HoHST : Homogeneous Human Supervisor Team
HeHST : Heterogeneous Human Supervisor Team

Table 2: Complexities of the controlled processes

The Controlled Process
Complexity Level : Nature
Level 0 : Electromechanical Machine
Level 1 : Automated Machine
Level 2 : Industrial Plant (automated system)
Level 3 : Industrial Plant + HoHOT
Level 4 : Industrial Plant + HeHOT
Level 5 : Territory + HeHOT
HoHOT : Homogeneous Human Operator Team
HeHOT : Heterogeneous Human Operator Team

2.2 The Human-Machine Cooperation

At the level of the piloting system, Human-Machine (H-M) cooperation has been the subject of numerous research studies since the 1980s (Millot 1999)(Millot 2012)(Aguiar 2015)(Benloucif 2018), questioning the automation and optimization of the distribution of supervisory tasks, the ergonomics of the H-M relationship and the behaviour of human operators and supervisors facing diverse work situations.

According to (Millot 1999), H-M cooperation can take two structural forms: the vertical and horizontal structures. With the vertical structure (or hierarchical structure), the human operator / supervisor is responsible for generating all orders. It can use a computer tool for decision support or problem solving support. With the horizontal structure (or heterarchical structure), the decision-support or problem-solving support computer tool is also connected to the control inputs of the process. It becomes an agent at the same hierarchical level as the human operator / supervisor. The problem that arises then is the dynamic distribution of tasks between man and machine.

2.3 The Human Errors

Whether at the level of the controlled process or the control system, the human component regularly and inevitably produces errors that can be interpreted as the result of dysfunctions of cognitive functions such as perception, recognition, comprehension, interpretation, planning. , action, etc. Many authors have studied this problem of human error since the 1980s. Among those that have strongly influenced scientific advances in this area are (Amalberti 1996, 1999) (Hollnagel 1998) (Rasmussen 1982) (Reason 1990) (Vanderhaegen 2003). The results obtained make it possible to better understand their classification, their genesis, their causes, their consequences or their statistics. These human errors are naturally superimposed on the problems of H-M cooperation and those of the complexity of dynamic systems, making the control of work situations more and more complex.

If this complexity can be controlled and mastered in "normal" situations, it can become a real source of danger in critical situations where decisions must be taken and executed under high stress. In this context, the design of new software tools to support piloting tasks must take into account the experience and vision of implied actors according to the issues raised by the complexity of critical situations. Errors and their uncontrolled propagation can call into question the stability of the system or aggravate its state according to whether it is in a normal functionning state or an abnormal functionning state. In both cases, there is the problem of governability, accident avoidance or piloting within accidental situation.

The treatment of errors, with a view to minimizing their occurrence, propagation and consequences, is based on a set of measures that can be combined:

- the training of operators / supervisors on simulator,
- the development of decision support tools that can be integrated in a vertical or horizontal structure,
- the design of these help tools as well as those dedicated to the control and monitoring according to a cognitive logic similar to that of their users (cognitive ergonomics),
- the development of automatic error detection functions and filtering,
- the experience feedback to improve training, procedures, tools and process.

3. METHODOLOGICAL APPROACH

3.1. The methodological process

The proposed methodological approach is based on the assumption that reducing the occurrence and severity of the consequences of pilot errors, despite the increasing complexity of work situations, requires the coherence of conceptual representations of each agent, whether human or artificial, as well as their communication languages. Ontologies and works currently developed by the community of cognitive and knowledge engineers can provide relevant answers to problems raised in the previous paragraph.

The term *ontology* is often associated to the knowledge related to objects of a delimited universe and their relations. Ontology refers to a conceptual language used for the description of this delimited universe (domain). A domain ontology is an example of knowledge level model (Ushold 1998). The emergence of this notion in Knowledge Base System (KBS) engineering comes from the fact that the way to observe the world and its

interpretation are directly dependent of the observer culture, his (her) means to observe it as well as to his (her) intentions. One of the objectives of ontologies is to facilitate the exchange of knowledge between humans, between humans and machines as well as humans via machines (Ushold 1996). In this sense, it becomes necessary to resolve the difficulties caused by observation, representation and interpretation of (normal or critic) situations to facilitate problem solving (intent).
Ontologies can also be defined according to their level of genericity as proposed by Guarino in (Guarino 1998) (Figure 2). The so-called top-level ontologies describe very generic concepts independent of any particular problem or area. They must be "reusable from one domain to another and are designed to reduce inconsistencies in terms defined downstream" (Vandecasteele, 2013). Domain ontologies and task ontologies respectively describe the concepts of a generic domain (such as medicine, production, accidentology, etc.) or the concepts of a generic task (or problem) (such as diagnosis, prognosis, planning, simulation, etc.). They specialize terms introduced by high-level ontologies. Application ontologies (the most specific) describe concepts related to a task (or problem) occurring in a particular field (such as medical diagnosis, road traffic accident diagnosis, industrial planning, etc.) . They are both a union and a specialization of ontologies of tasks and domains (Maedche and Staab, 2001).

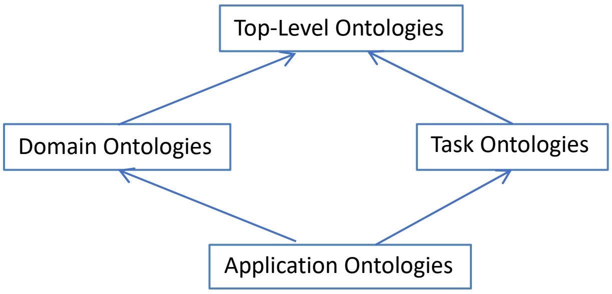


Figure 2 : Classification of ontologies according to their genericity. Arrows represent specialization relationships. From (Guarino 1998).

The proposed methodological process (Figure 3) consists in adopting approaches and methods from Knowledge Engineering (KE) combined with formal modelling. KE approach consists in developing application ontologies aiming to model in a unified way the triplet Td = <Domain, Problem, Method>. In this sense, the ontology structures the Domain according to the Problem to be solved and taking into account the Problem Solving Methods. Tools so built are carrying knowledge shared by actors of a domain, what makes them more effective to accomplish complex tasks in a collaborative way within normal or critical situations.
The inductive process is based on a corpus of documents describing each element of the Td triplet: the Domain corpus, the Problem corpus and the Method corpus. The corpus constitution is really a fundamental step of the process because it has to content an exhaustive knowledge. To illustrate our discourse, previous works can be cited as examples: traffic road accident (Mercantini et al. 2003), aircraft piloting errors (Sadok et al. 2006), industrial plant piloting errors (Mercantini et al. 2004), accidental seaside pollution (Mercantini 2015) or simulation of supply chain vulnerability (Sakli et al 2018).
The Domain corpus must encompass the set of knowledge defining the limits and a deep description of "what is the Domain". It gives a pertinent vision of the cultural dimension of the Domain actors and the different ways the domain can be perceived.
The Problem corpus must encompass a set of representative (pertinent) practical cases of the studied problem. The aim is to get a complete vision of what could happen and the different forms they are taking on. The Problem corpus give a pertinent vision of the wrong behaviours of the Domain actors.
The Method corpus must encompass a set of representative practical cases of the implemented methods to solve the studied problem (practical technics, good practices, formal procedures, quality procedures, etc.). The Method corpus give a pertinent vision of the actors "Know How" of the Domain.
On the second step of the process, the ontology elaboration is based on the "Knowledge Oriented Design" (KOD) method (Vogel 1988). KOD was designed to guide the knowledge engineer in its task of developing knowledge based systems. This method was designed to introduce an explicit model between the formulation of a problem in natural language and its representation in the chosen formal language. The inductive process of KOD is based on the analysis of a corpus of documents, speeches and comments from domain experts, in such a way to express an explicit cognitive model (also called conceptual model).
Depending on the type of result desired, the third step of the process is to use the application ontology to perform one or a combination of the following operations: writing specifications, formal modelling, software modelling. The dashed arrows symbolize this choice or combination.
The final fourth step is the production of the tool. It can be a software tool (computer tool for decision support, problem solving support or simulation), a methodological tool (not necessary computerized), a formal model, a mathematical tool.

3.2. The KOD method

KOD is based on an inductive approach to explicitly express a cognitive model (or conceptual model) based on a corpus of documents, comments and experts' statements. The main features of this method are based on linguistics and anthropological principles. Its linguistics basis makes it well suited for the acquisition of knowledge expressed in natural language. Thus, it proposes a methodological framework to guide the collection of terms and to organize them based on a terminological analysis (linguistic capacity). Through its anthropological basis, KOD provides a methodological framework, facilitating the semantic analysis of the terminology used to produce a cognitive model (conceptualisation capacity). It guides the work of the knowledge engineer from the extraction of knowledge to the development of the conceptual model.

The implementation of the KOD method is based on the development of three successive models: the practical models, the cognitive model and the software model (Table 1). Each of these models is developed according to the three paradigms: <Representation, Action, Interpretation / Intention>.

The Representation paradigm gives the KOD method the ability to model the universe such as experts / actors represent it. This universe is made of concrete or abstract objects in relation. The KOD method provides methodological tools to develop the structure of this universe of knowledge according to this paradigm. The Action paradigm gives the KOD method the ability to model the behaviour of active objects that activate procedures upon receipt of messages. The Interpretation / Intention paradigm gives the KOD method the capability to model reasoning used by experts / actors to interpret situations and elaborate action plans related to their intentions (reasoning capacity).

The practical models are the representation of speeches or documents expressed in the terms of the domain, by means of “taxemes” (static representation of objects – French word), “actemes” (dynamic representation of objects – French word) and inferences (base of the cognitive reasoning pattern). A “taxeme” is a minimum grammatical feature; it is the verbalisation of an object or a class of objects. An “acteme” is the verbalisation of an act or a transformation, a unit of behaviour. An inference is the act or process of deriving logical conclusions from premises known or assumed to be true. The cognitive model is obtained by abstracting the practical models. The cognitive model is composed of taxonomies, actinomies and reasoning patterns. The software model results from the formalization of the cognitive model expressed in a formal language independently of any programming language.

3.3. The ontology building process using KOD

Research work in Ontology Engineering has highlighted five main steps for building ontologies (Dahlgren 1995; Uschold 1996; Aussenac-Gilles 2000; Gandon 2002):

1. *Ontology Specification*. The purpose of this step is to provide a description of the problem as well as the method to solve it. This step allows one to describe the objectives, scope and granularity of the ontology to be developped.
2. *Corpus Definition*. It consists to select among available information sources, those that will allow the objectives of the study to be attained.
3. *Linguistic Study of the Corpus*. It consists in a terminological analysis to extract the relevant terms and their relations. Linguistics is specially concerned to the extent that available data for ontology building are often expressed as linguistic expressions. The characterization of the sense of these linguistic expressions leads to determine contextual meanings.
4. *Conceptualization*. The candidate terms and their relations resulting from the linguistic study are analyzed. The relevant terms are transformed into concepts and their lexical relations are transformed in semantic relations. The result of this step is a conceptual model.
5. *Formalization*. The step consists in expressing the conceptual model by means of a formal language.

The projection of the KOD method on the general approach for developing ontology shows that KOD guides the corpus constitution and provides the tools to meet the operational steps 3 (linguistic study) and 4 (conceptualization) (Table 2).

4. CASE STUDY

4.1. The CLARA 2 project

The purpose of the CLARA 2 (Calculations Relating to Accidental Releases in the Mediterranean) project is to design a problem solving software to assist stakeholders from crisis centres to plan fight actions against marine pollutions (hydrocarbon and chemical products) in Mediterranean area. Stakeholders usually implied in an crisis centre for managing maritime accidents are: the Navy, the National Administrations, the local administrations, the National Meteorology and expert institutes like the French Research Institute for Exploitation of the Sea (IFREMER) or the Centre of Documentation, Research and Experimentation on Accidental Water Pollution (CEDRE). Managing such accidents generates complex and critical work situations. According to table 1 and 2, the complexities of the piloting system and of the controlled process are at level 5, and the structural form of the H-M cooperation is vertical. The potential users of the tool are experts from CEDRE.

Decisions and actions undertaken by crisis center Stakeholders need to mobilize a large number of information from various sources and under high time pressure. These information need to be integrated in a coherent way prior to be interpreted and finally to be the base of any decision and action. Among the main activities carried out by operational center actors it can be cited: situation acquiring, situation analysis, determining fight strategies, choosing the right fight

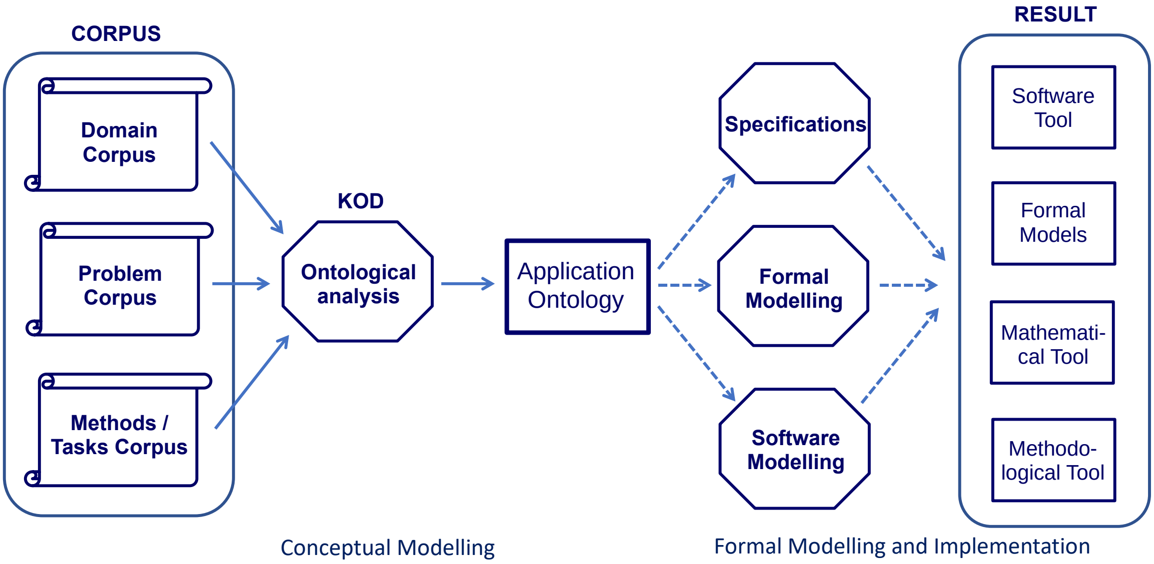


Figure 3: The generic methodological process based on application ontologies to produce specific tools

Table 1. KOD, the three modelling levels according to the three paradigms.

Paradigms Models	Representation	Action	Interpretation
Practical	Taxeme: object static representation	Acteme: dynamic representation of active objects	Inferences
Cognitive	Taxonomy: object static organization according to theirs properties	Actinomy: dynamic object organization	Reasoning Pattern
Software	Classes	Methods	Rules

Table 2. Integration of the KOD method into the elaboration process of ontology.

Elaboration process of Ontology	KOD process	Elaboration process of ontology with KOD
1. Specification		1. Specification
2. Corpus definition		2. Corpus definition
3. Linguistic study	1. Practical Models	3. Practical Models
4. Conceptualisation	2. Cognitive Model	4. Cognitive Model
5. Formalisation		5. Formalisation
	3. Software Model	6. Software Model

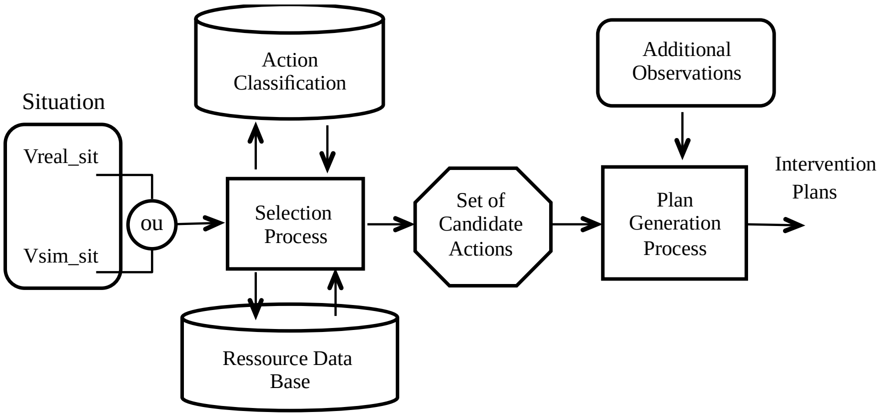


Figure 4: Data flow diagram of the GENEPI module from (Mercantini 2015b)

strategies, choosing the right fight actions, elaborating fight action plans and anticipating future situations.

In this paper, the focus is on the implementation of the generic process (Figure 3) for the study of the GENEPI module (the Generation Module of Intervention Plans – Figure 4) integrated into the CLARA 2 project. See (Mercantini 2015b) for a wider and deeper presentation.

4.2. Elaboration of the Application Ontology

4.2.1 Ontology specification

The domain is that of maritime accidents with the release of pollutant products (hydrocarbon or chemical) and causing a marine pollution. The problem is to assist crisis management teams to elaborate action plan to fight the pollution. The problem solving method consists in the elaboration of a cooperative software tool, which implement the generation process of fight actions against marine pollutions.

4.2.2 Corpus Definition

Documents to be collected must be both representative of the triplet <Domain, Problem, Method> and meet the criteria of suitability required by the three paradigms <Representation, Action, Interpretation / Intention>. The corpus has been established on the basis of documents from CEDRE and REMPEC (the REgional Marine Pollution Emergency Response Centre for the Mediterranean Sea). The types of documents that make up this corpus are the following:

- Documents relating to the evaluation of each fight technique or method,
- Documents about the general organization of emergency plans (plan ORSEC: Organization of the Civil Security Response),
- Return on experience documents about the major maritime disasters such as that of the Erika, Prestige, etc..
- Return on experience documents about maritime accidents of lower magnitudes.
- Quality procedures (from CEDRE) for crisis or accidental event management.

4.2.3 The Practical models

This phase consists in extracting from each document of the corpus, all the elements (objects, actions, and inferences) that are relevant to accident representation and fight action implementation.

Taxeme Modelling

The linguistic analysis is performed in two steps: verbalization and modelling into taxems. Verbalization consists in paraphrasing corpus documents in order to obtain simple sentences allowing to qualify the employed terms. Modelling consists in organizing terms representing objects and concepts of the triplet Td by means of binary predicates such as <Object, attribute, value>. Attribute defines a relationship between the object and a value. Five kinds of predicative relationships are defined: Classifying (is-a, type-of), Identifying (is), Descriptive (position, failure mode, error mode, cause…), Structural (composed-of) and Situational (is-in, is-below, …).

The following example is an extract from the "Prestige" oil tanker accident.

"... On November 13th, 2002, the Prestige oil tanker flying the Bahamian flag, sends an emergency message from the Finisterre Cape ..."

Paraphrases

1. The Prestige is a oil tanker
2. The Prestige flies the flag of the Bahamas
3. On November 13, The Prestige is located at the Finisterre Cape
4. On November 13, the Prestige sends an emergency message

Taxems

1. <Prestige, IS A, oil tanker>
2. <Prestige, FLAG, Bahamas>
3. <Prestige, LOCATION, Finisterre Cape>
4. <Prestige, DATE, November 13th>

The last paraphrase is related to an action, it will be modelled as an actem. The extent of this analysis at the whole Corpus, has produced to the set of taxems needed for the representation of the universe. An object of the real world is modelled by the sum of related taxemes.

Acteme Modelling

Obtaining actemes consists in identifying verbs of the corpus documents that represent activities (or tasks) carried out by human or artificial operators. An activity is performed by an action manager, by means of instruments, to modify the state of the addressee. The following example illustrates how to extract actemes from the Corpus.

"... the Prestige sends an emergency message..."

The activity is "SENDING an emergency message" and it is translated into a 7-tuple (the acteme):

<Action Manager, Action, Addressee, Properties, State1, State2, Instruments>

Where: Action Manager performs the action; Action causes the change; Addressee undergoes the action; Properties represent the way the action is performed; State 1 is the state of the addressee before the change; State 2 is the state of the addressee after the change; Instruments, are means used to cause the change.

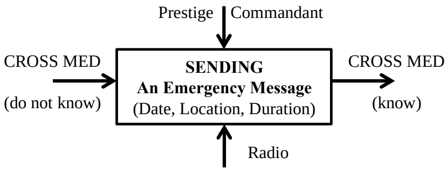
The actem "SENDING an emergency message" is represented as following:

<Prestige Commandant, SENDING an emergency message, CROSS MED, (date, location, duration), CROSS MED (do not know), CROSS MED (know), Radio>.

CROSS MED (Centre Régional Opérationnel de Secours et de Sauvetage en Méditerranée), is the French organism that receives emergency messages from ships. Figure 5 illustrates this acteme and the case of a fight action where the formalism has been extended to take in account suitability criteria:

<Action Manager, Action, Addressee, Properties, Suitability Criteria, State1, State2, Instruments>

Each element of the 7-tuple (or 8-tuple for fight actions) must be previously defined as a taxeme.



Action: FLUSHING	
Components	**Values**
Action Manager: Operator	{Human Means}
Addressee: Substratum	{Sand, Stone, Concrete, Rock, etc.}
Addressee State1	{Polluted, Cleaned}
Addressee State2	{Polluted, Cleaned}
Instruments	{Pump + Water_Hose + Recovery_Means}
Properties	Efficiency
Suitability Criteria	Viscosity Pollutant, Pollution level, Kind Of Substratum

Figure 5: Two examples of actemes. One is in a datagram form (SENDING An Emergency Message) and the other (FLUSHING) in a the table form.

Inferences Modelling

Inferences modelling consists in representing the elements of the corpus that characterize the cognitive activities of humans or machines.

Inferences are the basic elements of the Interpretation / Intention paradigm.

In this study, the Interpretation addresses pollution situations and the Intention concerns fight action planning. Premise propositions are resulting from the interpretation of the situation observed. The conclusion is related to choose (or not) actions (planning).

Let us consider the following example:

"*... dispersants should not be used in areas where water circulation is not good, close to spawning, coral reefs, shell deposits, wetlands and industrial water intakes...* "

where the following inferences have been produced:

IF *spawning areas close* THEN *do not use dispersants*

IF *coral reefs close* THEN *do not use dispersants*

IF *shell deposits close* THEN *do not use dispersants*

IF *industrial water intakes close* THEN *do not use dispersants*

Where *spawning areas close*, *coral reefs close*, *shell deposits close* and *industrial water intakes close* are premise propositions. The observation and interpretation will give them the value True or False. To use dispersants, all the values have to be True. The suitability criteria associated to each fight action are the result of inference analyses.

4.2.4 The cognitive model (conceptualisation)

It consists in developing the cognitive model by abstraction of the practical models. The abstraction from practical models into a cognitive model is based on the operation of classification to produce taxonomies, actinomies and patterns of reasoning.

Taxonomy Building

The first step consists in solving problems induced by homonym and synonym terms, with the objective to build a coherent and common terminology.

The second step consists in analysing the nature of attributes (or relationships) that characterize each object. From the nature of these attributes will depend the building of taxonomies (relationships "kind-of" or "is-a") or others kinds of tree structures (relationships "is-composed-of", “is-on", etc.).

As an example, the term “Skimmer” is meaningful and thus it deserves the concept status. It is significant of a set of recovery devices (previously modelled by means of taxems). As a result of the analysis of the knowledge related to “Skimmer”, the taxonomy of the figure 6 has been built and the “Skimmer” concept is defined through his attributes as follow:

Skimmer attritutes:

<Type, Flow, Quantity, Storage Location, City, Dimension, Weight, Performance Limit, Selectivity, Recovery Rate>

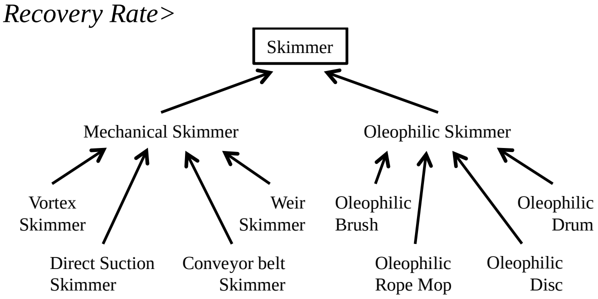


Figure 6: The Skimmer taxonomy (“kind-of” relation)

All the taxems of the corpus are organized in taxonomies and each concept is defined by a set of attributes.

Actems abstraction

One result of the actem analysis is that actems can be organized into five main action categories:

- Actions related to pollutant behaviour,
- Actions related to stricken ship behaviour,
- Actions related to reasoning patterns,
- Actions related to CLARA 2 services,
- Actions related to operations against pollution.

Amongst actions related to pollutant behaviour it can be cited: Evaporation, Dissolution, Drift, Emulsion, etc.

Amongst actions related to stricken ship behaviour, it can be cited: Listing to starboard, Sinking, Sending an emergency message, Requesting evacuation, etc.

The actions related to reasoning patterns such as « Choosing the shoreline clean-up methods » are used to select or to plan fight actions. To be performed, they use the suitability criteria associated to each actem.

The actions of the CLARA 2 services category are implemented to improve the GENEPI functionalities. As examples: Coastal Mapping, Evaluating the Pollution Movement, Evaluating the Pollution Impact.

The actions of the last category are fight actions. They are divided into two main classes: the shoreline clean-up methods and the clean-up methods on the sea. The set of actems from this category has been structured by means of a Taxonomy. Figure 7 is an extract of this taxonomy. Some of these actems can be organized in a structural and temporal way to form actinomies. The interest of this kind of structure is that actions are already planned.

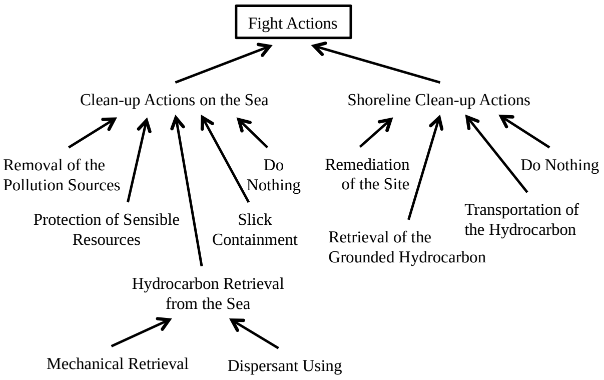


Figure 7: Extract of the Fight Action Taxonomy

4.3. Architecture of the GENEPI module

The architecture of the GENEPI module (Figure 7) has been designed around the ontology enriched with the instances of the concrete classes to constitute the knowledge base (Maedche 2002). For the formal representation of the GENEPI ontology, the frame-based language of the Protégé platform has been used.

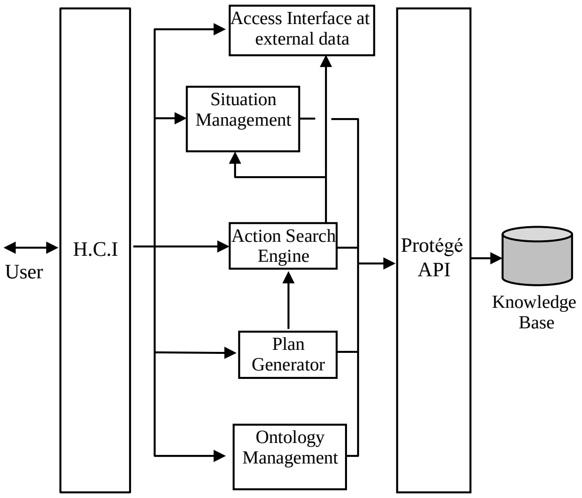


Figure 7: Architecture of the GENEPI module

4.3.1 The Situation Management

Each accident has its own characteristics and for a particular accident, circumstances and context change from one moment to another. To take this into consideration, the notion of Situation is defined. A Situation consists of a set of attributes (S) that characterizes accident and its context. The set of these attributes is a superset of the set of suitability criteria (Ca) associated to fight actions. Thus, attributes common to Ca and S have the same types. Instances of the Situation are obtained from data delivered by the access interface to external data (coming from others CLARA2 modules), and from data supplied by the user (Figure 7).

4.3.2 The Action Search Engine

The search engine receives as input the Situation. As results, it provides four sets of fight actions:

- The set A, which contains the actions where all criteria are verified,
- The set B, which contains the actions where at least one of the criteria could not be assessed by lack of information in the situation,
- The set C, which contains the shares of which at least one criterion was not satisfied,
- The set D, which contains the actions of the set B enriched by criteria not assessed.

Rules to select fight actions are based on the suitability criteria and values taken by the corresponding attributes of the situation. Rules are of the form:

$$c1 \wedge c2 \wedge ... \wedge cn \rightarrow \text{True / False}$$

With c1, c2, ... cn, the criteria associated to a fight action. The conclusion of the rule is about the possibility whether or not to select the action. A criterion is satisfied if the value taken by the corresponding attribute of the situation is compatible the criterion constraints.

Upon the receipt of the Situation, the action-selecting algorithm analyzes actems. From each actem, it extracts the criteria and it applies the selection rules previously presented. According to the results obtained, the actem is placed in the corresponding set (A, B, C or D).

After running the algorithm, if the user is not satisfied with the result, it can enrich the initial situation to assess the criteria that have not been. This new running should reduce the size of the B set, by transferring actions in the set A or in the set C. The algorithm is independent of changes in the ontology.

4.3.3 The Plan Generator

Fight action plans are the result of a collaborative work between GENEPI and the user. From the set A (set of actions where all criteria are satisfied), the user selects actions to constitute the Plan. Once the actions are selected, the Plan Generator produces a document where every action is completely defined: a detailed description of the fight action, a detailed description of human and material means required for its implementation, a detailed description of precautions and safety measures to be followed for its implementation, a reminder of the suitability criteria.

4.3.4 The ontology management module

This module provides users with the functions needed for maintenance (updating, adding and deleting classes, attributes and instances) and consultation (searching knowledge) of the ontology.

3. Conclusion

The aim of this paper was to show that cognitive approaches offer powerful engineering environments to tackle the issues raised by complex system piloting. The responses proposed concerns the design of intelligent machines to assist operators and supervisors in their tasks of problem-solving and decision-making with the purpose to minimize piloting errors.

The methodological process proposed is based on the elaboration of an application ontology combined with the use of formal languages. The purpose of that ontology is to structure the domain according to the problem to solve

and to the problem solving method (the conceptual model). The ontology is obtained by means of a cognitive approach, which consists in applying the KOD method, which has proven to be adequate. The choice of the formal language depends of the final resulting tool.
To illustrate the process implementation, the case study of the CLARA 2 project has been presented.

REFERENCES

Aguiar Y. P.C., Vieira M. F. Q., Galy-Mari E., Santoni C., 2015. Analysis of the User Behaviour When Interacting with Systems During Critical Situations. In: Mercantini JM., Faucher C. (Editors), *Risk and Cognition*, Springer-Verlag, pp 129-154. Intelligent Systems Reference Library, Vol. 80, January 2015.

Amalberti R., 1996. *La conduite des systèmes à risques*. Le travail humain, Editions PUF.

Amalberti R., 1999. Dysfonctionnement des systèmes et dysfonctionnement de la cognition. In J.-G. Ganascia (Coordonnateur), *Sécurité et cognition* (pp. 185-201). Éditions HERMES, Paris.

Aussenac-Gilles N., Biébow B. and Szulman S., 2000. Revisiting Ontology Design: a method based on corpus analysis. EKAW-2000, Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management, R. Dieng et O. Corby (éd.), LNAI 1937, Springer, pp. 172-188.

Benloucif M. A., 2018. Coopération homme-machine multi-niveau entre le conducteur et un système d'automatisation de la conduite. Interface homme-machine [cs.HC]. Université de Valenciennes et du Hainaut-Cambresis, 2018. Français. NNT : 2018VALE0012.

Ganascia J.G., 1999. *Sécurité et cognition*. Hermes, Paris.

Gandon F., 2002. Ontology engineering: a survey and a return on experience. Research Report n° 4396. INRIA Sophia-Antipolis, mars 2002.

Gersh J.R., McKneely J.A., Remington R.W., 2005. Cognitive Engineering: Understanding Human Interaction with Complex Systems. *Johns Hopkins APL Technical Digest*, *26*(4), 377-382.

Gruber T.R., 1993. A translation approach to portable ontology specifications. *Knowledge acquisition*, pp 199-220.

Hollnagel E., 1998. *Cognitive reliability and Error Analysis Method (CREAM)*. Elsevier Sciences Ltd, Oxford.

Lee J.D., Kirlik A., 2013. Cognitive Engineering: History and Foundations. In Lee, J.D., Kirlik, A. (Ed.), *The Oxford Handbook of Cognitive Engineering*. Oxford, New York: Oxford University Press.

Mercantini J.-M., Capus L., Chouraqui E., Tourigny N., 2003. Knowledge Engineering contributions in traffic road accident analysis. In: Ravi K. Jain, Ajith Abraham, Colette Faucher, Berend Jan Van der Zwaag, eds. Innovations in Knowledge Engineering. pp 211-244.

Mercantini J.-M., Turnell M.F.Q.V, Guerrero C.V.S, Chouraqui E., Vieira F.A.Q et Pereira M.R.B, 2004. Human centred modelling of incident scenarios. IEEE SMC 2004, Proceedings of the International Conference on Systems, Man & Cybernetics, pp. 893-898. October 10-13, The Hague, The Netherlands.

Mercantini J.-M., 2015a. Introduction. In: Mercantini JM. and Faucher C. (Editors), *Risk and Cognition*, Springer-Verlag, pp 1-20. Intelligent Systems Reference Library, Vol. 80, January 2015.

Mercantini J.-M., 2015b. Building a domain ontology to design a decision support software to plan fight actions against marine pollutions. In: Mercantini JM. and Faucher C. (Editors), *Risk and Cognition*, Springer-Verlag, pp 197-227. Intelligent Systems Reference Library, Vol. 80, January 2015.

Millot P., 1999. La supervision et la coopération homme-machine dans les grands systèmes industriels ou de transport. In J.-G. Ganascia (Coordonnateur), *Sécurité et cognition* (pp. 125-145). Éditions HERMES, Paris.

Millot P. and Boy G. A., 2012. Human-machine cooperation : a solution for life-critical systems ?. *Work*, vol. 41, no. Supplement 1, pp. 4552–4559.

Rasmussen J., 1982. Human Errors: a Taxonomy for Describing Human Malfunctions in Industrial Installations. *Journal of Occupational Accidents*, v.4, p.311-35.

Reason J., 1990. *L'Erreur Humaine*. Paris, PUF.

Sadok S., Chouraqui E., Mercantini J.-M., 2006. Modeling the deviation to improve safety aircraft piloting activities. In: *Proceeding of The 1st IEE International Conference on System Safety*, 6-8 June 2006, London.

Sakli L., Mercantini J.-M., Hennet J.-C., 2018. Study of Supply Chain Vulnerability Based on Cognitive Engineering and ARIMA Formal Models. In: *Proceeding of the 11th International Conference on Integrated Modelling and Analysis in Applied Control and Automation*, 17-19 September 2018, Budapest.

Tiberghien G., 2002. *Dictionnaire des sciences cognitives*. Colin, A., Paris.

Uschold M., 1998. Knowledge level modelling: concepts and terminology. *The Knowledge Engineering Review*, vol. 13, n° 1, 5-29.

Uschold M. and Grüninger M., 1996. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, volume 11, n° 2, pp. 93-136.

Vanderhaegen F., 2003. *Analyse et contrôle de l'Erreur Humaine*. Lavoisier Hermes Science publications, Paris. ISBN 2-7462-0722-2

Vogel C., 1988. *Génie cognitive*. Paris, Masson (Sciences cognitives).

Wiener N., 1948. *Cybernetics: Or Control and Communication in the Animal and the Machine*. Cambridge, Massachusetts: MIT Press.

Wikipedia 2019. https://fr.wikipedia.org/wiki/Cybernétique

A NEW PROPOSED SHEPHERD MODEL OF A LI-ION OPEN CIRCUIT BATTERY BASED ON DATA FITTING

Hanane Hemi$^{(a)}$, **Nacer K M'Sirdi**$^{(b)}$, **Aziz Naamane**$^{(c)}$

$^{(a)(b)(c)}$Aix Marseille University, CNRS, LIS, SASV, Marseille, France

$^{(a)}$hanane.hemi@uqtr.ca, $^{(b)}$nacer.msirdi@lis-lab.fr, $^{(c)}$ziz.naamane@lis-lab.fr

ABSTRACT

In this paper, the effect of environment parameters on the Lithium ion (Li-ion) battery behaviour is studied. In fact, the experimental database collected from a Lithium-ion battery is used to study its dynamic behaviour and then propose a dynamic battery model who can describe the relationship between the Open Circuit Voltage (OCV), the State Of Charge (SOC) and the ambient temperature (T_a). The proposed Shepherd model of a Li-ion open circuit battery voltage OCV is adjusted by using the experimental database of the CALCE battery group CALCE and then this model is implemented and simulated by using MATLAB/Simulink tools.

Keywords : Data fitting; Li-ion battery; LiFePO4; Open Circuit Voltage; State Of Charge; Electrical vehicle.

1. INTRODUCTION

Energy storage with the Lithium ion (Li-ion) batteries are nowadays more and more deployed in many applications such in e-mobility and stationary storage. The rechargeable Li-ion in the electrical or the hybrid vehicles (EV/HEV) represent 72 % of the total rechargeable Li-ion battery cell market's value in 2022 knowmade.com (2017).

In fact, Li-ion batteries technology has a high power/energy density, with high cell voltage. In addition, it has a high life span, and a low self discharge rate Tian et al. (2017) Waldmann et al. (2018).

For Example, in hybrid and electrical vehicles application, a Li-ion battery pack is considered as a secondary source of power, and it used to support and to reduce the fuel consumption from the primary source of power. During a driving cycle, Li-ion battery can assist a primary source of power during acceleration and can be charged either per it or by recovering braking energy during deceleration. Then, it plays a significant role in the overall efficiency. It allows also to reduce the consumption of the primary power source (as for example, the consumption of a hydrogen in the case of a fuel cell vehicle, or the consumption of the conventional fuel in the case of the Hybrid electric-petroleum vehicles).

However, some environmental or conditions of operation can have a negative effect on its behavior (current, voltages and State of Charge) and also on its life span. The influence of environmental conditions like ambient and storage system temperatures are very important factors that can help to predict its life time decrease and to prevent the probability of its damage and thermal runaway. Therefore, a good knowledge of the system heating and the heat transfer by the battery cell and then the battery pack, becomes crucial.

To protect battery pack from these inconveniences. A Battery Management System (BMS) has to be added to the system. The BMS will indeed may avoid undervoltage / over-voltage, short-circuit and thermal runaway of the battery pack and the then system can operates in safe zone Castano et al. (2015).

Modeling the behaviour of a Li-ion battery becomes then paramount to study a battery performance, depending on several external or internal conditions. In another hand, a behaviour model is useful to estimate and predict the current and future battery state variables. Its use is also interesting to manage the control for a safe battery operation on the other.

In this part, we focus our study on the influence of ambient temperature on battery performance. In particular, the impact of temperature on battery open circuit voltage (OCV).

In fact, the ambient temperature has a significant impact on battery State Of charge (SOC). Then an estimation and / or a prediction of the battery SOC will be harder and then it will be harder too to know the optimal charge/discharge range to control of batteries. However, knowing that OCV and SOC are dependent on ambient temperature, it is important to model this dependence by studying data from experimental battery measurement.

This paper is structured in three sections and a conclusion.The section 2. describes the experimental data and Test Analysis. The section 3. presents and describes the Lithium Ion battery standard OCV model equations. The section 4. presents the Lithium Ion battery Proposed OCV Model. Section 5. presents the conclusions on our results.

2. EXPERIMENTAL DATA AND TEST ANALYSIS

In this paper, the A123 battery cell has been selected. This battery is a LiFePO4 battery and its database had been proposed by the CALCE battery group CALCE Xing et al. (2014). This choice of this type of battery is motivated by the fact that is typically used in HEV applications.

The objective of this section is to present a battery data. In fact, by managing and analyzing this data, the relationship between the OCV, SOC at different ambient temperatures (T_a) can be shown. In order to ensure this, a low-current OCV method had been used and provided for this test.

In fact, the Low-current OCV test consist by using a small current to charge and discharge the battery and then measure the battery output voltage. In this test, this output voltage is approximately equal to the OCV. This test had been performed for different ambient temperature varying from $T_a = -10$ °C to $T_a = 50$ °C with interval of $\Delta T_a = 10$ °C.

For each temperature, in the first step, the battery had been discharged at a constant current rate of $C/20$ until the output voltage reach to the value of lower cut-off voltage of $2\ V$. Then, in the second step, the battery had been charge using a constant current until the out put voltage reached the upper cut-off voltage of $3.6\ V$. These characteristics are given by battery manufacturer for battery security operation. Tab.1 shows the battery characteristics given by battery manufacturer A123Systems.

Table 1: Battery characteristics given by battery manufacturer A123Systems

Type	LiFePO4	
Nominal capacity	1.1	Ah
Nominal voltage	3.3	V
Upper cut-off voltage	3.6	V
Lower cut-off voltage	2	V
Maximum continuous discharge current	30	A

Fig. 1 presents the test bench for battery tests experiments He et al. (2014). Test bench detail has given by References CALCE He et al. (2014).

Fig.2a and Fig.2b present the OCV vs time at the discharge and the charge modes. As shown, the battery takes a long time to charge and discharge caused by the nature of this test (the Low-current OCV test). Also, by comparing OCV curves at different ambient temperatures, the time taken to charge or discharge a battery and OCV value depends on temperature.

Fig.3a and Fig.3b present the OCV vs the Capacity at the discharge and the charge modes. As shown in figures before, ambient temperature has an impact on battery behavior. In fact, the relation between OCV and battery capacity in both charge and discharge modes, is affected by the temperature.

As shown, the OCV is different for each temperature. In fact, at $t = 0\ s$, the value of V_{oc} is different according to temperature. In addition, the value of a couple $V_{oc} = 2\ V$ and battery capacity value at the lower cat-off voltage is affected. In fact, at 50 °C the discharge capacity value is higher comparing by lower temperatures. Also, for each OCV curve measured at ambient temperature, it is shown that is has some differences. By comparing the two temperature -10 °C and 50 °C, the battery behavior is affected and then OCV too.

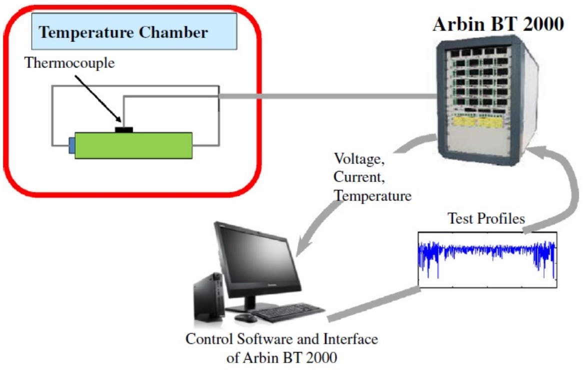
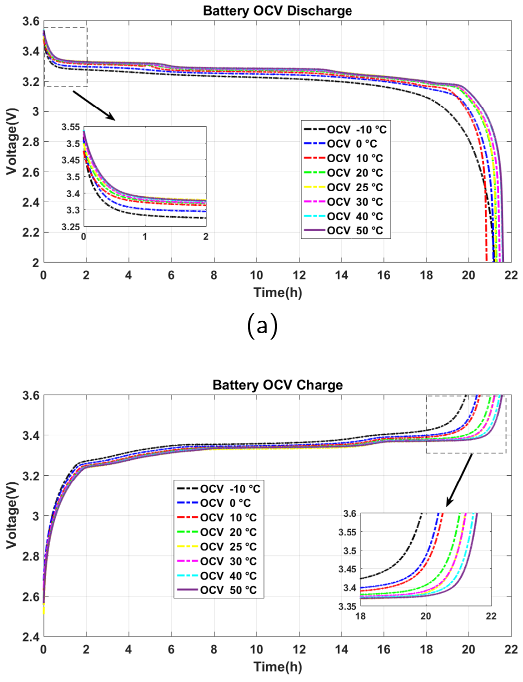


Figure 1: Test bench for battery tests experiments He et al. (2014)



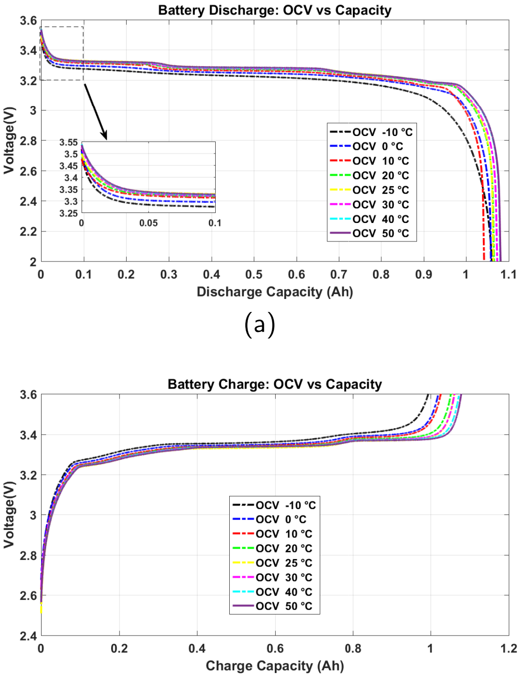
(a)

(b)

Figure 2: OCV vs time at discharge and Charge mode

3. LITHIUM ION BATTERY STANDARD *OCV* MODEL

Li-ion battery is considered as a part of the HEV system model. Therefore, to predict a HEV behavior in real time, a choice of a battery model is important for a real time



(a)

(b)

Figure 3: *OCV* vs Capacity at discharge and Charge mode

HEV operation, specially battery *SOC* for estimation and prediction BMS.
The Shepherd model consider the battery cell model composed by an internal resistance (R) and the open-circuit voltage (OCV). This model describes the mathematical relationship between voltage ($V_{Batt}(t)$), current ($i(t)$) for a constant current discharge. Eq.1 and Eq.2 represent this relationship Raszmann et al. (2017) Li and Ke (2011).

$$V_{Batt}(t) = V_0 - \frac{K.Q}{Q-i.t}.i(t) - R.i(t) \tag{1}$$

$$OCV(t) = V_0 - \frac{K.Q}{Q-i.t}.i(t) \tag{2}$$

where V_0 is the constant voltage (V), Q is the maximum capacity (Ah), $i.t$ is the discharged capacity (Ah), K is the polarization constant $(Ah)^{-1}$ and R is internal resistance. Fig.4 shows the ideal battery curve during discharge Raszmann et al. (2017). This figure shows that battery has tree zones. The fist one is the exponential zone. This zone presents an exponential curve of battery discharge which is described by two points: from the battery full charge (0,V_{full}) until the couple (Q_{exp},V_{exp}). The second zone is the nominal zone. this zone began from the end of exponential zone until the couple (Q_{nom},V_{nom}). This zone shows approximatively constant voltage. The latest zone is the full discharged battery zone. This zone began from the end of nominal zone until full battery discharge (Q_{full},0). This figure is important to find the battery parameters to be used in battery equation Eq.1. However, The Shepherd model equation does not include the exponential zone.

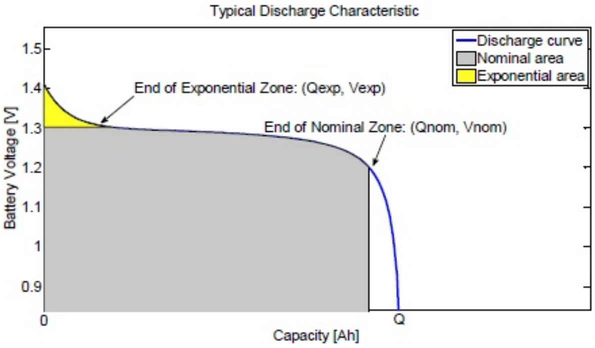


Figure 4: Ideal battery curve during discharge Gallo et al. (2013)

To improve a battery mathematical model to fit to battery charge and discharge curves, the Shepherd model had been modified by adding some terms. Eq.3 and Eq.4 show the *OCV* at discharge and charge modes respectively Raszmann et al. (2017) Gallo et al. (2013) Li and Ke (2011). If the current is positive, then the battery is in discharge mode, $V_{oc} = V_{oc.Discharge}$, as showed in Eq.3. If the current is negative, then the battery is in the charge mode, $V_{oc} = V_{oc.Charge}$, as presented in Eq.4. The *SOC* estimation has been given in Eq.5.

$$\begin{aligned} V_{oc.Discharge} = {} & V_0 - \frac{K.Q}{Q-i.t}.i^* - \frac{K.Q}{Q-i.t}.i.t \\ & + A * e^{(-B.i.t)} \end{aligned} \tag{3}$$

$$\begin{aligned} V_{oc.Charge} = {} & V_0 - \frac{K.Q}{0.1*Q-i.t}.i^* - \frac{K.Q}{Q-i.t}.i.t \\ & + A.e^{(-B.i.t)} \end{aligned} \tag{4}$$

$$SOC = 1 - \frac{1}{Q}\int_0^t i(t).dt \tag{5}$$

$$A = V_{full} - V_{exp} \tag{6}$$

$$B = \frac{\alpha}{Q_{exp}} \tag{7}$$

$$\begin{aligned} K = \beta * [V_{full} - V_{nom} + A(e^{(-B\ Q_{nom})} - 1)] * \\ \frac{Q_{full} - Q_{nom}}{Q_{nom}} \end{aligned} \tag{8}$$

$$R = V_{nom} * \frac{1-\eta}{0.2 * Q_{nom}} \tag{9}$$

$$V_0 = V_{full} + K + R * i_{1C-rate} - A \tag{10}$$

where A is the exponential voltage (V) and also presents the amplitude of the exponential zone which is the difference between the voltage of battery fully charged and the end of the exponentiel voltage V_{exp} as shown in Eq.6 Raszmann et al. (2017).
B is the exponential capacity $(Ah)^{-1}$, or also called the time constant inverse Eq.7. This parameter is calculated from the charge capacity at the end of the exponentiel zone and also uses a constant value α. The α value is usually determined to improve the fit to a battery data Raszmann et al. (2017) Ahmed (2017).
K is the polarization constant $(Ah)^{-1}$ Raszmann et al. (2017), or also named polarizing voltage/resistance factor (V) Ahmed (2017) given in Eq.8. This parameter uses V_{full} value and also the end of nominal zone parameters (Q_{nom}, V_{nom}). In Ref.Ahmed (2017), the constant β is added to improve the fit to a battery data.
V_0 is the constant voltage Eq.10. This parameter describe the battery voltage when a current is equal to zero Raszmann et al. (2017) Ahmed (2017).
R is the battery internal resistance (Ω) Eq.9. η is the efficiency of the battery and $i_{1C-rate}$ is the nominal current. i^* is the filtered current (A).
Generally, parameters of Eq.3 and Eq.4 are determined from battery datasheet provided by the battery manufacturer, especially from the curve of discharge at 1 C rate. However, those parameters depends on the variation of the temperature and the battery lifetime change. Therefore, The need to study and find the relation between those parameters and temperature is important.

4. LITHIUM ION BATTERY PROPOSED *OCV* MODEL

4.1. OCV Model parameter identification

The low-current test had been used on $LiFePO_4$ battery cell to identify the relationship between OCV - SOC and T_a. For each temperature T_a, the voltage and the SOC data are used to fit it with the Eq.3 by using data fitting, and then find parameters of this equation. Fig.5 presents a OCV fit and measured comparison for different temperatures. In this step, for each temperature, the parameters of the Eq.3 are founded by fitting the data to the curve. Then, seven equations for ambient temperature from $T_a = -10$ °C to $T_a = 50$ °C had been founded.
In addition, the Mean Absolute Error (MAE), the Root Mean Squared error (RMS) and the coefficient of determination or also named R-squared (R^2) are used to evaluate the efficiency of the fit as given in Eq.11, Eq.12 and Eq.13. The RMS error will be more sensitive to the variation of error than MAE due the fact that the calculation error is squared. In addition the R-squared will show how close the data measured are to the fitted regression curve. Then, the higher the R-squared, the better the model fits the measured voltage.

$$MAE = \frac{1}{n}\sum_{k=1}^{n} |V_{oc.measured,k} - V_{oc.calculated,k}| \tag{11}$$

$$RMS = \sqrt{\frac{1}{n}\sum_{k=1}^{n} (V_{oc.measured,k} - V_{oc.calculated,k})^2} \tag{12}$$

$$R^2 = 1 - \frac{SSR}{SSTot} \tag{13}$$

$$SSR = \sum_{k=1}^{n} (V_{oc.measured,k} - V_{oc.fit,k})^2 \tag{14}$$

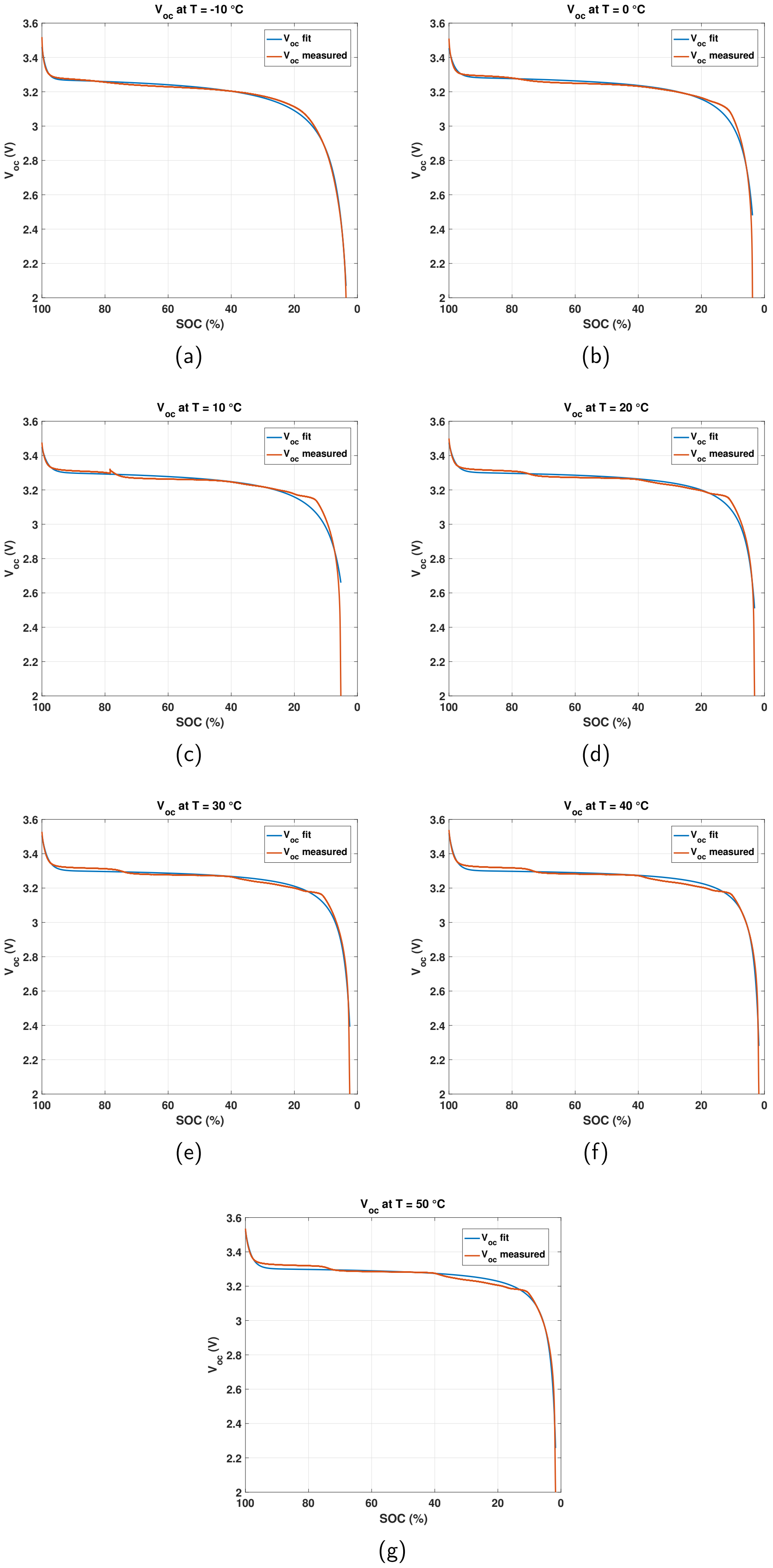
$$SSTot = \sum_{k=1}^{n} (V_{oc.measured,k} - \overline{V_{oc.measured,k}})^2 \tag{15}$$

Where $(V_{oc.measured,k} - V_{oc.calculated,k})$ is the calculation error at time k. The model parameters and statistics are given in Tab.2. The Sum of Squared Error (SSR) is the quantity of how much the measurement points, vary around the estimated regression as shown in Eq.14. The sum of squared total (SSTot) is the quantity of how much the measurement points, vary around their mean as given in Eq.15.

Table 2: Model parameters and statistics for data fitting

T_a	MAE	RMS	R^2
-10 °C	0.0113	0.0093	0.9962
0 °C	0.0233	0.0128	0.9667
10 °C	0.0339	0.0161	0.9176
20 °C	0.0242	0.0140	0.9582
30 °C	0.0207	0.0132	0.9717
40 °C	0.0187	0.0137	0.9763
50 °C	0.0190	0.01344	0.9761

The next step is to study the relationship between those parameters and temperature. For each ambient temperature T_a, the following parameters A, B, K, V_0 are fitted. Fig.6 presents the exponential voltage (A) vs ambient temperature. The equation Eq.16 results from this fit have been chosen as a rational polynomials function. This function has a numerator polynomial and a denominator polynomial with a second degree. The degrees of the numerator and the denominator polynomial have been chosen to get the best curve fitting and minimum number of coefficients. The same approach has been used to find the other equations.
Eq.17 presents the exponential capacity (B) wich is chosen as a constant.



(a) (b) (c) (d) (e) (f) (g)

Figure 5: OCV fit and measured comparison for different temperatures

Fig.7 presents the polarization constant (K) vs temperature. This relationship Eq.18 is chosen as a rational polynomials function with a first degree in numerator polynomial and a denominator polynomial.

Fig.8 shows the constant voltage (V_0) vs temperature. This relationship Eq.19 is also chosen as a rational polynomials function. The numerator and denominator are a third and second degree polynomial respectively.

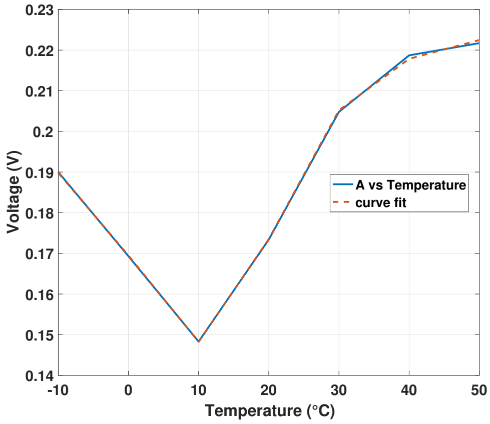


Figure 6: Exponential voltage A vs temperature

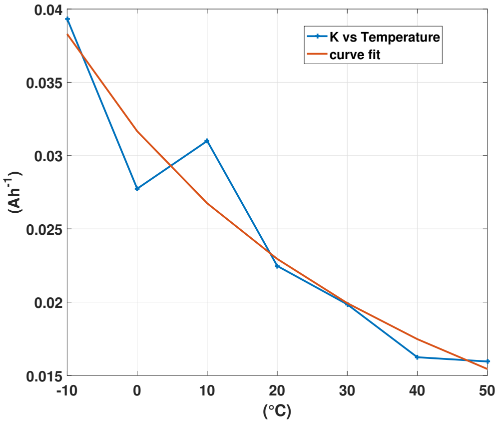


Figure 7: Polarization constant K vs temperature

$$A(T_a) = \frac{p_1\, T_a^2 + p_2\, T_a + p_3}{T_a^2 + q_1\, T_a + q_2} \tag{16}$$

$$B(T_a) = p_1 \tag{17}$$

$$K(T_a) = \frac{p_1\, T_a + p_2}{T_a + q_1} \tag{18}$$

$$V_0(T_a) = \frac{p_1\, T_a^3 + p_2\, T_a^2 + p_3\, T_a + p4}{T_a^2 + q_1\, T_a + q_2} \tag{19}$$

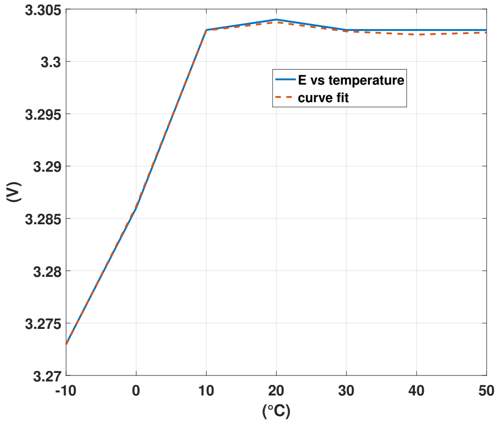


Figure 8: Constant voltage V_0 vs temperature

Table 3: Model parameters and statistics for A, K, and V_0

Parameters	MAE	RMS	R^2
A	0.00047	0.00037	0.9997
K	0.00229	0.00165	0.9167
V_0	0.00023	0.00019	0.9996

Those parameters performance resulting from fitting had been evaluated by using MAE, RMS and R^2 as shown in Tab.3.

The new equation of *OCV* is given in Eq.20. This equation highlights the relationship between the *OCV*, *SOC*, and T_a.

$$\begin{aligned} V_{oc.Discharge}(i.t,t,i^*,T_a) = V_0(T_a) - \frac{K(T_a).Q}{Q-i.t}.i^* \\ - \frac{K(T_a).Q}{Q-i.t}.i.t + A(T_a)*e^{(-B.i.t)} \end{aligned} \tag{20}$$

4.2. Validation of the proposed *OCV* model

The battery model shown in Fig.9 is used to simulate the proposed *OCV* model. In this case, the MATLAB/Simulink software and SimPowerSystems toolbox software packages is used for validation tests. As mentioned in section 2., the test used in this simulation is the Low-current *OCV* test. In fact, it consist by using a small current to charge and discharge the battery.

In this case, the current used to discharge a battery is equal to 0.05 *A* as shown in Fig.9a. Therefore, the output voltage measured is approximately equal to $V_{Batt} = V_{oc}$ as presented in Fig.9b. The *OCV* equation (Eq.20) is implemented as shown in the Fig.9c.

Fig.10 shows the *OCV* vs time measured and simulated comparison for different temperatures. In addition, Fig.11 shows the *OCV* vs *SOC* measured and simulated comparison for different temperatures.

The simulation results shows a small error between sim-

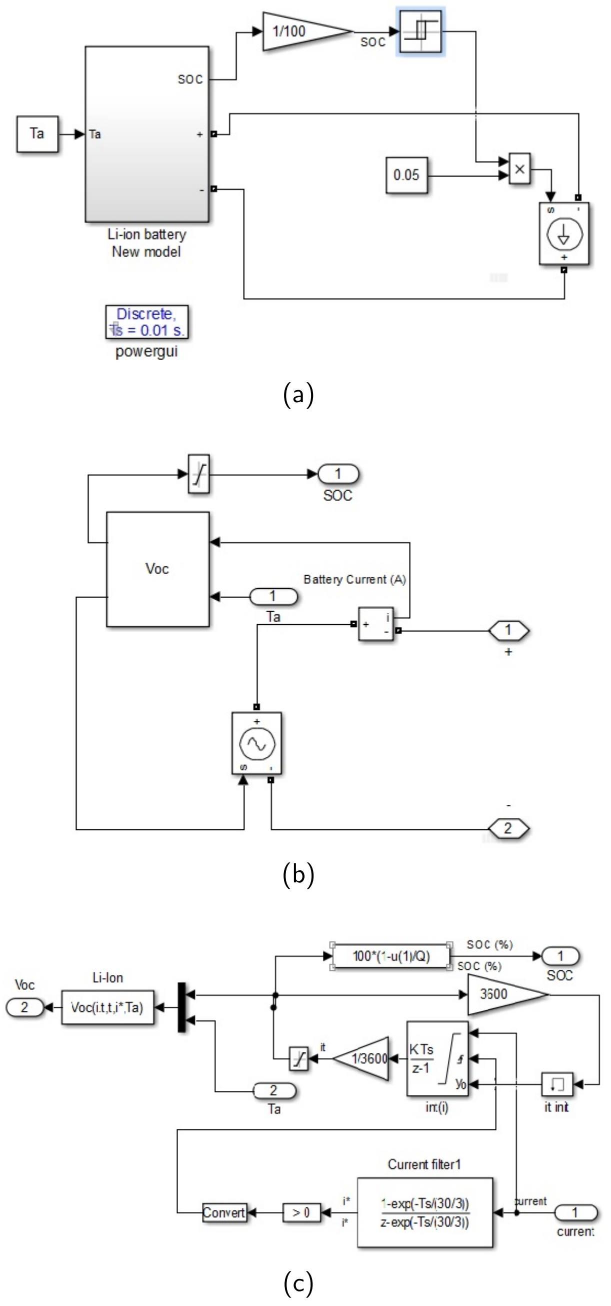


Figure 9: Battery model

ulations and measurements for ambient temperatures -10 °C, 20 °C, 30°C, 40°C, and 50 °C. However for ambient temperature 0 °C (Fig.10b and Fig.11b) and 10 °C (Fig.10c and Fig.11c), it is shown that the error is larger at the end of the discharge mode. Fig.10h and Fig.11h present *OCV* vs time and *OCV* vs *SOC* measured and simulated comparison respectively for all the ambient temperatures.

5. CONCLUSIONS

In this paper, we have proposed and compared the standard and the proposed *OCV* model. The first one is a Shepherd mathematical model modified and the second one is a proposed model based on a Shepherd mathematical model modified with the ambient temperature effect included to the model. The model has been fitted by using the experimental database of a LiFePO4 battery provided from the CALCE battery group CALCE. Also, the Low-current *OCV* test is used in this paper. In fact, this test consist to use a small current to charge and discharge the battery and then measure the battery output voltage. Therefore, the output voltage measured is approximately equal to $V_{Batt} = V_{oc}$.

The proposed model of a battery has been implemented and simulated by using MATLAB/Simulink tools.

The first step of this paper was to analyze a LiFePO4 database and find a relationship between *OCV*, *SOC* and ambient temperature T_a. In this case, a data fitting had been used to fit a data to a curve. In addition, MAE, RMS and R^2 have been used to evaluate the efficiency of the fit. In fact, the fit results presents a small error between the *OCV* fitted and measured for the different ambient temperatures from $T_a = -10$ °C to $T_a = 50$ °C.

The second step was to validate the proposed *OCV* model by simulation. Therefore, the simulation results have allowed to indicate small error between the *OCV* measured and simulated for the different ambient temperatures. However, for the ambient temperature 0 °C and 10 °C, the error is higher mostly in the end of the discharge mode.

In the next work, this model will be improved by reducing this error and also using other tests to study the complete battery system including the internal components.

ACKNOWLEDGMENT

The author would like to thank Professor Michael Pecht, from the CALCE battery group for sharing their experimental data and exchanges to let us use them for the model validation. “CALCE:Center for Advanced Life Cycle Engineering” https://web.calce.umd.edu/batteries/index.h

REFERENCES

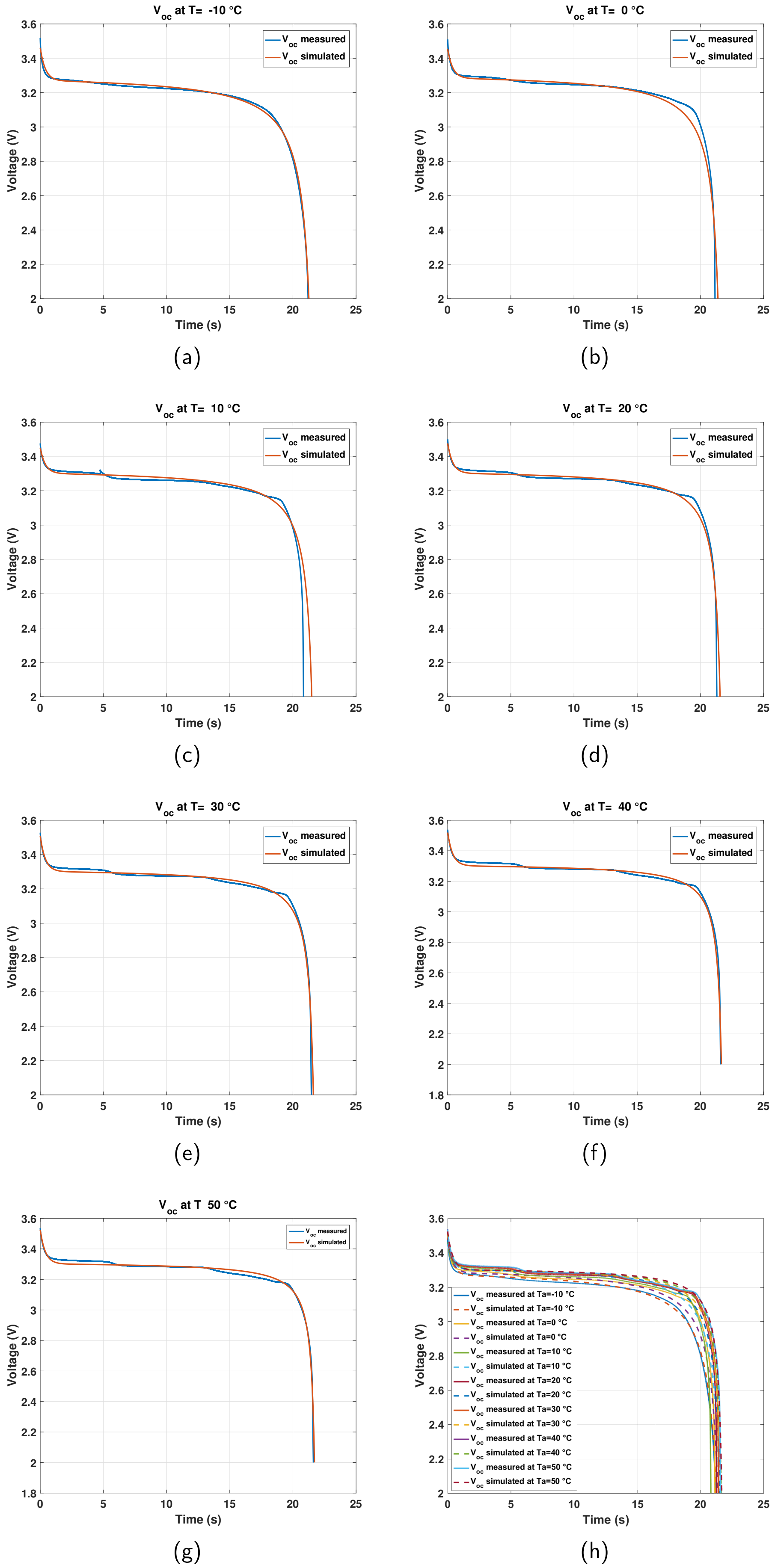
A123Systems, (). A123 systems high power lithium ion apr18650m1a datasheet. `https://www.batteryspace.com/prod-specs/6612.pdf`.

Ahmed M., (2017). Modeling lithium-ion battery chargers in plecs®.

CALCE, (). Center for advanced life cycle engineering. `https://web.calce.umd.edu/batteries/index.html`.

Castano S., Gauchia L., Voncila E., and Sanz J., (2015). Dynamical modeling procedure of a li-ion battery pack suitable for real-time applications. Energy Conversion and Management, 92:396–405.

Gallo D., Landi C., Luiso M., and Morello R., (2013). Optimization of experimental model parameter identification for energy storage systems. Energies, 6 (9):4572–4590.



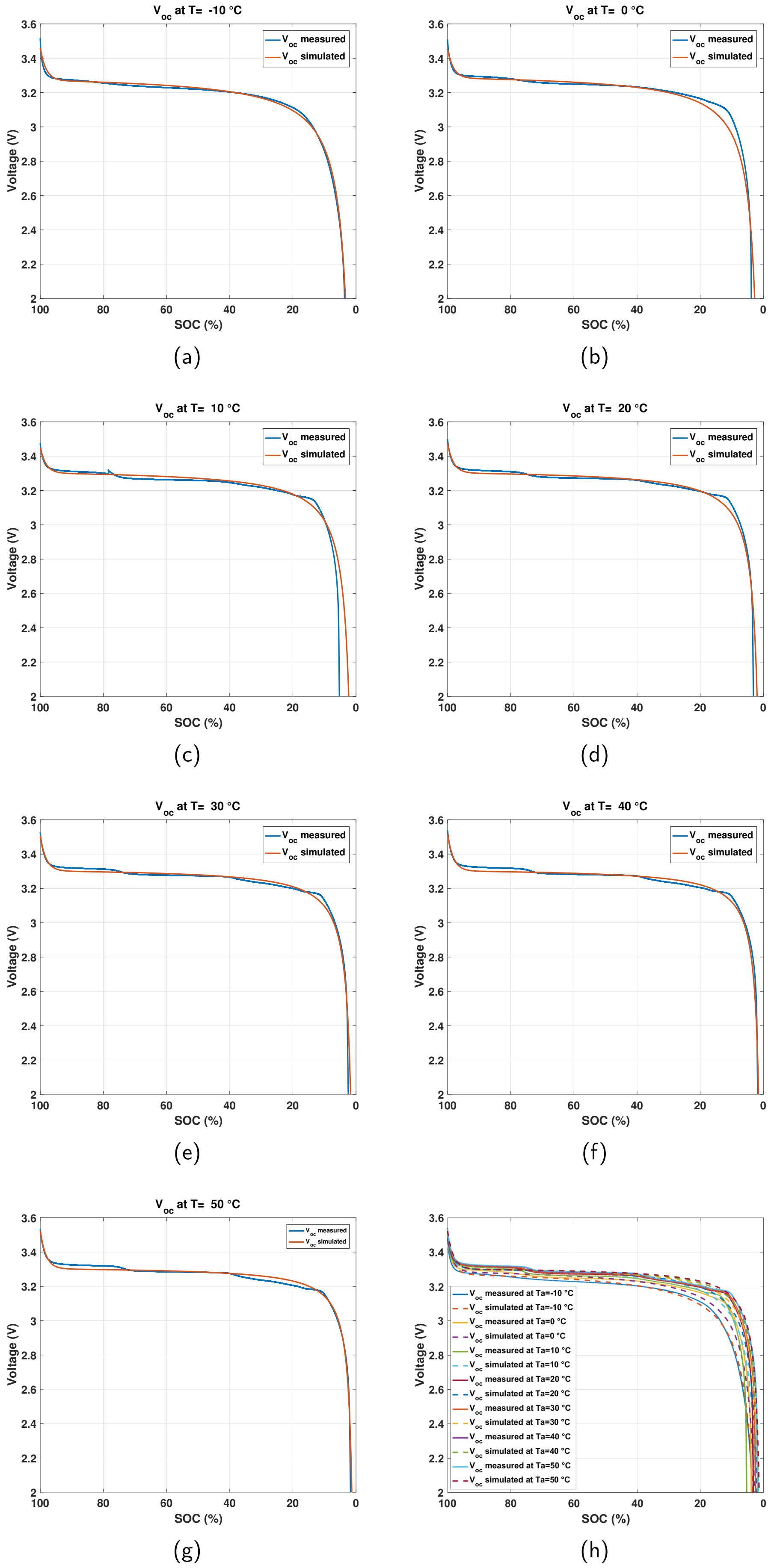
(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 10: *OCV* vs time measured and simulated comparison for different temperatures



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 11: *OCV* vs *SOC* measured and simulated comparison for different temperatures

He W., Williard N., Chen C., and Pecht M., (2014). State of charge estimation for li-ion batteries using neural network modeling and unscented kalman filter-based error cancellation. International Journal of Electrical Power & Energy Systems, 62:783–791.

knowmade.com, (2017). e-mobility: the new eldorado for li-ion batteries. `https://www.knowmade.com/e-mobility-new-eldorado-li-ion-batteries/`.

Li S. and Ke B., (2011). Study of battery modeling using mathematical and circuit oriented approaches. In: Power and Energy Society General Meeting, 2011 IEEE, IEEE, 1–8.

Raszmann E., Baker K., Shi Y., and Christensen D., (2017). Modeling stationary lithium-ion batteries for optimization and predictive control. In: Proceedings of the 2017 IEEE Power and Energy Conference at Illinois (PECI), Champaign, IL, USA, 23–24.

Tian Y., Li D., Tian J., and Xia B., (2017). State of charge estimation of lithium-ion batteries using an optimal adaptive gain nonlinear observer. Electrochimica Acta, 225:225–234.

Waldmann T., Hogg B.I., and Wohlfahrt-Mehrens M., (2018). Li plating as unwanted side reaction in commercial li-ion cells–a review. Journal of Power Sources, 384:107–124.

Xing Y., He W., Pecht M., and Tsui K.L., (2014). State of charge estimation of lithium-ion batteries using the open-circuit voltage at various ambient temperatures. Applied Energy, 113:106–115.

ADAPTIVE ESTIMATION OF THE THERMAL BEHAVIOR OF CPU-GPU SOCS FOR PREDICTION AND DIAGNOSIS

Oussama Djedidi[a], Nacer K. M'Sirdi[b], Aziz Naamane[c]

[a],[b],[c] Aix Marseille University, Université de Toulon, CNRS, LIS UMR 7020, SASV, Marseille, France

[a]oussama.djedidi@lis-lab.fr, [b]nacer.msirdi@lis-lab.fr, [c]aziz.naamane@lis-lab.fr

ABSTRACT

This paper proposes a dynamic behavioral model for temperature variations of systems on chips (SoC) in embedded systems. We use identification techniques (ARMAX modeling) to construct a data-driven online temperature model that estimates the temperature according to the CPU and GPU frequencies, the used RAM and the power consumed by the chip. Furthermore, we used two the Recursive Least Squares (RLS) to estimate the parameters of the ARMAX model. This method allows us to update the parameters of the model online in case of a change in the system or its characteristics. Finally, we validate the temperature model and compare between booth estimation methods.

Keywords: Identification, Embedded systems, Control

1. INTRODUCTION

Since their introduction in 1971, microprocessors have evolved from simple calculators to the center of all technological innovations (Faggin, Hoff, Mazor, and Shima, 1996). The ubiquity of microprocessor-based systems has pushed for the study of their behavior and reliability, notably when used in safety-critical and sensitive systems. Thus, the modeling and diagnosis of the microprocessor-based systems is, now, an ongoing scientific and engineering endeavor.

These systems are evermore evolving and increasing in complexity both on the microarchitectural and process levels, giving rise to new challenges with every new generation. This paper is a part of a project that explores yet another evolution enabled by these systems; the development of avionic cockpits operated by touchscreens (Figure 1). The embedded SoC used in such a critical system is required to be failproof, which require them to be studied from all physical and software aspects. In this particular work, we focus on the thermal behavior of the SoC behind the touchscreen.

Of the many aspects of modeling systems-on-chips (SoC), the temperature is one of the few that links the software, mechatronic and physical characteristics. Hence, many of the recent work studying it were studied on the thermal effects on system radiality (Löfwenmark and Nadjm-Tehrani, 2018), its management for a better reliability (Niu and Zhu, 2017, Zhou et al., 2018) or better scheduling and power management (Li, Yu, and Song, 2018). Our work, however, is oriented towards the real-time surveillance of the chip (Djedidi, Djeziri, and M'Sirdi, 2018). It concentrates on the monitoring of the chip to detect the presence of any anomalies of abnormal behavior.

Figure 1: A prototype of the cockpit of the future by Thales Avionics (Thales, 2017).

In the next section, we further detail the goal of our and put into the context of our previous works. In section 3, we discuss the thermal modeling of CPU-GPU, then we present and apply identification-based modeling to model the thermal devious of the SoC in section 4. Finally, the results and concluding remarks are presented in sections 5 and 6.

2. THE STUDIED SYSTEM

The objective of this work is the mechatronic modeling of the CPU-GPU SoC in embedded systems to predict their behavior. This behavior prediction can then be used for monitoring and diagnosis. This work is also a continuation of the works by Djedidi et al. (2017) and Djedidi, Djeziri, and M'Sirdi (2018), where the authors worked on the modeling and monitoring of systems designed for safety-critical environments. In the first study, Djedidi et al. (2017) developed an incremental interconnected modeling approach, to estimate key variables that determine the operating state of the system (Frequencies and voltages of the CPU, and GPU, Memory Occupation Rate (MOR), Chip Temperature and power consumption). Figure 2 shows a generalized diagram of the established model for mobile CPU-GPU SoC with n CPU cores.

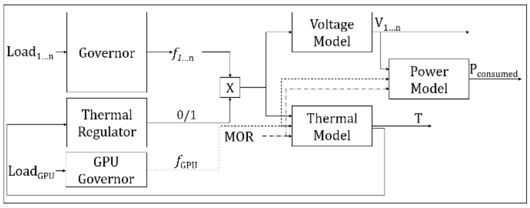


Figure 2: Synoptic diagram of the incremental interconnected model for a CPU-GPU SoC (Djedidi et al., 2017).

The developed model was then used to monitor the state the SoC and for the online detection of several types of faults such as software bugs and environmental faults (Djedidi et al., 2018).

This work focuses on temperature modeling and estimation. It aims to build a model that can be used to predict the temperature values of the SoC according to the current workload. The model is also to be integrated in the interconnected modelling framework as the thermal model (Figure 2). Finally, it is also intended to be used for the diagnosis of the chip in the future.

The case study we used to validate, and test model is a safety-critical certified development board (Figure 3). The board runs on Linux and is Android capable. It has a one core ARM Cortex-A9 processor and is equipped with 1 Gb of RAM (Freescale Semiconductor Inc, 2012b).

Figure 3: View of the test installation with the development board in the middle connected to the monitoring PC.

3. THERMAL MODELING

To model the thermal behavior of an embedded system, the first step is to follow the heat flow.

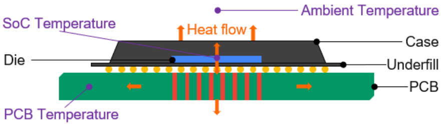


Figure 4: Simplified cross section of a typical SoC with a die containing the CPU, GPU and RAM, installed on a PCB.

Figure 4 shows, how in the SoC, heat is generated by the circuitry containing the processor cores and the RAM. It is then transferred through conduction in two directions to the silicon case (top), the underfill and the C4 bumps (bottom). The latter two would then conduct the heat to the substrate which itself would conduct it to the printed circuit board (PCB). Finally, the heat is dissipated by the case and the PCB to the air through convection and radiation. In these circuits, heat transfer occurs mostly from one layer to another (vertically, in the diagram).

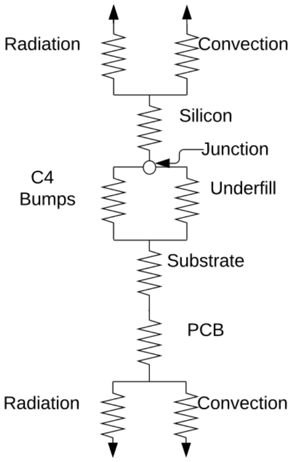


Figure 5: Equivalent thermal resistance circuit of a typical integrated circuit of an SoC.

Figure 5 shows the equivalent thermal resistance circuit (Freescale Semiconductor Inc, 2012a, Wang, Sun, and Pan, 2017). The thermal resistance circuit can be used to build a model that describes the evolution of the temperature from one layer to another. The heat (Q) can be written as:

$$Q = Q_{J/Si} + Q_{J/U} + Q_{J/C4} + Q_{Si/Air} + Q_{C4/Su} + Q_{U/Su} + Q_{Su/PCB} + Q_{PCB/Air} \tag{1}$$

Where Q is equal, in each layer, to the temperature difference divided by the thermal resistance of the said layer (Wang et al., 2017). For instance, the heat transfer between the junction and the silicon encasing is equal to:

$$\mathrm{Q}_{\mathrm{J/Si}} = \frac{\mathrm{T_j} - \mathrm{T_{Si}}}{R_{Si}} \tag{2}$$

Where R_{Si} is the thermal resistance of the silicon. Its value can be determined either by studying the material property (area and thermal conductivity), or empirically through identification.

Models built using this method are crucial for the thermal management of the SoC. They enable engineers and system builders to correctly design the optimal cooling method (heatsink, fan, vapor chambers…). However, they also require a deep knowledge of the system and an estimation of the energy drawn by the SoC and transformed into heat which increases the complexity of the modeling process. It also does not allow for the prediction of the temperature of the SoC according to operating conditions (workload and frequency), nor the change of its value with time and degradation.

Identified models, on the other hand rely mostly on a combination of human expertise and observations to choose which inputs correlate best with the output. In this work, we use an auto-regressive model with exogenous input to predict the temperature of the SoC.

4. IDENTIFICATION-BASED MODELING

ARMAX models are polynomial models used to estimate or predict the output depending on its previous values alongside those of the input vector (Landau and Gianluca, 2006). Our choice settled on a polynomial model—precisely an ARMAX one—because they are dynamic and also fast enough to be used to generate online estimation at frequencies up to 50 Hz. Furthermore, since these models are dynamic, they can also be used to accurately simulate the behavior of the system offline and predict the operating temperature of the SoC.

A discrete ARMAX process can be described by the difference equation (Landau and Gianluca, 2006):

$$\begin{aligned} y(k) = a_1 y(k-1) + \cdots + a_{n_a} y(k_n) \\ + b_{1,1} u_1(k-\tau) + \cdots \\ + b_{m,n_b} u_m(k-\tau-n_b) \\ + e(t) + c_1 e(t-1) + \cdots \\ + c_{n_c} e(t-n_c) \end{aligned} \tag{3}$$

where $y(k)$ represents the output, $u(k)$ the input vector with m width, and $e(t)$ the error value. The parameters $[a_1, \dots, a_{n_a}]$ are the regression parameters, $[b_{1,1}, \dots, b_{m,n_b}]$ are the input parameters, and $[c_1, \dots, c_{n_c}]$ are the moving average parameters. Finally, n_a, n_b and n_c are the orders of the model, and τ is the input delay.

In our case study, the output to be estimated is the temperature T of the SoC, and the inputs are the frequencies of the cores and the memory occupation rate (MOR). These inputs are the variables that correlate the most with the temperature (Mercati, Paterna, Bartolini, Benini, and Rosing, 2017, Niu and Zhu, 2017, Zhou et al., 2018).

The identification process consists of finding the best parameters ($A(q^{-1})$, $B(q^{-1})$, and $C(q^{-1})$) so that the estimated outputs would fit the real ones. In other words, minimize the prediction error ε:

$$\varepsilon_k = y_k - \hat{y}_k \tag{4}$$

$\hat{y}(k)$ is the predicted output. We rewrite equation (3) as a discrete linear model:

$$A(q^{-1}) y(k) = B(q^{-1}) u(k-\tau) + C(q^{-1}) e(k) \tag{5}$$

with q^{-1} being the delay operator, and $A(q^{-1}) = 1 + \sum_{i=1}^{n_a} a_i\, q^{-i}$, $B(q^{-1}) = 1 + \sum_{i=1}^{n_b} b_{j,i}\, q^{-i}$, and $C = 1 + \sum_{i=1}^{n_c} c_{j,i}\, q^{-i}$. Thus, predicted output at the sample k becomes:

$$\hat{y}(k) = \frac{B(q^{-1})}{A(q^{-1})}\, u(k-\tau) + \frac{C(q^{-1})}{A(q^{-1})} e(k) \tag{6}$$

Based upon this, we construct an adaptive predictor. This latter follows the model described in equation (6):

$$\hat{y}_k = \hat{\theta} \varphi_{k-1} \tag{7}$$

with $\hat{y}_k$ being the vector of the temperature value, $\varphi_{k-1} = [y(k-1), \dots, y(k-n_a), u_1(k-\tau), \dots, u_m(k-\tau-n_b), \varepsilon(k-1), \dots, \varepsilon(k-n_c)]$ being a vector composed of the output feedback, the inputs, and the prediction error, and $\hat{\theta} = [-a_1, \dots, c_{n_c}]$ being the vector the parameters vector.

The listing presented in Algorithm 1 is a pseudo-code describing how the parameters of the ARMAX model will be calculated with each iteration for a whole test vector $y(k)$ (Landau, M'Sirdi, and M'Saad, 1986).

Algorithm 1: Pseudo-code for the recursive Least square algorithm.

```
Begin
Define orders: $n_a, n_b, n_c$
Define the input delay: $\tau$
                                        // Initialization of $\varphi_{k-1}$
Initialize $\varphi_{y_{k-1}}$
Initialize $\varphi_{u_{k-\tau}}$
Initialize $\varphi_{\varepsilon_{k-1}}$
$\varphi_{k-1} \leftarrow [\varphi_{y_{k-1}}, \varphi_{u_{k-\tau}}, \varphi_{\varepsilon_{k-1}}]$
$\hat{\theta}_{k-1} \leftarrow$ zeros($\varphi_{y_{k-1}}$.length, 1)
                                        // Initialization of an empty vector
$F \leftarrow 100 \times$ eye($\varphi_{y_{k-1}}$.length)
$y(k) \leftarrow$Read(Output)
While $y(k) \neq$ null do
    $\hat{y}(k) \leftarrow \hat{\theta}_{k-1} \varphi_{k-1}$
    $\varepsilon(k) = y(k) - \hat{y}(k)$
                                        // Recalculation of the parameters vector
    $G \leftarrow F \cdot \varphi_{k-1}$
    norm $\leftarrow 1 + \varphi_{k-1} \cdot G$
    $F \leftarrow F - \frac{G \cdot G^T}{\text{norm}}$
    $\hat{\theta}_{k-1} \leftarrow \hat{\theta}_{k-1} + G \cdot \varepsilon(k)$
                                        // Updating$\varphi_{k-1}$
    $\varphi_{y_{k-1}}$.addFirst($-y(k)$)
    $\varphi_{y_{k-1}}$.poll($n_a + 1$)
    $\varphi_{u_{k-\tau}}$.addFirst(Read(Input(1:m)))
    $\varphi_{u_{k-\tau}}$.poll($n_b + 1 \colon n_b + 1 + m$)
    $\varphi_{\varepsilon_{k-1}}$.addFirst($\varepsilon(k)$)
    $\varphi_{\varepsilon_{k-1}}$.poll($n_c + 1$)
    $\varphi_{k-1} \leftarrow [\varphi_{y_{k-1}}, \varphi_{u_{k-\tau}}, \varphi_{\varepsilon_{k-1}}]$
                                        \\ Reading the next output
    $y(k) \leftarrow$Read(Output)
End
```

5. RESULTS AND DISCUSSION

The results presented in this section are established during controlled experiments. The experiment starts when the data acquisition starts. It begins with two standards benchmarks: AnTuTu (AnTuTu, 2019) and 3DMark (Futuremark Oy, 2019), then goes on to playing and interacting with a Sudoku game, followed by 4 minutes of web browsing, HD video playback, and 4 minutes of standby time.

During this scenario, data is gathered and sent to the monitoring PC where the $\text{ARMAX}_{\text{RLS}}$ model is trained at the same time with a predefined set of orders. Once the best set of orders is found, multiple trials are again launched with a different number of training samples each time. Once the training is finished, the accuracy of the model is then validated online.

Finally, to compare the methodologies, a similar ARMAX model is trained offline with the traditional least squares method (ARMAX$_{LS}$) using the same data as the equivalent ARMAX$_{RLS}$. The model is then validated, again, with same data used to validate the equivalent ARMAX$_{RLS}$ model.

The results presented for the accuracy of the model are the results obtained from validation trials and sets containing about 11×10^4 samples (about 3000 s).

5.1. Order selection

The first set of trials was launched with different sets of orders. The best set is chosen according to two criteria; the Mean Absolute Percentage Error (MAPE) and the average time required to generate estimations by the model.

While higher estimation accuracy is a virtue, models with higher orders may require longer times to generate estimations which can lead to missing the changes of variables values. The time required to generate estimation is also heavily affected by OS scheduling and interruptions on both the device and the monitoring PC. Thus, the optimal model needs to satisfy both accuracy and speed of estimation constraints.

Table 1: Evolution of the accuracy and the time needed to generated estimation of the ARMAX$_{RLS}$ model according to its orders.

Orders of the model			MAPE (%)	Average Sampling Time (s)
n_a	n_b	n_c		
2	2	2	11.1853	18 $\times 10^{-3}$
3	3	3	7.9672	18 $\times 10^{-3}$
4	4	4	1.3757	~30 $\times 10^{-3}$
4	**4**	**2**	**0.8377**	**~25 $\times 10^{-3}$**
5	5	2	0.7215	~65 $\times 10^{-3}$
6	6	2	1.3667	~135 $\times 10^{-3}$

Table 1 displays the MAPE and the average sampling time for several sets of model orders $[n_a, n_b, n_c]$. The data in the tables show that the accuracy of the model increases with the increase of the orders up until [4,4,4], where a lower order for the moving average actually results in an increase in accuracy. Furthermore, Table 1 also show how the average sampling time increases with the order of the model, until it even starts affecting the accuracy of the model due to the longer wait time for estimations. Hence, from these experimental data, the best model orders according to both the accuracy and sampling time are [4,4,2].

5.2. Number of samples

In theory, the ideal training set would contain data representing all possible information about the system. However, in practice, the information in the training set is limited by the sample number and information contained in that sample. Table 2 shows how the accuracy of the model increases with the number of samples. However, it also shows that this increase in the accuracy is not absolute, and the accuracy might decrease even with increase number of samples. This is also shown in the comparisons shown in Figure 6 and Figure 7. Hence, the solution to obtain the best model (In this case, $n = 3000$) is to by comparison of the the Mean Squared Error (MSE) as shown in Algorithm 2.

Algorithm 2: Pseudo-code the selection of the best model.

```
Begin
$MSE_{current} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}$
                                   // $n$ is the number of samples
If ($MSE_{Current} < MSE_{best}$) then
    $MSE_{best} \leftarrow MSE_{Current}$
    $Model_{best} \leftarrow Model_{current}$
End
```

Table 2: The influence of the number of samples on the accuracy of the model.

Number of samples	MAE (°C)	MAPE (%)
500	64.34	169.1200
1000	57.77	151.7027
1500	52.46	137.6220
2000	0.92	2.3688
2500	0.64	1.5077
3000	**0.34**	**0.8377**
3500	0.65	1.5100
4000	0.57	1.5005

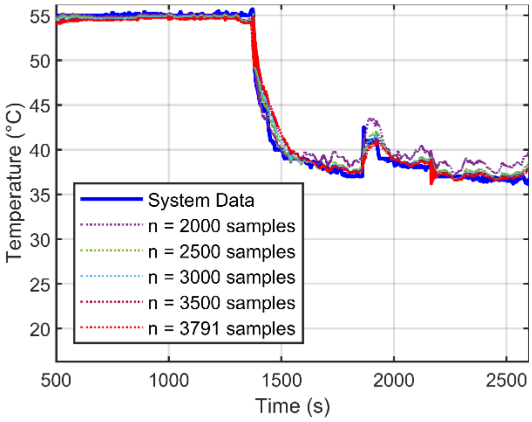


Figure 6: Estimations generated by multiple ARMAX$_{RLS}$ models with different sample numbers in their training sets against system measurements.

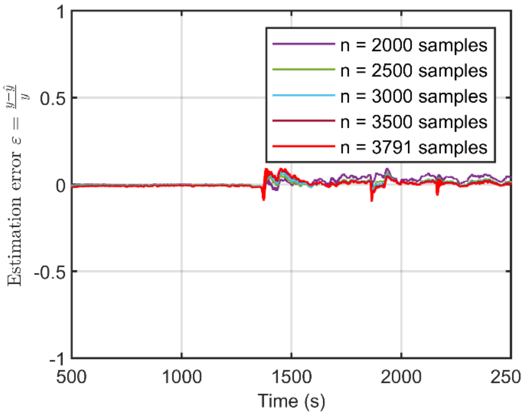


Figure 7: Estimation error (after model convergence) of the $\text{ARMAX}_{\text{RLS}}$ according to the number of samples (n) used in training.

5.3. Comparison with ARMAX_{LS}

Most of the ARMAX models are trained using least-squares method. Using the same data with which the $\text{ARMAX}_{\text{RLS}}$ model was trained, a new ARMAX_{LS} was also trained with [4,4,2] as a set of orders. The model was then tested and validated using the same test data used to validate $\text{ARMAX}_{\text{RLS}}$ one. Table 3 show a comparison of the MAE of the $\text{ARMAX}_{\text{RLS}}$ and ARMAX_{LS} models. While the ARMAX_{LS} shows a slight advantage in its offline validation results, online estimation—our case use— demonstrates an advantage for the $\text{ARMAX}_{\text{RLS}}$. A further comparison of the estimations and estimation errors are shown in Figure 8 and Figure 9.

Table 3: The MAE and MAPE validation results for the $\text{ARMAX}_{\text{RLS}}$ and ARMAX_{LS} models.

Model	**Number of samples**	**MAE (°C)**	**MAPE (%)**
$\text{ARMAX}_{\text{RLS}}$	3000	0.34	0.8377
ARMAX_{LS} (Offline)	4000	0.27	0.6324
ARMAX_{LS} (Online)	4000	0.56	1.3757

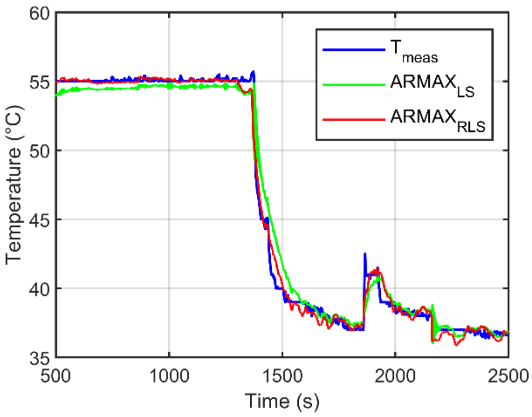


Figure 8: Estimations generated by the $\text{ARMAX}_{\text{RLS}}$ and the ARMAX_{LS} models against system measurements.

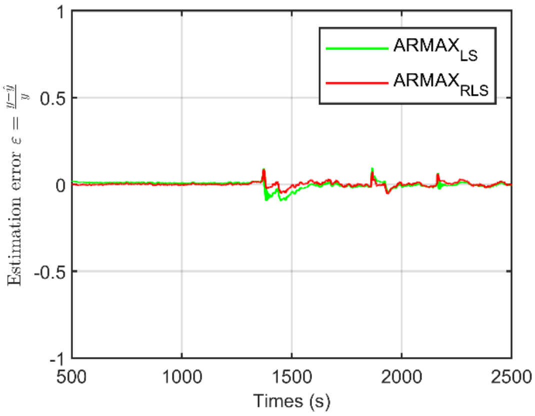


Figure 9: Estimation error (after model convergence) of the $\text{ARMAX}_{\text{RLS}}$ and the ARMAX_{LS} models.

All the results mentioned above clearly validate the $\text{ARMAX}_{\text{RLS}}$. However, the high accuracy of the model (being 99.1623%), along with its speed of estimations are not the only advantages of this model. One last advantage is the capacity to retrain the model at will without stopping the monitoring process this can prove useful when a change in operating condition or a drop in the accuracy occur.

6. CONCLUSION

In this paper, we have built and validated ARMAX model to predict the temperature of embedded SoCs according to the workload and operating conditions. The model is trained using RLS method which offer two clear advantages over the traditional LS method. These advantages are : a better online accuracy, and the capacity of training and retraining the model online without having to stop the monitoring process.

The model offers high accuracy with a mean absolute error of only 0.34°C, and also satisfy the required sampling time.

Having validated the model with satisfactory results, it will now integrated in the interconnected incremental framework we previously developed (Djedidi et al., 2018, 2017). In future works, we will be studying the effects of the temperature on the reliability of the system, and plan on using the $\text{ARMAX}_{\text{RLS}}$ model in the diagnosis of the state of health of the SoC.

REFERENCES

AnTuTu. , 2019. AnTuTu Benchmark - Android Apps on Google Play Retrieved from https://play.google.com/store/apps/details?id=com.antutu.ABenchMark.

Djedidi, O., Djeziri, M. A., and M'Sirdi, N. K. , 2018. Data-Driven Approach for Feature Drift Detection in Embedded Electronic Devices. IFAC-PapersOnLine, 51(24), 1024–1029.

Djedidi, O., Djeziri, M. A., M'Sirdi, N. K., and Naamane, A. , 2017. Modular Modelling of an Embedded Mobile CPU-GPU Chip for Feature Estimation. In Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics (Vol. 1, pp. 338–345). Mardrid, Spain: SciTePress.

Faggin, F., Hoff, M. E., Mazor, S., and Shima, M. , 1996. History of the 4004. IEEE Micro, 16(6), 10–20.

Freescale Semiconductor Inc. , 2012a. i.MX 6 Series Thermal Management Guidelines.

Freescale Semiconductor Inc. , 2012b. i.MX 6SoloX Automotive and Infotainment Applications Processors - Data Sheet. Freescale Semiconductor.

Futuremark Oy. , 2019. 3DMark - The Gamer's Benchmark - Android Apps on Google Play Retrieved from https://play.google.com/store/apps/details?id=com.futuremark.dmandroid.application.

Landau, I. D., and Gianluca, Z. (Eds.). , 2006. System Identification: The Bases BT - Digital Control Systems: Design, Identification and Implementation (pp. 201–245). London: Springer London.

Landau, I. D., M'Sirdi, N., and M'Saad, M. , 1986. Techniques de modélisation récursive pour l'analyse spectrale paramétrique adaptative. Revue de Traitement Du Signal, 3, 183–204.

Li, T., Yu, G., and Song, J. , 2018. Minimizing energy by thermal-aware task assignment and speed scaling in heterogeneous MPSoC systems. Journal of Systems Architecture, 89, 118–130.

Löfwenmark, A., and Nadjm-Tehrani, S. , 2018. Fault and timing analysis in critical multi-core systems: A survey with an avionics perspective. Journal of Systems Architecture, 87, 1–11.

Mercati, P., Paterna, F., Bartolini, A., Benini, L., and Rosing, T. Š. , 2017. WARM: Workload-Aware Reliability Management in Linux/Android. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 36(9), 1557–1570.

Niu, L., and Zhu, D. , 2017. Reliability-aware scheduling for reducing system-wide energy consumption for weakly hard real-time systems. Journal of Systems Architecture, 78, 30–54.

Thales. , 2017. What is new on Avionics 2020? | Thales Aerospace BlogThales Aerospace Blog Retrieved from http://onboard.thalesgroup.com/new-avionics-2020/.

Wang, K. J., Sun, H. C., and Pan, Z. L. , 2017. An analytical thermal model for Three-Dimensional integrated Circuits with integrated micro-channel cooling. Thermal Science, 21(4), 1601–1606.

Zhou, J., Yan, J., Cao, K., Tan, Y., Wei, T., Chen, M., … Hu, S. , 2018. Thermal-aware correlated two-level scheduling of real-time tasks with reduced processor energy on heterogeneous MPSoCs. Journal of Systems Architecture, 82, 1–11.

AN OVERVIEW OF STATE OF CHARGE(SOC) AND STATE OF HEALTH(SOH) ESTIMATION METHODS OF LI-ION BATTERIES

K. Saqli[(a)], H. Bouchareb[(b)], M. Oudghiri[(c)], N.K. M'sirdi[(d)]

[(a),(b),(c)]Sidi Mohamed Ben Abdellah University - National School of Applied Sciences, Fez, Morocco
[(d)]LIS - Informatics and Systems Laboratory (LIS CNRS 7020), Aix Marseille University, CNRS,13397 Marseille Cedex, France

[(a)]saqli.khadijah@gmail.com, [(b)]houda.bouchareb.seii@gmail.com, [(c)]oudghiri.ensafes@gmail.com
[(d)] Kouider-Nacer.Msirdi@lis-lab.fr

ABSTRACT

Battery Management System (BMS) is an essential component for lithium-ion battery-based devices. It provides a variety of functionalities that help improve the overall lifespan of the battery, including states estimation algorithms. An accurate estimation of the battery State Of Health (SOH) and State Of Charge (SOC) is a crucial task that an advanced battery management system should perform.

This paper aims to outline the most relevant battery model types that were used in literature for Electric Vehicle (EV) applications. An overview of the estimation algorithms that estimate the battery state of charge and state of health are presented and simulations of some methods are also illustrated in order to test their accuracy.

Keywords: Battery management system, State Of Health, State Of Charge, Lithium-ion.

1. INTRODUCTION

The global need for a clean and renewable energy sources that can replace fossil energy is essential to create a more sustainable planet. This global shift favours the use of electric vehicles as a safe and clean alternative of fossil fuels in transportation to help reduce air pollution. Due to their high power and energy density, Li-ion batteries are widely used to power electric cars. However, to ensure a long-life cycle and avoid any risk of explosion, a Battery Management System (BMS) is needed to boost the efficiency and guarantee a safe usage of the battery.

A smart battery management system uses the required data to estimate the battery states, whether it's the battery state of charge (SOC), state of health (SOH), or any other battery state that can help improve the performance.

The ability to predict the instantaneous battery state and conditions is a crucial task a BMS should perform with accuracy. The SOC indicates the battery available capacity to help avoid overcharging/discharging the battery pack. An accurate SOC can be achieved using the proper battery model and estimation algorithm.

Different estimation approaches were introduced in literature in the aim of predicting this parameter, Wen-Yeau Chang (2013) presented a classification of the different mathematical methods that were used in literature to predict the battery SOC. However, this variable only is not enough for proper utilization of the battery, since the battery is subject to different ageing mechanisms, it's important to track of the battery health. The battery ageing results in an increment in internal resistance and a decrease in capacity, which affects the performance and the ability to provide the same energy decreases. Therefore, estimation of the battery SOH can be achieved by tracking the change of one of these two metrics. This battery state was the subject of research of different authors who used different estimation algorithms with different battery models to provide an accurate estimation. (Ungurean, Cârstoiu, & Groza, 2016), presented a detailed review of the most relevant models, algorithms and commercial devices that were used in literature to estimate the battery Remaining Useful Life (RUL) and SOH.

This paper outlines the most relevant methods that were used in literature to estimate the battery SOC and SOH. First, we present the battery model categories that were used for EV applications. Then an overview of battery state of health and state of charge estimation algorithms, in particular, the coulomb counting method, internal resistance method, voltage-based method, Kalman filtering based methods, sliding mode observer, fuzzy logic and least squares-based method are presented. an equivalent circuit model with 2 RC networks is used to test the accuracy of some methods to determine the battery SOC and internal resistance.

2. BATTERY MODELING FOR EVS

Building a proper battery model that suits better the target application is a crucial task during BMS design; it helps capture the battery electric and thermal behaviors under different operating conditions, to ensure safe and fast charging for optimal utilization and secure discharging of the battery (Kailong, Kang, Qiao, & Cheng, 2019).

Literature has presented numerous battery models with different complexity and fidelity scales. For EV applications the battery model needs to be simple, computationally efficient and suitable for high discharge rates. Therefore, two main groups of EV battery models were presented: Equivalent Circuit Models (ECM) and Reduced Order Models (ROM) (A., J., K., & Longo S. and Wild, 2016).

2.1. Equivalent Circuit battery Model

Equivalent circuit models on the other hand are widely used when developing a BMS for EV application thanks to their speed, simplicity and reasonable accuracy (Rincon-Mora, 2006).

The model uses a combination of resistors, capacitors and voltage sources, to describe the battery behavior under load. Starting by the most basic model that represents the battery as an ideal voltage source mounted on series with internal resistance to describe the voltage polarization, researchers have include different electrical components, in the interest of building improved versions which take into account the dependence of the battery cell on state of charge, temperature, and other characteristics (Kempera, Li, & Kum, 2015).

Figure 2 illustrates a typical ECM that was widely used in literature for battery states estimation. The resistor-capacitor networks are used to describe the battery charge transfer or diffusion processes (Kailong, Kang, Qiao, & Cheng, 2019).

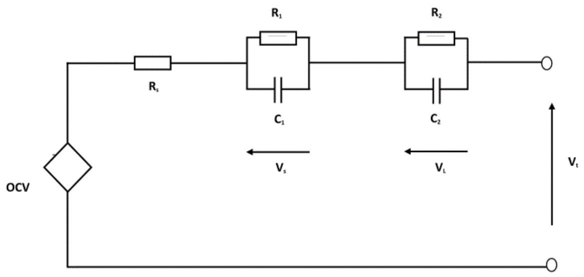


Figure 2 : Second order equivalent circuit model

2.2. Reduced-Order battery Model

Reduced-order models are simplified version of electrochemical battery model (A., B., & M., 2016) . The governing Partial Differential Equations (PDE) equations that describe the electrochemical reactions inside the battery are approximated into low order systems of ODEs equations using a set of Model Order Reduction (MOR) techniques (Fan & Canova, 2017).

The most basic electrochemical model is known as Single Particle Model (SPM) (Figure 1), it represents each electrode of the battery by a single spherical particle ignoring thermal effects, approximating spatial and time characteristics at the separator region to 0 and assuming all unknown states to be scalar and uniform Figure 1. SPM are simple and they can be adopted for real time applications, however, they lack accuracy at high C-rate.

To overcome this limitation an extended version of these model that incorporates the electrolyte dynamics has been developed and they were proved to maintain a high accuracy prediction even at high C-rates conditions (Moura, Argomedo, Klein, & Krstic, 2016).

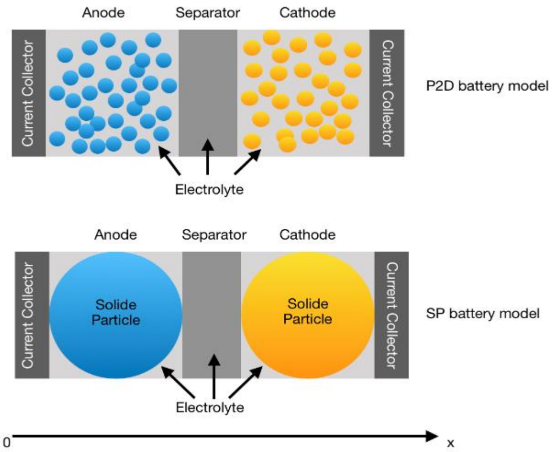


Figure 1: P2D and single particle battery model

Another version that aims to reduce the computational burden of the P2D model was presented in the literature as the Simplified P2D model (SP2D), it describes the dynamic concentration profiles derived from the P2D model to help improve the accuracy of the BMS (G., X., & M., 2018).

Once the battery model has been decided, it can be used as an input for the states estimation algorithms whether it is the SOC, SOH or any other state.

3. BATTERY SOC AND SOH ESTIMATION

3.1. Coulomb Counting

Coulomb counting, also known as Ampere-hour method is one of the most common techniques that were used to estimate the battery states and especially the battery state of charge. As the name suggests, this method calculates the accumulated current that flows in or out during the charge-discharge process to determine the battery state of charge (equation 1) (Fleischer, W., Z., & D.U., 2013) or state of health (equation 2) (Ungurean, Cârstoiu, & Groza, 2016).

$$SOC(t) = SOC(t_0) - \frac{1}{Q_{rated}} \int_{t_0}^{t} \eta(t) I_{bat} dt \quad (1)$$

Where $SOC(t_0)$ represents the initial SOC, Q_{rated} is the rated capacity and I_{bat} is the battery current.

$$SOH(t) = \frac{1}{Q_{rated}} \int_{t_0}^{t} I(t)dt \qquad (2)$$

Where I is the discharge current.

Coulomb counting method requires an accurate estimation of the initial SOC and a precise measurement of the battery current to be able to estimate the battery states as correctly as possible. However, in reality, the measured current includes sensor noise, and it doesn't take into consideration the self-discharge current and current losses during charging and discharging, which makes the measured current different from the true cell current.

To overcome those limitations, several researchers have proposed modified versions of the coulomb counting method. For example, in (Berecibar, et al., 2016) The initial value of the state charge was first estimated using a SOC-OCV mapping function, and then a periodic re-calibration of the capacity was performed. The measured results have shown a reliable estimation.

3.2. Direct Resistance Estimation Algorithm

The life evolution of Li-ion battery cells is affected by different degradation mechanisms that can be represented by two measurable quantities: Capacity loss and increment of internal resistance. Consequently, we can estimate the battery state of health by observing the changes of these two parameters.

By observing the step change in the voltage curve during the discharge process, the battery internal resistance can be approximated using the following equation (Yu, et al., 2017):

$$R = \frac{\Delta U}{\Delta I} \qquad (3)$$

Where ΔI represents the current step-change and ΔU is the corresponding voltage to the same step-change.

The estimation of the battery internal resistance using the DRE algorithm gives noisy results due to noise in the measurement reading. Therefore, Lievre et al. (2016) have used an Exponentially Weighted Moving Average (EWMA)(equation 4) filter to reduce the noise while maintaining the efficiency of the algorithm.

$$EWMA_k = \lambda \times R_s + (1 - \lambda) \times EWMA_{k-1} \qquad (4)$$

Where, $EWMA_k$ represents the current filtered resistance estimate, λ is a tunable constant that represents the depth of memory, $EWMA_{k-1}$ is the previous value of the filtered resistance estimate and R_s is the raw ohmic resistance.

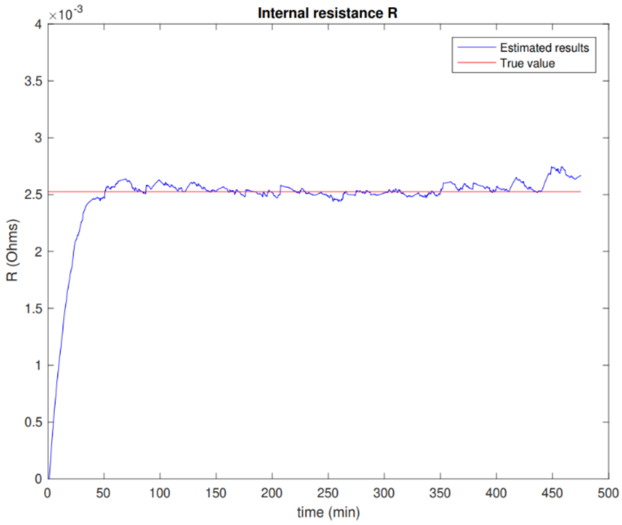


Figure 3 : Internal resistance estimation using DRE approach

Figure 3 shows the implementation of this approach using the same battery model as 3. The estimated results slightly converge to the true value of the battery internal resistance. The implementation of this method requires a smaller memory space since no training data or initial battery characterization are needed. Also, it does not require complex matrix calculation, which makes it suitable for an embedded system (Mathew, Janhunen, Rashid, Long, & Fowler, 2018).

3.3. Open Circuit Voltage (OCV)

Another simplistic approach to estimate the battery states is by measuring the open circuit voltage (OCV) of the cell. Literature has proven a strong dependence between OCV and SOC of the battery cell. This voltage-based method gives the equivalent SOC value of the given voltage value using the OCV vs. SOC discharge curve of the battery.

Based on a simplified electrical model we can define the battery OCV as follow:

$$U_{OCV}\big(SOC(t)\big) = U(t) - R_0 I(t) \qquad (5)$$

Where U_{OCV} is the battery OCV, $U(t)$ is the battery terminal voltage, R_0 is the internal resistance and I(t) is the battery current.

Using the same equivalent circuit model previously discussed, we can implement this method in MATLAB. The estimated SOC was filtered using the EWMA approach to reduce noises (see Figure 4).

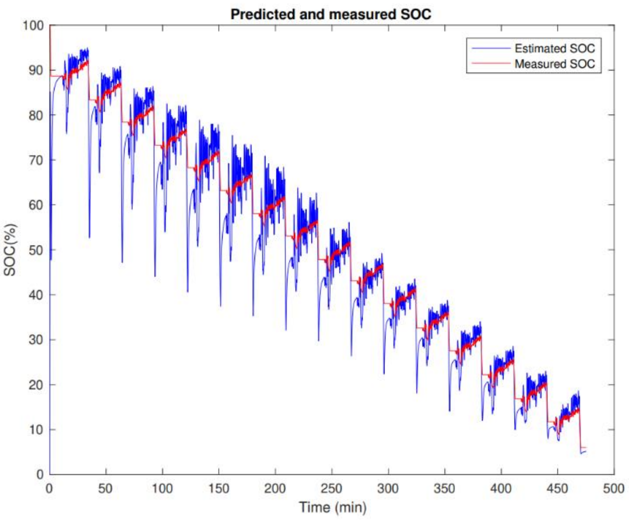


Figure 4 : SOC estimation using OCV

Figure 4 illustrates the estimate and the true SOC curve of the Li-ion battery cell, and by observing these curves we notice that the predicted values are noisy and that can be explained by the fact that this voltage-based method doesn't take into consideration the diffusion voltage and the hysteresis.

OCV has been used also to estimate the battery SOH. By knowing the SOC/OCV relation the estimation of the internal resistance can be easily conducted. (Mathew, Janhunen, Rashid, Long, & Fowler, 2018) used a combination of OCV and Coulomb counting approach to estimate the battery SOH. However, others considered this voltage-based method unsuitable for estimating SOH, (A., et al., 2016) present some disadvantages of using this method and tried to eliminate the OCV from the equations to simplify the computation of the battery SOH.

3.4. Kalman filtering

Kalman filter is a model-based algorithm that uses the mathematical representation of a linear system to determine its states. The literature defines this approach as a sturdy algorithm that operates in two fundamental steps (Ungurean, Cârstoiu, & Groza, 2016):

- Prediction phase: the system state is estimated using the following equations:

$$\hat{x}_{(t|t-1)} = F_t\hat{x}_{(t-1|t-1)} + B_t u_t \tag{6}$$

$$P_{(t|t-1)} = F_t P_{(t-1|t-1)} + Q_t \tag{7}$$

- Update phase: the algorithm updates the prediction based on the system errors as follow:

$$\hat{x}_{(t|t)} = \hat{x}_{(t|t-1)} + K_t(y_t - H\hat{x}_{(t|t-1)}) \tag{8}$$

$$K_t = P_{(t|t-1)} H_t^T \, (H_t P_{(t|t-1)} H_t^T + R_t)^{-1} \tag{9}$$

$$P_{(t|t)} = P_{(t|t-1)} - K_t H_t P_{(t|t-1)} \tag{10}$$

Where $\hat{x}$ is the estimated state, F is the state transition matrix, B is the control matrix, u is the input vector, P and Q are respectively the state and the process variance matrix, y is the output measurement, K is the Kalman gain, H is the measurement matrix and R is the measurement variance matrix.

Since the Kalman filter is limited for linear systems and has proven a reliable estimation of the states of a process, researchers have developed different extensions of this model-based algorithm to adapt it for nonlinear systems.

The extended Kalman Filter is one of the enhanced versions that deals with non-linear systems such as battery state estimation. This model-based approach was widely used to estimate the battery SOC. The process of estimation starts by choosing the battery model that describes its response in discrete-time, then the algorithm is applied to estimate the state. The state description of the battery model has to include the wanted unknown quantities that should be determined. For this paper, a second-order ECM (Figure 2) will be used to test the performance of this approach to estimate the battery state of charge.

(E. Kamal, 2015) and (Sepasi, Ghorbani, & Liaw, 2014) used the same battery model to estimate the battery SOC. The nonlinear system is first represented using equations 11 and 12, to represent respectively the system dynamics expressed in state equations and the output equation of the system (O., P., S., & Molinas, 2017).

$$x_{k+1} = f(x_k, u_k) + w_k \tag{11}$$

$$y_{k+1} = g(x_k, u_k) + v_k \tag{12}$$

Where $f(x_k, u_k)$ represents the nonlinear transition function, $g(x_k, u_k)$ represents the nonlinear measurement function, w_k and v_k denote respectively the process and the measurement noise.

To apply the EKF $f(x_k, u_k)$ and $g(x_k, u_k)$ are linearized at each time step using the first order of Taylor-series. 13 and equation 14 can be rewritten as follows (E. Kamal, 2015):

$$x_{k+1} = A_k x_k + B_k I_{L,k} + w_k \tag{13}$$

$$y_k = C_k x_k + D_k I_{L,k} + v_k \tag{14}$$

Where $A_k = \left.\frac{\partial f(x_k,u_k)}{\partial x_k}\right|_{x_k,u_k}$, $B_k = \left.\frac{\partial f(x_k,u_k)}{\partial u_k}\right|_{x_k,u_k}$, $C_k = \left.\frac{\partial g(x_k,u_k)}{\partial x_k}\right|_{x_k,u_k}$ and $D_k = \left.\frac{\partial g(x_k,u_k)}{\partial u_k}\right|_{x_k,u_k}$.

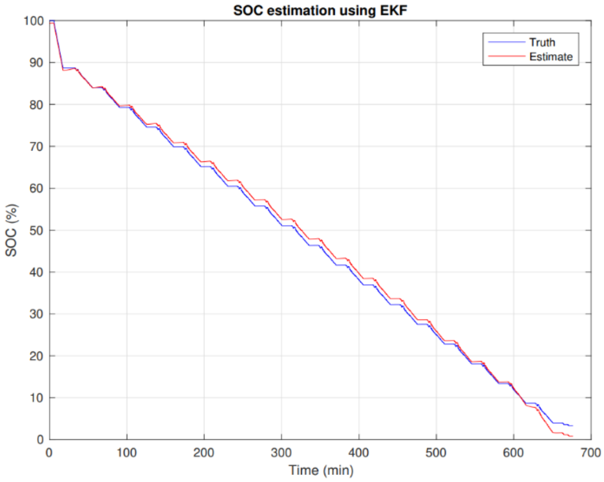


Figure 5 : SOC estimation using EKF

EKF was proven to give accurate results even with noisy input data, the algorithm is much lighter and can be implemented for real-time application. However, when the system is highly non-linear, linearization error would lead to highly unstable filters because of uncertainties in the first order Taylor series (Sun, Hu, Zou, & Li, 2011).

Many authors have used dual extended Kalman filter to provide an online estimation of both battery SOH and SOC. As the name suggested, this approach uses two extended Kalman filters to predict the battery states and update its parameters to give more reliable results. In (Wassiliadisa, et al., 2018) the authors provide an outstanding investigation of the DEKF performance for SOC and SOH estimation under different dynamics and degradation stages. Compared to a simple EKF, the DEKF was proven to improve the accuracy of the SOC over battery lifetime, while the battery internal resistance and capacity become inaccurate with advanced degradation.

3.5. Sliding Mode Observer

Another model-based approach that was used to estimate the internal states of the battery is known as the sliding-mode observer (SMO). This algorithm has the advantage of compensating the modelling errors caused by parameters variation of circuit model and can help overcome some drawbacks that other model-based methods present. Introduced by Emel'yanov (1959) the SMO algorithm was adopted to estimate the battery SOC. A combination of an improved ECM and a SMO were used by different authors to provide an accurate estimation of the battery SOC (Chen, X., Z., & A., 2012) (Nacer, Ahmed, & Naamane, 2012).

In (Ning, Xu, Cao, Wang, & Xu, 2016), the authors used SMO to estimate the battery SOC based on a parameter adaptive battery model to reduce the systematic errors. The estimation result shows a rapid convergence of SOC curve with an estimation error of less than 2%, which reflects the robustness of the algorithm.

3.6. Fuzzy Logic

Fuzzy logic is a useful mathematical concept that allows modelling non-linear and complex systems using the appropriate training dataset. It's a non-monotonic logic that uses crisp sets to categorize the measured data. The relationship between a member of a set and its degree of membership is defined using a membership function. In the case of SOH, the membership function outputs can be set as healthy, tolerable and not healthy.

Jonghoon Kim (2014), used a fuzzy logic-controlled methodology to predict the battery SOH, first the cell resistance and maximum capacity were determined based on voltage, current, temperature and time, then fuzzy logic approach was applied to estimate the battery SOH using the resistance and the maximum capacity values.

Burgos et al. proposed a novel fuzzy logic algorithm to predict the battery SOC. A fuzzy model that characterizes the relationship between the battery open-circuit voltage, SOC and the discharge current was used in combination with an EKF to predict the battery SOC (Burgos, Saez, Orchard, & Cardenas, 2015).

3.7. Least squares

The least square algorithm is a widely used approach that identifies the best fit function that minimizes the sum of quadratic errors between measured output and system response.

In (L, 2011), Gregory L.Pett proposed an enhanced version of the LS algorithm called Weighted Total Least square to estimate the battery capacity. Since the standard least square approach doesn't consider the uncertainties that the input measurement includes, the author used the WTLS algorithm that takes into account the noises of the accumulated ampere-hour measurements and the battery SOC. The estimation results are more accurate than a standard least squares approach, and the algorithm can be used for real-time applications.

4. CONCLUSION

SOC and SOH estimation is of a great importance when developing a battery management system, they provide an overview of the short- and long-term state of the battery.

The main goal of this paper is to provide a basic understanding of the different algorithms, the advantages and shortcomings of each to help build an advanced BMS for EV application. The review shows that there is no perfect approach to estimate the battery states, and the choice should be made based on the complexity of the system and the target application.

The battery model types that are used for EV applications are first presented. Then the most relevant estimation algorithms that were used in literature to predict the battery SOC and SOH are outlined. Certain algorithms were tested using a second order equivalent circuit battery model to test out their accuracy.

5. REFERENCES

A., F., J., A. D., K., P., & Longo S. and Wild, M. (2016). A review on electric vehicle battery modelling: From Lithium-ion toward Lithium–Sulphur. *Renewable and Sustainable Energy Reviews, 56,* 1008-1021.

A., J., B., R., & M., D. M. (2016). Review of simplified Pseudo-two-Dimensional models of lithium-ion batteries. *Journal of Power Sources, 327,* 44–55.

A., L., A., S., Venet, P., A., H., M., O.-B., & S., P. (2016). Practical online estimation of lithium-ion battery apparent series resistance for mild hybrid vehicles. *IEEE Transactions on Vehicular Technology, 65*, 4505–4511.

Berecibar, M., Gandiaga, I., Villarreal, I., Omar, N., Mierlo, J. V., & Bossche., P. V. (2016). Critical review of state of health estimation methods of Li-ion batteries for real applications. *Renewable and Sustainable Energy Reviews, 56* , 572-587.

Burgos, C., Saez, D., Orchard, M. E., & Cardenas, R. (2015). Fuzzy modelling for the state-of-charge estimation of lead-acid batteries. *Journal of Power Sources* , 355-366.

Chang, W.-Y. (2013). The State of Charge Estimating Methods for Battery: A Review. *International Scholarly Research Notices*, 7.

Chen, X., X., S. W., Z., C., & A., K. (2012). Sliding Mode Observer for State of Charge Estimation Based on Battery Equivalent Circuit in Electric Vehicles. *Australian Journal of Electrical and Electronics Engineering*, 225–234.

E. Kamal, A. E. (2015). State of charge estimation based on extended Kalman filter algorithm for Lithium-Ion battery. *23rd Mediterranean Conference on Control and Automation (MED).*

Fan, G., & Canova, X. L. (2017). A Reduced-Order Electrochemical Model of Li-Ion Batteries for Control and Estimation Applications. *IEEE Transactions on Vehicular Technology, 67*(1), 76-91.

Fleischer, C., W., W., Z., B., & D.U., S. (2013). Adaptive on-line state of available power prediction of lithium-ion batteries. *Journal Power Electron, 13,* 516–527.

G., F., X., L., & M., C. (2018). A Reduced-Order Electrochemical Model of Li-Ion Batteries for Control and Estimation Applications. *IEEE Transactions on Vehicular Technology, 67*(1), 76–91.

Kailong, L., Kang, L., Qiao, P., & Cheng, Z. (2019). A brief review on key technologies in the battery management system of electric vehicles. *Frontiers of Mechanical Engineering*, *14*(1), 47-64.

Kempera, P., Li, S. E., & Kum, D. (2015). Simplification of pseudo two dimensional battery model using dynamic profile of lithium concentration. *Journal of Power Sources, 286,* 510-525.

Kim, J. (2014). Fuzzy Logic-Controlled Online State-of-Health (SOH) Prediction in Large Format LiMn2O4 Cell for Energy Storage System (ESS) Applications.

L, P. G. (2011). Recursive approximate weighted total least squares estimation of battery cell total capacity. *Journal of Power Sources*, 2319–2331.

Mathew, M., Janhunen, S., Rashid, M., Long, F., & Fowler, M. (2018). Comparative Analysis of Lithium-Ion Battery Resistance Estimation Techniques for Battery Management Systems. *Energies*, *11*, 1490.

Moura, S. J., Argomedo, F. B., Klein, R., & Krstic, A. M. (2016). Battery State Estimation for a Single Particle Model With Electrolyte Dynamics. *IEEE Transactions on Control Systems Technology, 25*(2), 453–468.

Nacer, M., Ahmed, B., & Naamane, A. (2012). Battery Models for Estimation of State of Charge by Sliding Mode Observer. *Journal of Smart Innovation, Systems and Technologies*, 133-149.

Ning, B., Xu, J., Cao, B., Wang, B., & Xu, G. (2016). A Sliding Mode Observer SOC Estimation Method Based on Parameter Adaptive Battery Model. *Energy Procedia*, 619 – 626.

O., G., P., V., S., J., & Molinas, M. (2017). Battery modeling and Kalman filter-based State-of-Charge estimation for a race car application. *IEEE 14th International Conference on Networking, Sensing and Control (ICNSC).* .

Rincon-Mora, E. C. (2006). Accurate electrical battery model capable of predicting runtime and I-V performance. *IEEE Transactions on Energy Conversion, 21*(2), 504-511.

Sepasi, S., Ghorbani, R., & Liaw, B. Y. (2014). Improved extended Kalman filterfor state of charge estimation of battery pack. *Journal of Power Sources* , 368-376.

Sun, F., Hu, X., Zou, Y., & Li, S. (2011). Adaptive unscented Kalman filtering for state of charge estimation of a lithium-ion battery for electric vehicles. *Energy, 36*, 3531-3540.

Ungurean, L., Cârstoiu, G., & Groza, M. V. (2016). Battery state of health estimation: a structured review of models, methods and commercial devices. *International Journal of Energy Research, 41*(2), 151-181.

Wassiliadisa, N., Adermanna, J., Frericksb, A., Pakb, M., Reitera, C., Lohmannb, B., & Lienkampa., M. (2018). Revisiting the dual extended Kalman filter for battery state-of-charge and state-of-health estimation: A use-case life cycle analysis. *Journal of Energy Storage*, 73-87.

Yu, J., Mo, B., Tang, D., Yang, J., Wan, J., & Liu, J. (2017). Indirect State-of-Health Estimation for Lithium-Ion Batteries under Randomized Use. *Energies*.

COMPONENT APPROACH BASED ON PETRI NETS FOR THE DESIGN OF THE AUTOMATIC CONTROL FOR RAILWAY SYSTEM

Armand Toguyéni

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

armand.toguyeni@centralelille.fr

ABSTRACT

The automation of rail systems is a major challenge for the development of this mode of transport. This automation must affect all the functions of the control system and not just the replacement of train drivers. This study proposes a component approach for modelling control functions based on Colored Petri Nets. This component approach masks the complexity of the system components and their functions from the designers of a rail system. In this work we also propose a new formal verification method based on the construction of a reduced reachability graph of a global model. This approach makes it possible to verify the main properties of the components necessary for their implementation in software libraries that can be used by railway system designers.

Keywords: Discrete Event Systems, Colored Petri Net, formal modelling and verification, Automatic Train Control.

1. INTRODUCTION

The automation of rail systems is a major challenge for the development of this mode of transport with regard its competition with road and air transport. This study proposes to use a component approach to facilitate the design of railway systems. This requires to develop generic component libraries. Assisted by such libraries, a designer can model a rail system by instantiating the generic components of his library and specifying the interactions between these components. This requires to check that each generic component works properly.

This work proposes to use Jensen's Colored Petri Nets (CPN hereafter) for modeling railway systems. They allow to use the modularity and the parametric modeling to build generic components. These generic components can be instantiated to build a global model.
This paper is structured as follows. In the second section, we will present how actual railway systems operate and the main functions of an automatic control system. In section 3, we will propose a decentralized control architecture for the implementation of automatic control of railway systems. In the fourth section, we will present our modeling principles based on the concept of Petri Nets modules, with modules whose operation depends on parameters. In the fifth section we will propose a modeling of some of the components presented in the third section. The sixth part focus on the verification of our generic components. In particular, we will introduce a new semantics of PN interpretation in order to reduce the size of reachability graphs. Section 7 gives a case study to illustrate our approach. We will end with a conclusion and perspectives.

2. STATE OF ART OF AUTOMATIC TRAIN CONTROL

2.1. Description of a railway system and its operation in traditional mode

A railway system can be abstracted as consisting of railway nodes and railway lines. There are two categories of railway nodes: stations and junctions. A rail junction consists of switches end track elements that establish routes for routing trains. Stations have platforms that allow a train to stop for passenger loading/unloading. This is what differentiates them from junctions (Lusby, Larsen, Ehrgott, & Ryan, 2013). Rail lines are used to connect stations together. But several railway lines may cross at junctions allowing trains to move from one line to another to reach a destination station. Lines generally have two one-way tracks to connect two nodes in both directions of traffic (round trip). Normally, all trains on a track move in the same direction of travel. Some lines may have bidirectional track portions. Within a railway node, some track sections can also be bidirectional.

Safety is one of the main criteria for the proper functioning of a railway system. To do this, it is necessary to avoid collisions.
In order to ensure the proper functioning of railway systems, three categories of human operators take decisions and manually trigger control operations: train drivers, line regulators and railway dispatchers.

The role of the train driver is to control the advance of his train by respecting the signaling (lights and speed limits). He has no control over the train's itinerary, which depends on the other two operators. Today, in modern control and signaling systems, the driver operates under the control of an automated train protection system (ATP). This system is used to guarantee safety, especially on the lines. It can trigger an emergency stop of the train if necessary. The driver performs other functions such as opening and closing train doors.

The function of the line regulator is to regulate the traffic of the trains on the line it controls. To do this, it can switch a slow train onto a bypass track to allow a faster train to pass it. The slow train then returns to the main track as soon as possible.

The dispatcher role is to decide which trains pass through the node he controls. It establishes a route for each train. It can be assisted by a computer for assigning a route to a train. By default the signal at the node input is closed (red light). When the signal is open (green light), the train can enter the node (Lusby, Larsen, Ehrgott, & Ryan, 2013).

2.2. Automatic train control systems

Current rail systems are not very automated but there are many automated metro lines in the world because they are simpler. The main automatic control system (ATC) used by metros are Communication-Based Train Control also called CBCT (IEEE Std 14741, 2004). One of the main features of this system is the use of radio communication between trains and ground infrastructure. This system has inspired ERTMS Level 2 which is based on the GSMR communication system (European Railway Agency, 2016). This study assumes the use of such type of communication.

An automatic train control system is actually divided into three subsystems (Yin, et al., 2017): the train operating system (ATO), the automatic train protection system (ATP) and the train supervision system (ATS).

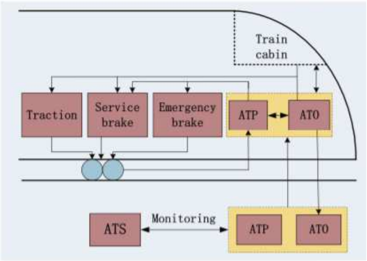


Figure 1: ATC structure (Yin, et al., 2017)

The function of the ATO is to replace the driver on board trains. This is the function to be developed as part of the autonomous train. It is responsible for controlling the advance (traction) and stopping of trains (braking) according to its operating modes and speed limits. It is a function that is both on board and on the ground (Figure 1). Indeed, the trackside controllers will calculate the train's movement authorizations according to the limitations due to its location in the network but also according to the position of the other trains in the network.

The role of the ATP function is to monitor the execution of train operations. In manual driving mode, it controls the commands given by the driver to the train. In automatic mode it controls the orders of the ATO. The ATP will directly control the emergency braking. It is also a function that is both on board the train and on the ground (Figure 1). The majority of current rail control and signaling systems are based on the blocks' technique. This technique guarantees safety on the tracks of a line. Indeed, each track is divided into electrically isolated blocks. Safety requires that there be only one train per block. In actual systems, the safety implemented by ATP is based on the concept of interlocking. Also, there are many studies on the verification of interlocking by formal methods including CPN (Vanit-Anunchai, S., 2014). ATP is a critical function from safety point of view.

The ATS function is a function of the ground system (Figure 1). It monitors that the train movements are in accordance with the planned scheduling. It is also responsible for the dynamic routing of the train for crossing railway nodes.

2.3. Conclusion

More specifically, this work is a contribution to the implementation of sub-functions of the ATO (train movement authorization) and ATS systems. For this, we will be inspired by ERTMS/ETCS level 2 which is in fact an ATP function for protecting the movement of a train on a railway track (European Railway Agency, 2016). In next sections, we propose a new architecture and a methodology to develop automatic control system.

3. DECENTRALIZED ARCHITECTURE FOR THE CONTROL OF A RAILWAY SYSTEM

This study concerns the management of multiple trains in a railway network with a full automation of the system. To propose a new architecture, we are going first to propose a structural and a functional decomposition of a railway network. A top-down approach is used to conduct this decomposition for a railway system by first considering separately the two points of view and then by mapping together the elementary components of the system with their functions.

3.1. Structural decomposition

A railway network is a complex system. Its control is distributed in several components of the ground infrastructure and onboard the trains. In order to be able to model such a system, a structural decomposition is necessary. Figure 2 summarizes the proposed decomposition.

It is important to notice that the structure of each subsystem (stations, junctions and railway tracks) is different depending on the railway network requirements, but the basic components (block, switch, track section, balise and track circuit) used to implement them are generic.

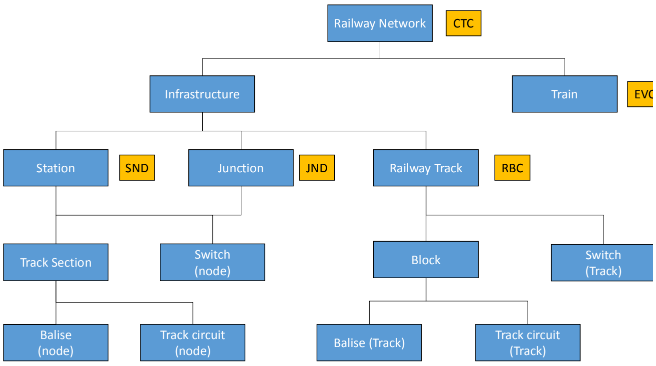


Figure 2: Structuration Decomposition of a Railway Network

3.2. Functional Decomposition

Because this study aims to propose a methodology for automatic control, the structural decomposition also shows the different controls centers that allow implementing a decentralized control of a railway system.

At the top of the hierarchy, there is the Centralized Traffic Control (CTC hereafter). The CTC checks in real time that the planned transport plan is implemented throughout the network. It has a global but macroscopic view of rail traffic in the system it controls. Thus, it defines and updates the itinerary of each train. Each itinerary is defined in terms of departure station, arrival station (final destination), intermediate nodes through which the train will pass. The arrival dates of a train at each node of its itinerary are defined by the planned transport plan. The CTC regularly receives feedback from the lower-level control centers that execute locally this plan. It calculates the differences between the executed and the planned plan and sends back to local centers, updated local transport plans. It implements rail traffic supervision that is an ATS function.

The types and number of local controllers are consistent with the breakdown of the infrastructure into railway nodes and lines composed of tracks. A local control center is associated with each railway node. Each junction is controlled by the JTC (Junction Traffic Controller) whose main function is to implement automatic train routing (an ATS function) within the node. Taking into account the local planned transport plan (list of trains to cross the junction in a time slot), the JTC allocates in real time the resources necessary for each train arriving at the node to set its route. It is based on a planned scheduling of traffic to cross the junction in accordance with the local transport plan received from the CTC. In case of fault, the local transport plan is updated. If this update does not absorb all the disruptions, the CTC is informed so that more global actions can be taken to find a solution. Trains can be slowed down, accelerated or even their itinerary modified. The JTC is also responsible to set up the train route (it is an ATO function) and to give to each train its movement authorization inside the junction (it is an ATP function).

The stations are controlled by the STCs (Station Traffic Controller). STCs implement automatic train routing like JTCs. In addition, they must manage the assignment of platforms to trains that stop at the station.

The Radio Bloc Center (RBC) operates in the same way as in ERTMS/ETCS level 2 (European Railway Agency, 2016). Each train regularly sends its position to it. Taking into account the position of each train on the track and their time constraints, the RBC calculates their respective movement authorizations (authorized travel distance and speed profile) and regulates the traffic of the trains on the line. Movement authorizations are transmitted to trains in response to their requests for movement authorization.

The fourth local control center is onboard in each train. It is implemented by the main computer of the train called EVC (European Vital Computer). It has a role similar to that defined in the standard (European Railway Agency, 2016). It implements both the ATP and ATO functions of the train. The role of this computer is to control the train's advance (an ATO function) by respecting the movement authorizations (an ATP function) transmitted by the RBC when it is on a track. When it must cross a node, it communicates with the controller of the node (JTC or STC) who gives it his movement authorizations in accordance with the route it has assigned.

3.3. Mapping Relations

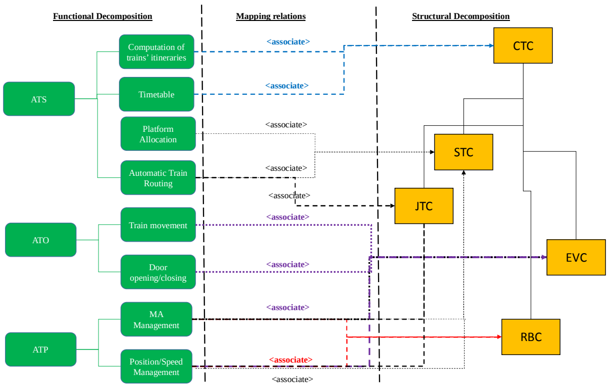


Figure 3: Mapping Relations from Control Viewpoint

Figure 3 depicts the relations mapping between the control components of a railway network and the different functions hold by each of them. On the left of the figure, the green nodes represent the functional decomposition of a Railway Network from control viewpoint. On the right of Figure 3, the yellow square model the main control components. As an example, one can see that STC is associate with platform allocation and automatic train routing that are ATS sub-functions and is also associate with MA Management and Position/Speed Management that are ATP sub-functions.

In the rest of the paper, we will focus on the design of control functions of JTC, RBC and EVC components.

4. MODELING OF DISCRETE CONTROL COMPONENTS

The objective of this part is to show how to model components in order to allow the modelling of a rail control system. We focus here on the train EVC, RBC for line control and STC for junction control.

4.1. Colored Petri Nets

4.1.1. Introduction

For DES, there are three types of formal models (Cassandras and Lafortune, 2008): regular languages, automata and Petri Nets (PN hereafter). PN were defined in 1962 in the thesis of the German mathematician Carl Adam Petri. He then showed that it was the best formalism to model DES characterized by several subsystems evolving in parallel and sometimes having to be synchronized. The initial formalism is called, Places/Transitions Petri nets (notation P/T-nets). But since that time, many abbreviations or extensions of PN have been proposed by other authors. Colored Petri Nets (CP-nets hereafter) are basically an abbreviation for P/T-nets. It means that any colored model can be unfolded to find the equivalent P/T-nets model. Coloration consists of defining sets of objects. Thus, the model's tokens can model real-world objects. With P/T-nets, it is necessary to make a specific net for each object. CP-nets make it possible to factorize the behavior common to several objects into a single model of the same size as the model of an object in the case of P/T-nets. Thus, they make it possible to gain in concision and thus to reduce the size of the model representing the set of physical objects.

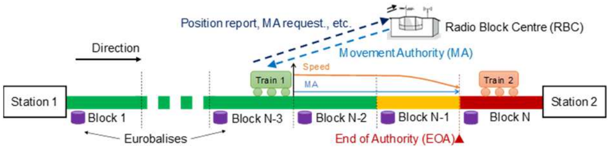


Figure 4: Example of two trains on a railway track

To illustrate the power of expression of Colored Petri Nets, consider the example given in Figure 4. This example defines two trains moving on a railway track composed of N blocks. N is a parameter that represents the length of the track in number of blocks. Figure 5 illustrates the modelling of this system in P/T-nets. In this case, in order to distinguish between the two trains, a model must be constructed for each train. The overall model of the system is therefore composed of 2N places and 2N transitions. The corresponding automaton would be composed of N^2 states illustrating the combinatorial explosion of this formalism.

Figure 6 illustrates the same problem modeled in CPN which is an extension of CP-Nets (Jensen et al., 2007). In this case, the problem parameters can be specified by constants (NTr for the number of trains and N for the number of blocks).

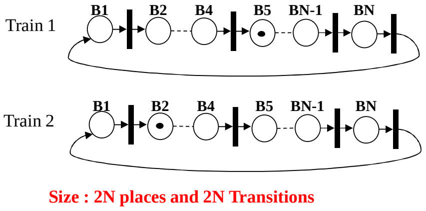
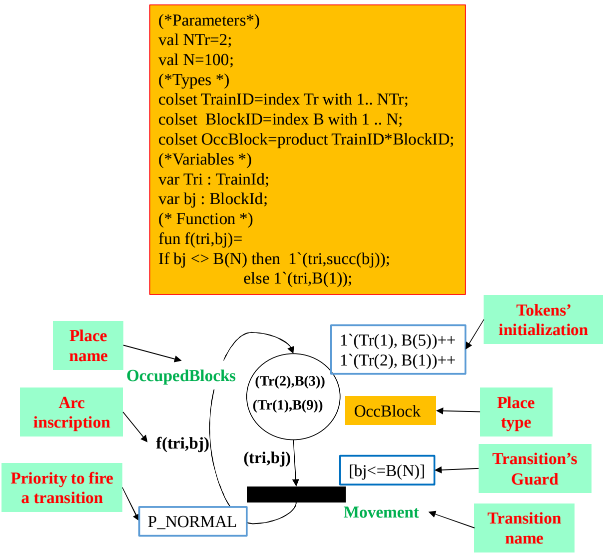


Figure 5: P/T-nets of the two trains on a track

This allows to define train identifiers (*Tr(1)* is the train identifier of train 1 and *Tr(2)* is the identifier of train 2) using the *ML language* of CPN Tools. Similarly, we can specify the block identifiers on which the trains are positioned *(B(2)* and *B(5)* respectively). CPN Tools' ML language allows to define composite types such as *OccBlock* which specifies the blocks occupied by a train. The definition of these types allows to fold the P/T-nets of Figure 5 and to obtain the CP-nets of Figure 6 composed of 1 place and 1 transition. It can be seen that this model is much more compact than the one obtained with P/T-nets or finite state automaton. However, it requires the creation of functions such as function *f*, which reflects the advance of each train on the rail track. But there are several types of CP-nets. In particular, this study is based on Karl Jensen's Colored Petri Nets (CPN hereafter (Jensen et al., 2007). They are not pure CP-nets. Actually, they become a High Level Petri Nets (Jensen and Rozenberg, 2012).



```
(*Parameters*)
val NTr=2;
val N=100;
(*Types *)
colset TrainID=index Tr with 1.. NTr;
colset  BlockID=index B with 1 .. N;
colset OccBlock=product TrainID*BlockID;
(*Variables *)
var Tri : TrainId;
var bj : BlockId;
(* Function *)
fun f(tri,bj)=
If bj <> B(N) then  1`(tri,succ(bj));
              else 1`(tri,B(1));
```

Figure 6: CP-nets of the two trains on a track

In CPN model, each place has a type such as *OccBlock* that defines the type of the place *OccupedBlocks*. It is initialized with two tokens that are each a 2-uple. Based on the operator of multiset ('++' is the addition operator for a multiset), if *M* represents the marking function, one can write the marking of *OccupedBlocks* by the equation (1).

$$M(\mathrm{OccupedBlocks}) = 1\text{`}(\mathrm{Tr}(1), \mathrm{B}(5)\ ) + \\ +1\text{`}(\mathrm{Tr}(2), \mathrm{B}(2)) \qquad (1)$$

$1\text{`}(\mathrm{Tr}(1), \mathrm{B}(5)\ )$ is a token composed of 2 elementary colors: *Tr(1)* is the identifier of train 1 and *B(5)* defines the block that it occupies.

4.1.2. Main features

The choice of CPN is also justified by the existence of CPN Tools, a suite of tools for editing and analyzing them. It makes it possible to develop a model in a modular and hierarchical way (see section 4.3). CPN

Tools also offers designers the ability to analyze the properties of their models. The analyses are based on the generation of the reachability graph of the CPN model called occurrence graph by the authors (Jensen and Rozenberg, 2012). Its generation makes it possible to obtain a report on the usual properties of a PN such as boundedness or liveness. This feature will be used in section 6. CPN Tools has also a model-checker called ASK-CTL that allows to analyze specific properties of a model (Christensen and Mortensen 1996). It is out the scope of this study.

4.2. Component modelling principles

The objective of our modeling is to build generic components that can be placed in a library after checking their properties. These components can then be used to build global models of a railway system by instantiating these generic components. For that, each component is defined as a CPN module interface places to allow it communicating with its environment. Because the global model is distributed in different computers, the components communicate by the semaphore technique (Murata, 1989). The orientation of the arc linking an interface place to a module indicates the nature of the semaphore. So when a module (component) requests a service from another component, it will use a semaphore request. The requesting component then acts as a client and the receiving component as a server component. When the server component responds to the client component, it will use an acknowledge type semaphore.

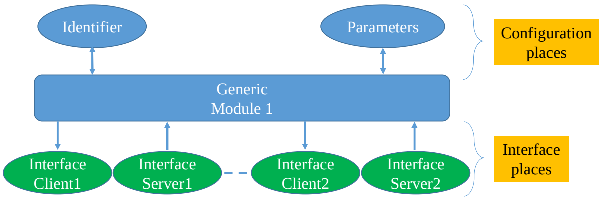


Figure 7: Generic component architecture

In order to be able "to instantiate" the generic components to implement a global model, let us use configuration places of each module. A configuration place is used to specify the identifier of each component at the time of instantiation (copy in the global model) of each generic module. One or more other places will make it possible to define the specific parameters of each component (Figure 7). It can be noted that the arcs linking the configuration places to the module are bidirectional. This means that the tokens of these places are read by the module and then rewritten in the place for later use.

Figure 8 shows how to build a global model from generic components. This global model uses two instances of module 1 and one instance of module 2. The configuration places are initialized to define the identifier and parameters of each module when building the system model. It is noted that the identifier of each instance is unique even if they are instances of the same module. The parameters of two instances of the same module can be identical. Thus, P1.1 and P1.2 can contain the same values. It is noted in this construction that the communication interfaces between two types of modules are merged. Thus it is the same place M1toM2 that models the communications of the instances of module 1 to the instances of module 2. This choice is made to illustrate the fact that in modern architectures communications are based on media operating in broadcast mode. Thus, the exchanged messages must contain the sender's identifier and the recipient's identifier so that the latter can recognize the messages sent to him and can respond to the sender.

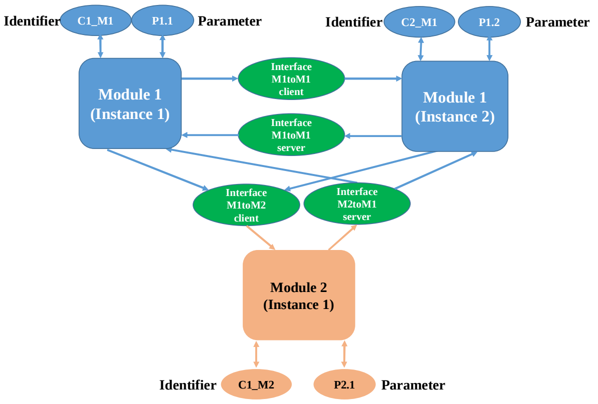


Figure 8: System architecture built from generic component instances

4.3. Implementation of modelling principles in CPN Tools

To implement our modeling principles, CPN Tools offers 3 specific services: the ability to define component identifiers based on index color set ('*colset*' type constructor in CPN's syntax), module definition through substitution transitions, and places' integration through socket/port or fusion places (Jensen and Rozenberg, 2012; Jensen and al., 2007).

An *index colorset* are sequences of values defined by an *identifier-name* and an *index specifier*. Other parameters are represented by their convenient types and can be merged into a *record* or *product* colset to have a compact representation. Figure 9 shows an example of the declaration of the parameters of a train component (EVC). Among these parameters, colorset *TRAINNO* is defined as an *index colset* and is a type used to define different identifiers of trains. The examples of its values could be *Train(1)*, *Train(2)… Train(10)*. A *record colset TrainAttribute* assigns other parameters to an instance of the train component, such as train type, train mass, its origin and destination stations.

```
val maxtrain=10;
colset TRAINNO = index Train with 0..maxtrain;
var tno: TRAINNO;
colset TrainType = with Passenger | freight;
colset TrainMass = int with 0..100000;
colset StationName = string;
colset TrainAttribute = record tType:TrainType * tMass:TrainMass * tOrigin:StationName * tDesti:StationName;
```

Figure 9: Declarations for configuration places of component train (part of)

The notion of substitution transitions makes it possible to distinguish two modeling layers of a system under CPN Tools: the global layer (upper level) and the component layer (lower level). The *global layer* mainly defines the configuration of the instances of the components and illustrates the connections between them, as shown in Figure 11. This global layer is also modeled as a colored Petri net model. In CPN Tools, the module body of each instance is represented by a substitution transition (rectangles with double-line borders in Figure 10) in this global layer.

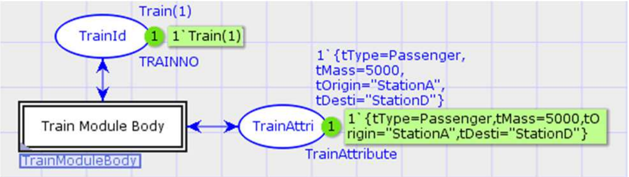


Figure 10: Example of configuration places of train component

The hierarchical feature supported in CPN Tools offers a possibility of implementing the interfaces between different component models or their instances. Figure 11 shows an example of system modeled with two layers: the *global layer* and the *component layer*. Two components “Train” and “RBC” are modeled using the component modeling method of the parametric module representation. In the example, one instance of each component is used to build a global model. Transitions “*Train*” and “*RBC*” are *substitution transitions* in CPN Tools and their details are modeled in the module body of the corresponding component modules. Places “*Train1*” and “*RBC1*” are configuration places used to assign different identifiers and parameters to these instances the corresponding generic components. Place “*T2RBC*” is an interface place which is used send information from a train instance to an RBC instance.

An interface modeled by CPN hierarchy is implemented by *port/socket assignments*, which are used to merge places on the two layers. Such a place on the *lower layer* (component layer) is called a *port*, and that on the *higher layer* (global layer) is called a *socket*. A *port* is always associated with a *port-type tag* (the blue tags in Figure 11) and can be one of the three kinds according to the direction: tag “In” for “*input*”; tag “Out” for “*output*” and tag “In/Out” for both the two directions.

A *socket* is an input place or an output place of a substitution transition, i.e. there is always at least one arc between a substitution transition and a socket.

By using the port/socket assignments, a component module can be “glued together” with the surroundings of its corresponding substitution transition. Each socket must be assigned to a port on the corresponding subpage. A port with a tag “In” must be assigned to a socket which is an input place of the substitution transition. Analogously, an “Out” port requires a socket which is used an output place of the substitution transition. In the example, the “In/Out” ports are used by the configuration places because that the identifiers and parameters in these places are normally to be referred to, other than to be generated nor to be consumed.

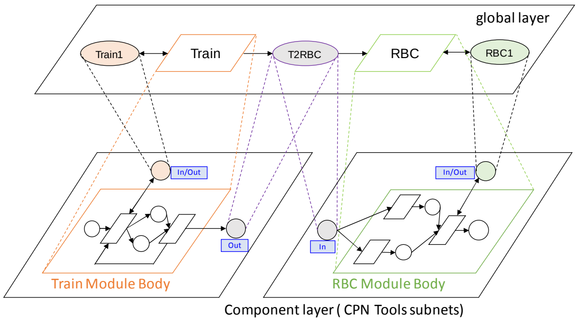


Figure 11: Modelling of interfaces by CPN Tools hierarchy

In order to distinguish the different instances of the same component as a concrete sender or receiver, a colored token to be used in the interface places need to be defined as a *product colset* as given by equation (2):

$$(\text{Sender Identifier}, \text{Receiver Identifier}, \text{Message}) \quad (2)$$

The “*Message*” in (2) can also be a *product colset*, which is usually composed of a “*MessageType*” filed and one or more value(s). An example of a position report sent by a train instance to an RBC instance to update its position could be:

$$(\text{Train }(1), \text{RBC }(1), (\text{UPDATE}, \text{Train}(1), 10)) \quad (3)$$

In this message, “*UPDATE*” is the message type, followed by the values “*Train (1), 10*” that means *Train(1)* is located on block *10*.

5. MODELLING OF GENERIC COMPONENTS

In this section, we will propose models of the three generic components used by our actual global model of a railway system: EVC, RBC and JTC.

5.1. Modelling of the EVC component

As indicated in paragraph 3.2.3, the main functions of a train are the management of its position, the management of its movement authorizations and the management of its movement. The train's itinerary in the rail system is defined by the data of its departure and destination stations. Also, in its departure station, the train waits to receive a departure authorization from the STC. Following this authorization, it requests the STC for a route to go out of the station. The established route is notified to it by the STC. The train will then travel through the track sections to reach the first block of line at the station exit. In parallel, the station has pre-registered the train with the RBC managing the line the train will use (see the modeling of JTC in section 0 and the case study in section 7). A train departure operation is modeled by the CPN of Figure 12. The red part of the figure models the crossing of the last section of track before reaching the railway line. When the train enters the first block of the railway line, this is confirmed by the signal sent by the block's balise (see place *Balise2EVC* in the upper left part of Figure 12). It then sends his position to the RBC to confirm his arrival on the line (see place *EVC2RBC* in in the upper right part of Figure 12). From that moment, it is registered by the RBC, which then sends it its first movement authorization (MA). This MA allows him to retrieve the last block number (End of Authorization or EOA) of the line it is authorized to join given the occupation of the line by other trains. The place *Balise2EVC* models the train's communication interface with its environment and in particular the reception of notifications of the train's position by the balises of the infrastructure. These notifications distinguish between line balise messages that are processed by the blue part and node balise messages that are processed by the purple part. The green part models the treatment of MA by ECV. It is noted that for reasons of simplicity, this model does not integrate the train's operating modes.

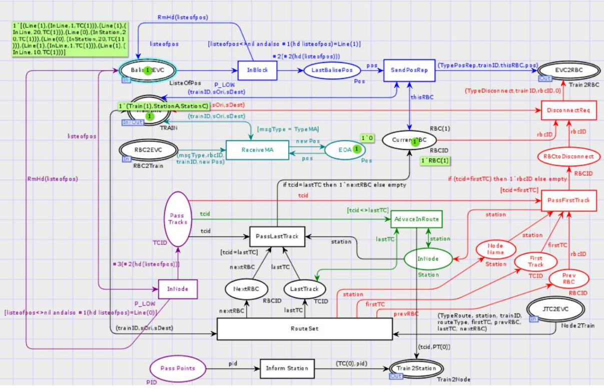


Figure 12: Model of EVC

5.2. Modeling of the RBC component

An RBC manages a line between two nodes (see Figure 4). Its main function is to regulate train traffic by giving them movement authorizations according to the location of each train on the line. Each train arriving on the line must be registered with the RBC for it to take into account its requests for movement authorization. This registration is done in two steps. The first step is a pre-registration by the rail node that the train leaves before it even crosses it (a token is placed in the *STC2RBC* place and arrives in *WaitingTrains* after crossing the *PreRegistration* transition – see Figure 13). As soon as the train arrives on the first block of the track, it sends its position to the RBC to confirm its registration (A token is placed in the *EVC2RBC* place and the *TrainPosition* place after firing of the *PositionReport* transition - see Figure 13).

For the sake of simplicity, each sending of a position is interpreted as a request for authorization of movement. Also when crossing the *PositionReport* transition, a token is placed in the *MAReq* place, which is then processed by the RBC based on the list of trains modeled by the *Managed Trains List* place (see Figure 13). It is important to note the difference in priority between transitions *Update Position* (priority P_HIGH) and *CreatMA* (priority P_HIGH+1). This difference ensures that the position update is performed before the movement authorization is managed. Indeed, in the case of simultaneous validation, the transition with the highest case of same priority, the firing is random based on token semantic.

When the train arrives on the final block of the line, it makes a route request to the controller of the reached node (STC for a station or JTC for a junction). This sub-function is modelled by the green part of Figure 13. When the train is connected with another controller, and has left the line, it sends a disconnect request to the RBC that is processed by crossing the *Disconnect* transition, which removes the train's record in the *Managed Trains List* place (part in red in Figure 13).

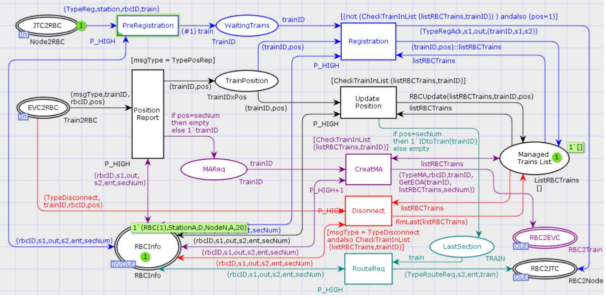


Figure 13: RBC Model

5.3. Modeling of the JTC component

In this section, we are interested in the modeling of the controller of a junction node. This controller is responsible for the dynamic routing of trains crossing the node. Given the train's destination station, the JTC searches in its base for a route with all resources (switches and track sections) available. For this purpose, the potential crossing routes are predefined. When the controller receives a route request from the train, it assigns an available route with respect to the arrival line at the node and the destination station. Once the route has been allocated, the route is established by specific switch controllers. These controllers are not modelled in this study and are considered part of the environment. At the end the JTC notifies the train of the route availability (place *JTC2EVC*) and notifies the RBC (place *JTC2RBC*) of the destination line in order to trigger the pre-registration of the train. All of these operations are modelled in Figure 14.

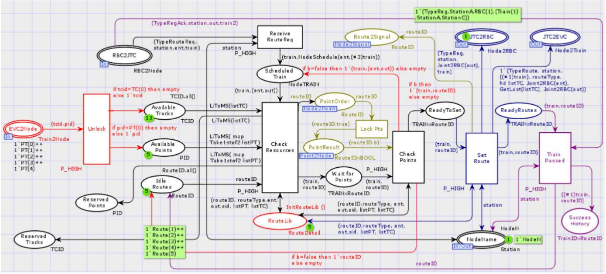


Figure 14: Model of the junction component

6. CHECKING MODEL PROPERTIES

In order to add generic components in the library, it is first necessary to check them. The necessary verifications are primarily the good engineering properties of PN models: liveness, boundedness, no deadlock. But any formal verification under CPN tools requires the prior construction of the reachability graph (also called occurrence graph) of the system in question. The difficulty in verifying these properties is related to the modular approach used for the design. Each component being a module with interface places, it can only evolve thanks to the interaction with its environment. There are several modular verification techniques available. The simplest technique consists in placing the expected answers in the interface places of the server components. This method is simple but does not offer the necessary quality guarantees. Indeed, it can also require to initialize internal places of the model with values according to what we want to verify. Since there are several interface places, it is necessary to anticipate all the tokens necessary for the evolution of the model. Unfortunately, in this case, this can lead to the simultaneous validation of several transitions. As the token game semantics of PN interpretation leads to a random firing of the model transitions, this can lead to test scenarios that do not conform to the design logic of the component in relation to its use in a railway application.

Another method is the calculation of colored invariants. But this method is complicated in the general case of CP-nets and even more High-Level Petri Nets. CP-nets classes such as Well-formed PN have characteristics that allow colored invariants to be calculated. But it is more difficult to model complex systems with this type of PN, which imposes restrictions on the colors and functions used in system modeling (Xie et al., 2017).

A third method is to use the compositional verification technique. For component modeling, this technique consists of constructing a reduced model of the component environment. Although interesting, the challenge is to build scale models that are compatible with the test scenarios. If some models are too small the verification may not be complete.

To address these difficulties, we have developed a new form of modular verification: reactive modular verification. This form of verification is based on work done on reactive nets (Eshuis and Dehnert, 2003). Our objective is to reduce the size of the reachability graph of a component and its environment. For that, one distinguishes the semantics of interpretation of the component to be checked from that of its environment. The principle is to keep the models of the other components as they are but to interpret them with reactive semantics. The component to be checked keeps the token game semantics. This allows the environment to react by giving quickly acknowledges to the requests made to it by the component. Figure 15 illustrates the application of state space construction with the use of both semantics. In this illustration, we have given process A (that represents the component to be checked), a token game semantics and process B (that represents the environment of component A) a reactive semantics. We can see the resulting reachability graph corresponding to part in red on the Figure 15. With the reactive semantics applied to process B, the interleaving of the firings of the transitions of the two processes is eliminated. The black part of the reachability graph is therefore deleted with the reactive semantics applied to process B. This allows reducing the state space without having to abstract the environment model in order to reduce the global model (of the component to be checked and its environment) and then its state space. As a result, considering the state space corresponding to the red part of Figure 15, one can conclude that process A is live, bounded and reversible.

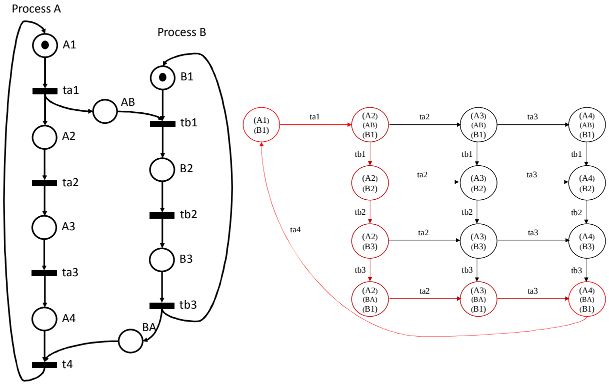


Figure 15: Principle of reactive modular verification

To implement this differentiated semantics in CPN Tools, it is sufficient to use different priorities between the component to be checked (default normal priority - P_NORMAL) and its environment. For example, to check the train model, we gave maximum priority (P_HIGH) to the RBC and JTC components (compare priority of transitions of EVC model in Figure 12 and RBC model in Figure 13). The priority P_INSTATION is less than the priority P_HIGH but greater that the priority P_LOW.

7. ILLUSTRATIVE CASE STUDY

In this section, we will illustrate how to use the previous generic components to model a railway system. To do this, we will use as a case study the example presented by Figure 16.

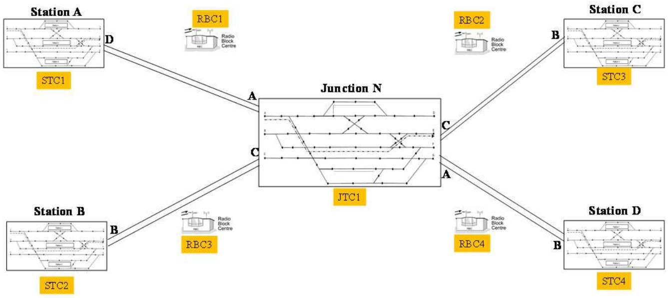


Figure 16: Example of railway network

It has 4 stations named station A to station D and a junction for the interconnection of the lines leading to these stations. Let us suppose that two trains are initially located in station A and they must go to station C at different times. To reach station C, they must go through junction N. The exit door from station A to go to N is door D. Station A and junction N are connected by Line 1 managed by RBC1. To reach Station C, trains must exit through door C and take Line 3 controlled by RBC2.

7.1. The global model

Following the decomposition of Figure 3, one abstracts the railway control system as composed of two EVC to model the trains (EVC1 and EVC2), one RBC for line 1 (RBC1) and another one fore line 3 (RBC2) and JTC1 to model the controller of junction N.

In the global layer (Figure 17), each instance is modeled as a substitution transition and is parametrized by its configuration place. As an example, the configuration place of EVC1 instance is called *T1info* and the configuration place of EVC2 is called *T2info*. In order to have a more compact representation, a unique configuration place with a compound colored token in a form of (4) to represent both the train identifier and the necessary train attributes is used.

$$(\text{Identifier}, \text{Attribute 1}, \ldots, \text{Attribute n}) \quad (4)$$

As an example, the initial marking of *T1info* is:
$M_0(T1info) = 1`(Train(1), StationA, StationC)(5)$

Train(1) is the identifier of the first train. *StationA* and *StationC* define respectively the departure and the destination station of *Train(1)*. In, the same way is defined the parameters of each RBC.

$$M_0(RBC1Info) = 1`(RBC(1), StationA, D, NodeN, A, 20) \quad (6)$$

Equation (6) means that *RBC(1)* controls the line from *StationA gate D* to junction *N gate A*. This line is composed of 20 blocks. This last parameter is essential for the RBC to know when a train has reach the end of the line that it controls in order to request to the JTC a route for the train (corresponds to the firing of transition *RouteReq* in RBC model in Figure 13) .

The interface places enable to model the communication from one type of components to another type of components. For example, in Figure 17 the place *EVC2RBC* models the communications from the EVC instances to the RBC instance. In this case study, it is not possible for EVC to request itself a route to the junction N. The structure of colored tokens put in the interface places enables to define the concrete participants of the communication. As an example, it is worth to notice the initialization of place *JTC2RBC* in Figure 17.

$$M_0(JTC2RBC) = 1`(TypeReg, StationA, RBC(1), (Train(1), StationA, StationC)) \quad (7)$$

This initialization enables us to simulate the request of *StationA* that requests *RBC(1)* to perform a preregistration of *Train(1)* that is arriving on the line controlled by *RBC(1)*. It also gives to *RBC(1)* all the data of the itinerary of *Train(1)*. One can also notice that the initial marking of *T1BM* is a list of data that are like *(Line(1), (Inline,1,TC(1))* or *(Line(0), (InStation, 20, TC(1))*. *Line(1)* means that it defines a train position on Line(1). As an example *(Line(1), (Inline,1, TC(1))* means that the train is on the block 1 of the Line(1) and will reach NodeN by its track section TC(1). *(Line(0), (InStation, 20, TC(1))* means that *Train(1)* is in a node (station or junction) on the track section *TC(1)* and is arrived by the block 20 of the line before the node. So the list allows modeling the sequence of the train positions for simulation or the construction of its reachability graph. The global layer model can be easily modified by connecting/disconnecting components to the corresponding interfaces.

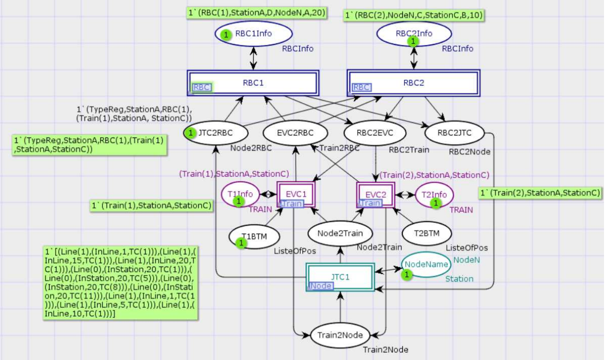


Figure 17: An example of global model based on instantiation of generic components.

7.2. Reachability graph construction for formal verification of a generic component

In order to perform properties' verification, it is necessary to build the reachability graph of the component to be checked with its environment. Considering the global PN model of section 7.1, let us suppose that one wants to check EVC component (more precisely EVC1). To apply the reactive modular verification, let us modified the priority of all the transitions of RBC1, RBC2 and Node N, from P_NORMAL to P_HIGH (a way to perform reactive semantics). The priority of transitions of the EVC components remain at P_NORMAL (a way to keep the token game semantics) except the transitions *InBlock* and *InNode* that keep the priority P_LOW.

To give an idea of the complexity of this verification, we have constructed the reachability graph of the global model of our case study by just considering the evolution of one train (*Train(1)* corresponding to EVC1) going from station A to station C. This corresponds to the initial marking in Figure 17. One can note that the *T2BTM* place that models *Train(2)* positions is not marked indicating that it remains immobilized.

Table 1 shows the results of this evaluation in terms of size and duration. They show the interest of building reachability graphs using a mix of the two semantics (reactive semantics for the environment). These results have been obtained on an Intel CORE i7-5600U CPU at 2,60 GHz, with 16 GO of RAM.

Table 1: Comparison of the size of the reachability graphs constructed with the two semantics

	Reachability graph			
Type of semantics	**Number of nodes**	**Number of arcs**	**Number of dead markings**	**Duration (in seconds)**
Token semantics	5414	15042	35	13
Mix of the two semantics	38	37	1	1

As indicated by the fact that the number of arcs is equal to the number of states minus one, in the case of reactive semantics, the reachability graph is linear. The last state is blocking, modeling the arrival at station C. With the same initial marking and the use of token semantics for all the global model, the reachability graph is much larger. One can especially note that it has three times more arcs than nodes indicating the strong interleaving of the firing of transitions due to this semantics. This example illustrates the great efficiency of the construction of reachability graphs using reactive semantics for the environment.

The obtained reachability graph shows that our model is correct since the train has effectively followed its itinerary. In the obtained scenario, all the transitions of the EVC have been fired at least once (once for the transitions corresponding to the travel inside the node N, and several times for the others), proving that the global models is L1-live (Murata, 1989). In fact the modification of the global model with a transition that resumes the initial marking of this case study from de dead-marking obtained when *Train(1)* reaches station C, one shows that the EVC model is live and bounded.

Applying this approach to each generic component, proves that all our generic components are live and bounded.

8. CONCLUSIONS AND PERSPECTIVES

This work made it possible to propose a decentralized control architecture for the complete automation of rail system control. The objective is to go far beyond the automation of train operation as envisaged by the autonomous train concept. It showed that it was possible to develop a system modeling approach based on generic components placed in a library. The developed components correspond to our points of view that will be refined through standardization. They do not integrate all the complexity of modelling such systems. For example, we did not take into account the modeling of a train's operating modes, which include dozens of modes. We have not modelled train speed regulation that would require the use of a formalism such as hybrid PN (David and Alla, 2005). In this context, the question will arise of having a formalism that allows the characteristics of the two types of PN to be integrated.

The use of reactive semantics is an important way to reduce the combinatorial explosion of reachability graphs. The proposed approach allows proving the correctness of each generic component building a kind of global model with a distinction of the component to be checked and its environment. However, in order to check some properties such as the absence of possibility of collisions between two trains, it is necessary to develop another approach of verification that takes into account a more global model with several instances of trains.

REFERENCES

Christos G. Cassandras, & Lafortune, S., 2008. *Introduction to discrete event systems*. NewYork (USA), Springer.

Christensen, S., & Mortensen, K. H. Design/CPN ASK-CTL Manual, version 0.9, 1996. *URL http://cpntools.org/_media/documentation/askctl manual. pdf.[checked on 2013-12-16]*.

David, R., & Alla, H, 2005. *Discrete, continuous, and hybrid Petri nets* (Vol. 1). Berlin: Springer.

Eshuis R, Dehnert J. 2003. Reactive Petri Nets for Workflow Modeling. *Appl Theory Petri Nets 2003*, vol. 2679, pp. 295–314.

European Railway Agency. 2016. ERTMS/ETCS System Requirements Specification (SUBSET-026) v3.6.0

Jensen, K., & Rozenberg, G, 2012. *High-level Petri nets: theory and application*. Springer Science & Business Media.

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.

Lusby, R. M., Larsen, J., Ehrgott, M., & Ryan, D. M. (2013). A set packing inspired method for real-time junction train routing. *Computers & Operations Research*, 40(3), pp. 713-724.

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

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.

Xie, Y., Khlif-bouassida, M. & Toguyeni, A., 2017. Well-formed Petri Net Based Patterns for Modeling Logic Controllers for Autonomous Trains. Proc. of IMAACA2017, Barcelona (Spain), pp. 25–34

Yin, J., Tang, T., Yang, L., Xun, J., Huang, Y., & Gao, Z., 2017. Research and development of automatic train operation for railway transportation systems: A survey. *Transportation Research Part C: Emerging Technologies*, 85, 548-572.

IDA-PBC CONTROLLER FOR DC-DC POWER ELECTRONIC CONVERTERS WITH NONLINEAR LOAD ENHANCED WITH ADDITIVE DISTURBANCE ESTIMATION

Juan Tomassini[a], Sergio Junco[b], Alejandro Donaire[c]

[a] CONICET, Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina.
[a],[b] LAC, Laboratorio de Automatización y Control, Departamento de Control, Escuela de Ingeniería Electrónica,
Facultad de Ciencias Exactas e Ingeniería, Universidad Nacional de Rosario, Argentina.
[c] University of Newcastle, School of Engineering, Australia

[a]tomajuan@fceia.unr.edu.ar, [b]sjunco@fceia.unr.edu.ar, [c]alejandro.donaire@newcastle.edu.au

ABSTRACT

This work presents the design of stabilising controllers for the DC-DC boost and buck-boost power electronic converters using a passivity-based approach. The first step in the controller design is the definition of a convenient transformation of the state vector. The first variable of the transformation is the flat output of the converter, and the second is a bijective function of the charge of the output capacitor. This alternative to a previous work by the authors, which also considers flat outputs for the state vector transformation, ensures the bijectivity of the complete transformation. The disadvantage is that the designer is not allowed to choose a closed loop energy function, thus having to solve a partial differential equation to find one. A nonlinear state feedback control law is finally obtained. Disturbance rejection is addressed using a dynamic estimator of the load current, using a technique from the literature. The controller performance is validated via digital simulation.

Keywords: DC-DC power electronic converters, passivity-based control, port-Hamiltonian systems, flatness-based control, Bond Graphs.

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. Closed loop control design of PEC is a key topic, as for high performance applications not only asymptotic stability must be assured but performance too. The challenge in the coming years lies in developing new techniques at the lowest possible cost, size and weight for emerging applications (Ojo, 2019). This motivates a new approach of solving control problems in PEC feeding nonlinear loads.

PEC are highly nonlinear dynamical systems whose dynamics can be represented by means of averaged models, where the control input is the duty cycle of the PWM controlled electronic switch. For the second order boost and buck-boost converters models, the duty cycle has relative degree one with respect to both system states, making the controller synthesis a difficult task. This obstacle can be overcome through a conveniently designed coordinate transformation using the flat variable of the converter and choosing the other variable as a bijective function of the output capacitor charge to ensure output voltage regulation.

The rationale of the controller design relies on finding a state feedback control law in the new coordinates using interconnection and damping assignment passivity-based control (IDA-PBC), such that the closed loop can be written in port-Hamiltonian (pH) form. This allows using the closed loop Hamiltonian as a Lyapunov candidate function to analyze stability. As the coordinate transformation is bijective, asymptotic stability of the output voltage is achieved via regulation of the equivalent equilibrium of the transformed coordinates.

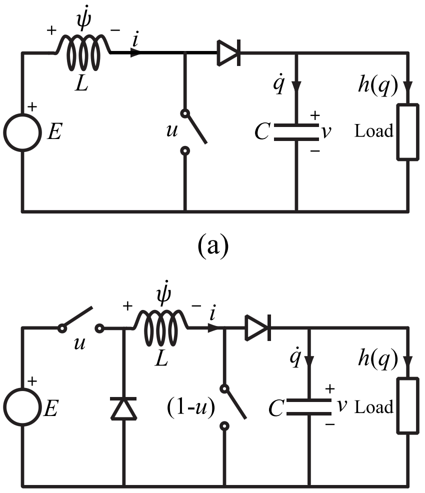
The design is enhanced using a disturbance dynamic estimator (He, et al., 2018) allowing the system to reject constant additive disturbances on the load side.

The remainder of this article is organized as follows: Section 2 presents the averaged models of the PEC and the control problem formulation. Section 3 introduces the general concepts of IDA-PBC and presents the controller design for the PEC. Disturbance estimators are developed in Section 4. Section 5 validates the controller designs via digital simulation, and conclusions are given in Section 6.

2. AVERAGED MODELS OF PEC

In this section the averaged models of the Boost and the Buck-Boost converters are presented. Figure 1 shows the equivalent circuits of the PEC under study. The inductance and the capacitor are assumed linear components with known parameters L and C. The averaged duty cycle of the electronic switch is represented by the continuous control signal $u \in (0,1)$. It is assumed that the PEC operate in the continuous conduction mode (CCM), and the state variables magnetic flux and electric charge $(\psi, q) > (0,0)\ \forall\, t$. Further, $h(q)$ is the current absorbed by a truly dissipative nonlinear load satisfying

$\{h(q) > 0 \, \forall \, q > 0\}$. The load and capacitor terminal voltages being the same allows to write the load current in terms of the capacitor charge. The reader is referred to (Mohan, Undeland, & Robbins, 1995) for further details on PEC averaged models.



(a)

(b)

Figure 1: Equivalent Circuits of (a) the boost converter, and (b) the buck-boost converter.

2.1. Boost Converter

The averaged state equations of the boost converter are given in (1):

$$\begin{aligned} \dot{\psi} &= E - u\frac{q}{C} \\ \dot{q} &= u\frac{\psi}{L} - h(q) \end{aligned} \; ; \text{with } \{u \in \mathbb{R} \mid u \in (0,1)\} \qquad (1)$$

2.2. Buck-Boost Converter

The averaged state equations of the boost converter are given in (2):

$$\begin{aligned} \dot{\psi} &= uE - (1-u)\frac{q}{C} \\ \dot{q} &= (1-u)\frac{\psi}{L} - h(q) \end{aligned} \; ; \text{with } \{u \in \mathbb{R} \mid u \in (0,1)\} \qquad (2)$$

2.3. Control Problem Formulation

The control problem for the converters consists in finding a map ξ such that the state feedback controller:

$$u = \xi(\psi, q) \qquad (3)$$

stabilizes the output voltage to a desired set-point $v^\star$ whilst ensuring internal stability. Since $v = q/C$, then output voltage regulation is equivalent to regulation of capacitor charge to the set point $q^\star = Cv^\star$. Internal stability is ensured if the equilibrium $(\psi^\star, q^\star)$ is stable.

3. PASSIVITY-BASED CONTROL

In this section, the general ideas of passivity-based control (PBC) for pH systems are first summarized and then the main result of the article is presented. Prior to the controller design, the coordinate transformation for each PEC is introduced. This will end up in a very similar matching equation for both converters reducing the effort needed to solve it.

3.1. General Concepts of IDA-PBC

Consider a dynamical system

$$\dot{x} = f(x) + g(x)u \qquad (4)$$

with $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$, the problem of stabilizing (4) using IDA-PBC consist on finding a mapping $\xi: \mathbb{R}^n \to \mathbb{R}^m$ such that the system (4) in closed loop with the controller $u = \xi(x)$ can be written in pH form as follows:

$$\dot{x} = [\boldsymbol{J}(x) - \boldsymbol{R}(x)]\frac{\partial H(x)}{\partial x} \qquad (5)$$

where the matrices $\boldsymbol{J} = -\boldsymbol{J}^T$ and $\boldsymbol{R} = \boldsymbol{R}^T \geq 0$ describe the interconnection and dissipation structure, and the function $H: \mathbb{R}^n \to \mathbb{R}$ is the Hamiltonian representing the total energy stored in the system. Let $x^\star$ be the minimizer of the Hamiltonian: $x^\star = \arg\min\{H(x)\}$, then $x^\star$ is a stable equilibrium point of the system (5). Moreover, under some detectability conditions, the equilibrium is asymptotically stable. If the following matching equation has a solution:

$$g(x)^\perp f(x) = g(x)^\perp [\boldsymbol{J}(x) - \boldsymbol{R}(x)]\frac{\partial H(x)}{\partial x} \qquad (6)$$

with $g(x)^\perp$ is the full-rank left annihilator of $g(x)$, then the control law can be synthesized as follows

$$u = [g^T g]^{-1} g^T \left[f - (\boldsymbol{J} - \boldsymbol{R})\frac{\partial H(x)}{\partial x}\right] \qquad (7)$$

See (Ortega & García-Canseco, 2004) for further details on IDA-PBC method.

3.2. Rationale of the Design Method

In the sequel, prior to the controller design a coordinate transformation for each converter will be introduced to ease the effort needed for the controller synthesis. The main methodological contribution of this paper consist in providing:

- A bijective state transformation $(\psi, q) \to (y, z)$ ensuring that driving $(y, z) \to (y^\star, z^\star)$ is equivalent to drive $(\psi, q) \to (\psi^\star, q^\star)$
- A matching equation with a similar structure for both converters, which can be easily solved by integration

- Finding a similar control law for both converters under the proposed coordinate transformation.

3.3. Boost Converter

3.3.1. Coordinate transformation

Consider the following coordinate transformation:

$$\begin{aligned} y &= \frac{\psi^2}{2L} + \frac{q^2}{2C} \\ z &= \frac{q^2}{2C} \end{aligned} \tag{8}$$

The original variables, are related to the new ones as follows:

$$\begin{aligned} \psi &= \sqrt{2L}\,(y-z)^{\frac{1}{2}} \\ q &= \sqrt{2C}\,z^{\frac{1}{2}} \end{aligned} \tag{9}$$

Time derivation of y and z leads to the following dynamics:

$$\begin{aligned} \dot{y} &= E\frac{\psi}{L} - \frac{q}{C}\,h(q) \\ \dot{z} &= \frac{q}{C}u\frac{\psi}{L} - \frac{q}{C}h(q) \end{aligned} \tag{10}$$

A new control input is defined as $m = u\frac{q}{C}\frac{\psi}{L}$. In terms of the new coordinates and the load power $P(q) = \frac{q}{C}h(q)$, the boost converter state space model can be written in the form $[\dot{y} \quad \dot{z}]^{\top} = f(y,z) + g\,m$:

$$\begin{bmatrix} \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z) \\ -P_z(z) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} m \tag{11}$$

where $P_z(z) = P\big(q(z)\big)$ stands for the load power, in terms of the coordinate z. As expected, the new states equations are power balances. Recall that y stands for the total energy stored in the system and z for the energy stored in the capacitor. As the averaged model is valid for $(\psi, q) > (0,0)$, then $(y-z) > 0,\ \forall\,(y,z)$.

3.3.2. Controller Design

We present here the main result for the boost converter, under the assumption of a known load VA (Volt-Ampère) characteristic.

Proposition 1: Consider the following controller for system (11):

$$m = E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z^\star) + K_y(y-y^\star) + P_z(z) + r\left(E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z)\right) \tag{12}$$

with $K_y > 0$ and $r > 0$. $z^\star = {q^\star}^2/(2C)$ is the desired equilibrium value for the capacitor energy, and $y^\star = \frac{{\psi^\star}^2}{2L} + \frac{{q^\star}^2}{2C}$ is the equilibrium for the total stored energy in the boost converter. It is assumed that $\frac{\partial P_z(z)}{\partial z} \geq 0\ \forall\, z > 0$. System (11) in closed loop with the controller given in (12) has the following properties:

P1. The closed loop dynamics can be written in pH form as given in (13) with the Hamiltonian in (14):

$$\begin{bmatrix} \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & -r \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial H(y,z)}{\partial y} \\ \frac{\partial H(y,z)}{\partial z} \end{bmatrix} \tag{13}$$

$$H(y,z) = E\frac{2}{3}\sqrt{\frac{2}{L}}(y-z)^{\frac{3}{2}} + \int_0^z P_z(\alpha)d\alpha + \frac{K_y}{2}\left(y - \frac{P_z(z^\star)}{K_y} - y^\star\right)^2 \tag{14}$$

$H(y,z)$ is a positive definite function, as $P\big(q(z)\big) = P_z(z)$ is also a positive definite function $\forall\, z > 0$.
From (13), the closed loop interconnection and dissipation matrices are the following constant matrices:

$$\boldsymbol{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{R} = \begin{bmatrix} 0 & 0 \\ 0 & r \end{bmatrix}$$

P2. The controller ensures the asymptotic stability of the equilibrium $(y^\star, z^\star)$.

Proof: The claim in P1 is easily proved taking into account that:

$$\begin{bmatrix} \frac{\partial H(y,z)}{\partial y} \\ \frac{\partial H(y,z)}{\partial z} \end{bmatrix} = \begin{bmatrix} E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - K_y(y-y^\star) - P_z(z^\star) \\ -E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} + P_z(z) \end{bmatrix} \tag{15}$$

and matching $\dot{y}$ given by (11) and (13). The next step is to use (12) in (11) to obtain:

$$\dot{z} = E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z^\star) + K_y(y-y^\star) + {} + r\left(E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z)\right) \tag{16}$$

$$\dot{z} = \frac{\partial H(y,z)}{\partial y} - r\,\frac{\partial H(y,z)}{\partial z} \tag{17}$$

completing the proof of the claim in P1. Asymptotic stability will be proved using the closed loop Hamiltonian (14) as a Lyapunov candidate function. First recall that $\sqrt{(2/L)}\ (y-z)^{\frac{1}{2}} = E\ (\psi/L)$:

$$E\sqrt{\frac{2}{L}}\ (y^{\star}-z^{\star})^{\frac{1}{2}} = E\frac{\psi^{\star}}{L} = P(q^{\star}) = P_z(z^{\star}) \qquad (18)$$

Then, we conclude:

$$\begin{bmatrix} \frac{\partial H(y,z)}{\partial y} \\ \frac{\partial H(y,z)}{\partial z} \end{bmatrix}_{(y^{\star},z^{\star})} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad (19)$$

Integrability condition is fulfilled as:

$$\frac{\partial}{\partial z}\left(\frac{\partial H(y,z)}{\partial y}\right) = \frac{\partial}{\partial y}\left(\frac{\partial H(y,z)}{\partial z}\right) = -E\frac{1}{2}\sqrt{\frac{2}{L}}\ (y-z)^{\frac{-1}{2}} \qquad (20)$$

Finally, it must be ensured that $(y^{\star},z^{\star})$ is a minimum of $H(y,z)$:

$$\begin{aligned} \frac{\partial}{\partial y}\left(\frac{\partial H(y,z)}{\partial y}\right)\Big|_{(y^{\star},z^{\star})} &= \frac{E}{2}\sqrt{\frac{2}{L}}\ (y^{\star}-z^{\star})^{-\frac{1}{2}} + K_y y^{\star} \\ \frac{\partial}{\partial y}\left(\frac{\partial H(y,z)}{\partial y}\right)\Big|_{(y^{\star},z^{\star})} &> 0 \end{aligned} \qquad (21)$$

and the determinant of the Hessian of $H(y,z)$ evaluated in the equilibrium:

$$\nabla^2 H(y,z)|_{(y^{\star},z^{\star})} = \frac{E}{2}\sqrt{\frac{2}{L}}(y^{\star}-z^{\star})^{-\frac{1}{2}}\left(K_y\, y^{\star} + \frac{\partial P_z(z)}{\partial z}\Big|_{z^{\star}}\right) + \left(K_y\, y^{\star}\cdot\frac{\partial P_z(z)}{\partial z}\Big|_{z^{\star}}\right) \qquad (22)$$

under the assumption that $\frac{\partial P(q)}{\partial q} \geq 0$ $\left(\Rightarrow\ \frac{\partial P_z(z)}{\partial z} \geq\ 0\ ;\ \forall\, q \geq 0\right)$ and recall $K_y > 0$ then $\nabla^2 H(y,z)|_{(y^{\star},z^{\star})} > 0$. This completes the proof. ■

Remark 1: $P_z(z) = P\big(q(z)\big)$ and $\frac{\partial P_z(z)}{\partial z} = \frac{\partial P(q(z))}{\partial q}\frac{\partial q(z)}{\partial z}$. $\frac{\partial q(z)}{\partial z} > 0$, this is given by the coordinate transformation. Then, if $\frac{\partial P(q)}{\partial q} \geq 0 \Rightarrow \frac{\partial P_z(z)}{\partial z} \geq 0$.

Remark 2: Equilibrium $(\psi,q) = (\psi^{\star},q^{\star})$ is achieved through the equilibrium of the energy variables $(y,z) = (y^{\star},z^{\star})$.

3.4. Buck-Boost Converter

3.4.1. Coordinate transformation

Recall the dynamics of this converter given by system (2). The following convenient coordinate transformation is proposed:

$$\begin{aligned} y &= \frac{\psi^2}{2L} + \frac{q^2}{2C} + Eq \\ z &= \frac{q^2}{2C} + Eq \end{aligned} \qquad (23)$$

The original variables then, can be written in terms of y and z as:

$$\begin{aligned} \psi &= \sqrt{2L}\ (y-z)^{\frac{1}{2}} \\ q &= C\left(\left(E^2 + \frac{2}{C}z\right)^{\frac{1}{2}} - E\right) \end{aligned} \qquad (24)$$

Time derivation of y and z leads us to the following dynamics:

$$\begin{aligned} \dot{y} &= E\frac{\psi}{L} - \left(\frac{q}{C} + E\right)\ h(q) \\ \dot{z} &= \left(E + \frac{q}{C}\right)\frac{\psi}{L}(1-u) - \left(E + \frac{q}{C}\right)h(q) \end{aligned} \qquad (25)$$

For this case the new control input defined is $m = (1-u)\left(\frac{q}{C} + E\right)\frac{\psi}{L}$, and the current absorbed by the load in terms of z is written as $h\big(q(z)\big) = h_z(z)$. It is convenient to define the following quantity $P_z(z) = h_z(z)\left(E^2 + \frac{2}{C}z\right)^{\frac{1}{2}}$, which has units of power. Then, the state space dynamics of the buck boost converter can be written in the form $[\dot{y}\quad \dot{z}]^{\mathsf{T}} = f(y,z) + g\, m$:

$$\begin{bmatrix} \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} E\sqrt{\frac{2}{L}}\ (y-z)^{\frac{1}{2}} - P_z(z) \\ -P_z(z) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} m \qquad (26)$$

As expected, once again, the new state equations are power balances. Recall that y stands for the total energy stored in the converter plus the term Eq, and the variable z is the output capacitor energy plus the term Eq. The averaged model is valid for $(\psi,q) > (0,0)$ implying that $y > z > 0$, then $(y-z) > 0$, $\forall\,(y,z)$. Recalling $h_z(q) > 0$ then $P_z(z) > 0$.

3.4.2. Controller Design

The controller is designed for system (26), assuming a known load VA characteristic.

Proposition 2: Consider the following controller for system (26):

$$\begin{aligned} m = {} & E\sqrt{\frac{2}{L}}\ (y-z)^{\frac{1}{2}} - P_z(z^{\star}) + K_y(y-y^{\star}) + \\ & P_z(z) + r\left(E\sqrt{\frac{2}{L}}\ (y-z)^{\frac{1}{2}} - P_z(z)\right) \end{aligned} \qquad (27)$$

with $K_y > 0$ and $r > 0$. $z^\star = \frac{q^{\star 2}}{2C} + Eq^\star$ and $y^\star = \frac{\psi^{\star 2}}{2L} + \frac{q^{\star 2}}{2C} + Eq^\star$. It is assumed that $\frac{\partial P_z(z)}{\partial z} \geq 0\ \forall\, z > 0$.

Remark 3: Properties P1 and P2 are guaranteed by the control law (27), since the coordinate transformation defined by (23) has the same structure as (8) defined for the boost converter. Hence the proof will be omitted.

4. DISTURBANCE ESTIMATION

A constant disturbance estimation method is presented in this section. Constant current disturbances in the output stage of the two converters under study are considered. The method is based on the design for unknown constant power load in (He, et al., 2018) and is an Immersion & Invariance design.

4.1. Boost Converter

Consider the state equation of the capacitor charge:

$$\dot{q} = u\frac{\psi}{L} - h_o(q) - \bar{\imath} \tag{28}$$

where $\bar{\imath}$ is the constant unknown current disturbance and $h_o(q)$ is the known original nonlinear load. The estimated current is defined as:

$$\hat{\imath} = \bar{\imath} + \tilde{\imath} \tag{29}$$

hence $\tilde{\imath}$ is the estimation error that must be driven to zero. To do so, consider the following proposition:

Proposition 3: Dynamic estimator for $\bar{\imath}$:

$$\begin{aligned} \hat{\imath} &= -\tfrac{1}{2}k_q\, q + k_q\alpha \\ \dot{\alpha} &= \tfrac{1}{2}\left(u\tfrac{\psi}{L} - h_o(q) - \hat{\imath}\right) \end{aligned} \tag{30}$$

where $\hat{\imath}$ converges to $\bar{\imath}$, with $k_q > 0$.

Proof: Computing the time derivative of $\hat{\imath}$ and then using the expression given for $\dot{\alpha}$ yields:

$$\begin{aligned} \dot{\hat{\imath}} &= -\tfrac{1}{2}k_q\, \dot{q} + k_q\dot{\alpha} \\ \dot{\hat{\imath}} &= -\tfrac{1}{2}k_q\left(u\tfrac{\psi}{L} - h_o(q) - \bar{\imath}\right) + k_q\tfrac{1}{2}\left(u\tfrac{\psi}{L} - h_o(q) - \hat{\imath}\right) = \tfrac{1}{2}k_q\bar{\imath} - \tfrac{1}{2}k_q\hat{\imath} \\ \dot{\hat{\imath}} &= -\tfrac{1}{2}k_q\tilde{\imath} \end{aligned} \tag{31}$$

As $\bar{\imath}$ is constant $\Rightarrow \dot{\bar{\imath}} = 0 \Rightarrow \dot{\hat{\imath}} = \dot{\tilde{\imath}}$. Finally, $\dot{\tilde{\imath}} = -\frac{1}{2}k_q\tilde{\imath}$, completing the proof. ■

The gain k_q will then be used to tune the time response of the estimator.

4.2. Buck-Boost Converter

Consider a constant unknown disturbance on the output stage of this converter. Then, the capacitor charge dynamics is:

$$\dot{q} = (1-u)\frac{\psi}{L} - h_o(q) - \bar{\imath} \tag{32}$$

where $\bar{\imath}$ is the constant unknown current disturbance and $h_o(q)$ is the known original nonlinear load.

Proposition 4: Dynamic estimator for $\bar{\imath}$:

$$\begin{aligned} \hat{\imath} &= -\tfrac{1}{2}k_q\, q + k_q\alpha \\ \dot{\alpha} &= \tfrac{1}{2}\left((1-u)\tfrac{\psi}{L} - h_o(q) - \hat{\imath}\right) \end{aligned} \tag{33}$$

with $k_q > 0$.

Remark 4: The convergence proof of the estimator is omitted as it follows the same procedure used for the boost converter estimator in the previous subsection.

4.3. Controller Enhancement

This section is intended to introduce the disturbance estimation into the control law previously designed for the PEC under the assumption of a perfectly known load VA characteristic and perform an enhancement that preserves the desired equilibrium of the output voltage. The approach the authors applied in (Tomassini, Donaire, Junco, & Pérez, 2017) for the Buck converter case is not applicable now as the equilibrium of the variable y is unknown in presence of a disturbance and both gradients of the Hamiltonian $H(y,z)$ depend on both y and z variables. This motivates the estimator design development we propose in this section. Taking into account the constant disturbance, the total current of the output stage of the converter is given by:

$$h(q(z)) = h_o(q(z)) + \bar{\imath} \tag{34}$$

where $\bar{\imath}$ is a constant unknown current, so consider the following load current estimation:

$$\hat{h}(q(z)) = h_o(q(z)) + \hat{\imath} = h_o(q(z)) + \bar{\imath} + \tilde{\imath} \tag{35}$$

Recall for both Boost and Buck-Boost converters, the term $P_z(z) = P\big(q(z)\big)$ needs the value of $\bar{\imath}$. The estimation of P_z:

$$\hat{P}_z(z) = P_z(z) + \tilde{P}_z(z) \tag{36}$$

with, for the Boost converter:

$$\hat{P}_z(z) = \frac{q(z)}{C}\, h(q(z)) + \frac{q(z)}{C}\tilde{\imath} \tag{37}$$

and for the Buck-Boost converter:

$$\hat{P}_z(z) = \left(E + \frac{q(z)}{C}\right) h(q(z)) + \left(E + \frac{q(z)}{C}\right) \tilde{\imath} \qquad (38)$$

Using the power estimated functions, the control laws (12) and (27):

$$\widehat{m} = E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - \hat{P}_z(z^\star) + K_y(y-y^\star) + \hat{P}_z(z) + r\left(E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - \hat{P}_z(z)\right) \qquad (39)$$

$$\widehat{m} = E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z^\star) + K_y(y-y^\star) + P_z(z) + r\left(E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z)\right) - \tilde{P}_z(z^\star) + \tilde{P}_z(z) - r\tilde{P}_z(z) = m + \widetilde{m} \qquad (40)$$

where the $\widetilde{(\cdot)}$ terms vanish as $\tilde{\imath} \to 0$. The closed loop dynamics in the (y,z) variables are:

$$\begin{bmatrix}\dot{y}\\ \dot{z}\end{bmatrix} = \begin{bmatrix} E\sqrt{\frac{2}{L}}\,(y-z)^{\frac{1}{2}} - P_z(z) \\ -P_z(z)\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix} m + \begin{bmatrix}0\\1\end{bmatrix} \widetilde{m}(\tilde{\imath}) \qquad (41)$$

$$\dot{\tilde{\imath}} = -\frac{1}{2} k_q \tilde{\imath} \qquad (42)$$

The dynamics given by (41) and (42) define a cascaded system. Proposition 4.1 of (Sepulchre, Jankovic, & Kokotovic, 2012) ensures asymptotic stability of system (41), (42) as the origin of the $\tilde{\imath}$-subsystem is asymptotically stable.

5. SIMULATION RESULTS

In this section the controller performance is tested via digital simulation on a Buck-Boost converter system. The model parameters are $L = 16mH$, $C = 1.2mF$ and $E = 50V$. The load VA characteristic is defined in terms of the capacitor charge in (43) and graphically shown in VA characteristic in Figure 2.

$$h(q) = \frac{q}{C}\frac{1}{51} - \left(\frac{q}{C}\frac{1}{51}\right)^3 + \left(\frac{q}{C}\frac{1}{68}\right)^5 + \text{atan}\left(\frac{2}{3}\frac{q}{C}\right) \qquad (43)$$

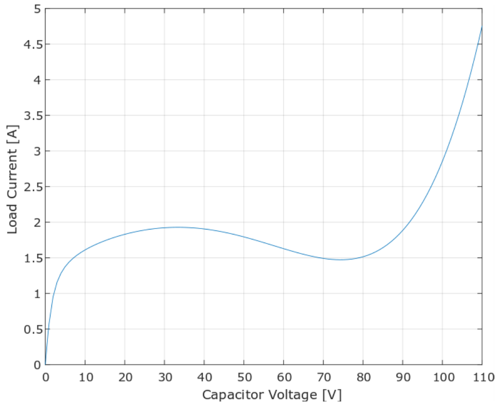
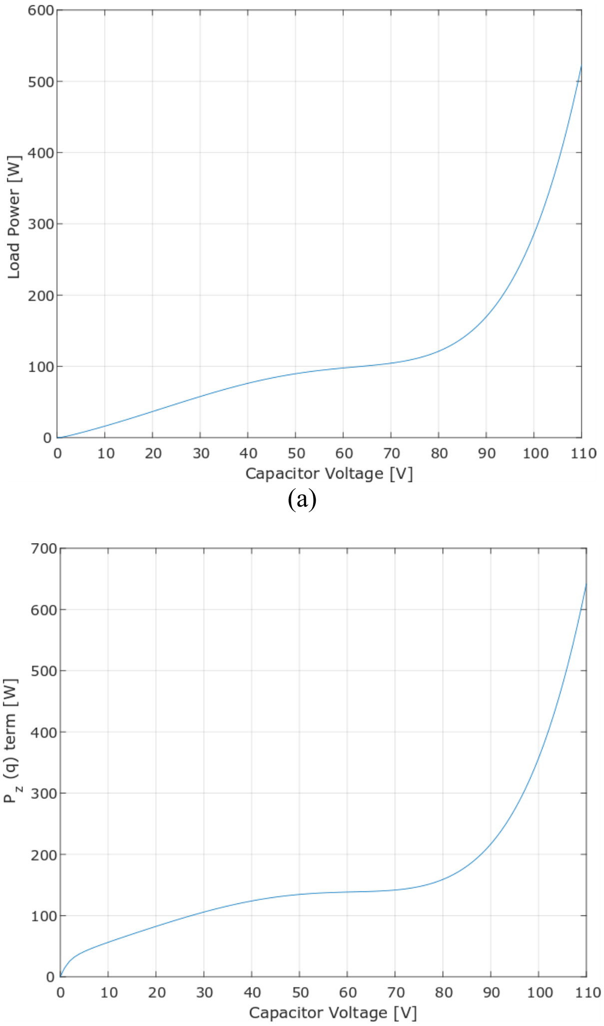


Figure 2: Load current, in terms of the output capacitor voltage.

where the capacitor voltage is $v_c = q/C$. The figure below shows the Load Power and the quantity $h(q)\left(E + \frac{q}{C}\right)$ where the last stands for the term $P_z(z)$ in terms of the capacitor voltage:



(a)

(b)

Figure 3: Load Power (a), and term $P_z(q)$ (b) in terms of the capacitor voltage.

The controller tuning parameters are $k_y = 100$ and $r = 12$. In the first simulation set no disturbance is considered while in the latter a constant current is considered and consequently rejected using a dynamic estimator. The controller can incorporate (43) as a "lookup table", with the necessary amount of breakpoints.

Remark 5: As for the controller tuning, considering that the closed loop is nonlinear, a set of parameters for K_y and r was obtained using linearization of the closed loop around the equilibrium $\frac{q^\star}{C} = 50V$, for an acceptable response time. Finally, noting that $\dot{H}(y,z) = -r\left(\frac{\partial H(y,z)}{\partial z}\right)^2$ it is concluded that the value of the gain r has a direct impact in the response time. The performance will be tested for different values of r in the next subsection.

5.1. Controller performance test

In this simulation experiment the controller performance is tested via an output voltage reference change. The initial conditions are set corresponding to $\frac{q}{C} = 50V$. The output voltage reference (dashed line) is changed to $35V$ at $t = 0.1s$, then changed to $60V$ at $t = 0.8s$ and finally to $85V$ at $t = 1.4s$. Notice that the two first reference values correspond to incremental negative resistance zone of the load VA characteristic of the load. Different response times are obtained by tuning the parameter r starting with $r = 12$ and reducing it.

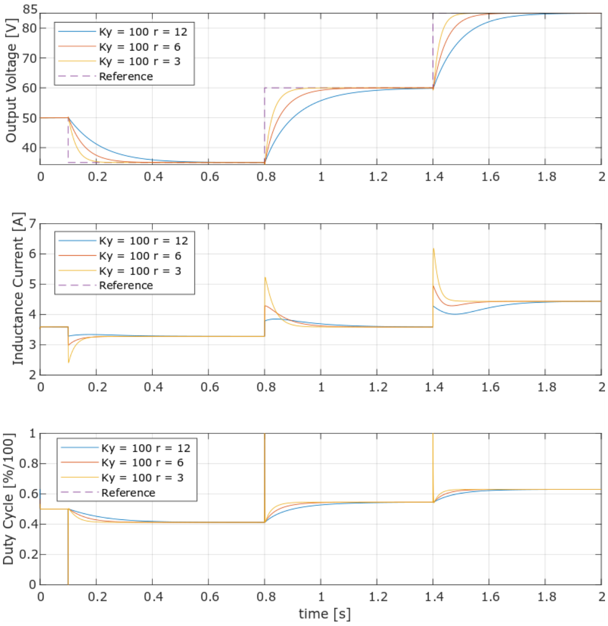


Figure 4: Time plots of the Buck-Boost converter original variables for different values of r.

And the time evolution of the transformed variables:

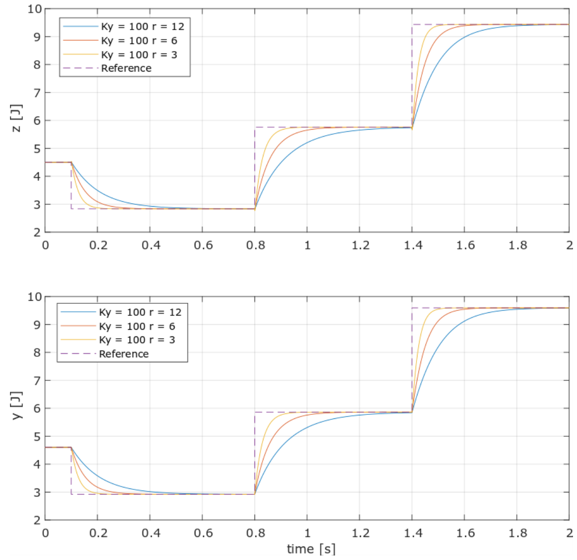


Figure 5: Time plots of the transformed variables for different values of r.

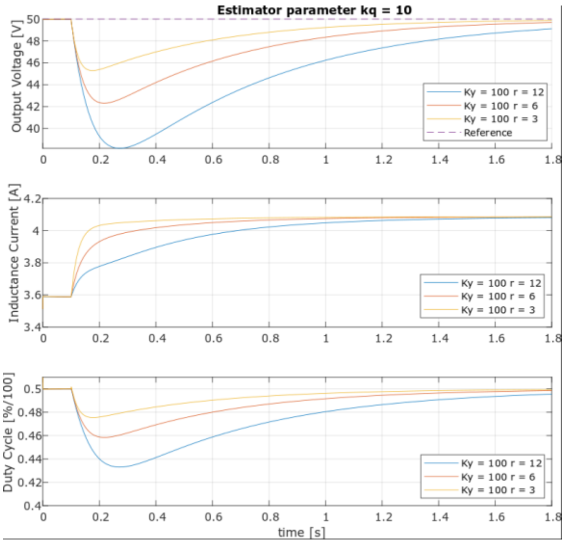
The controller achieves the control objective through the regulation of the variables y and z, for output reference voltages greater and lower than E; and including the negative incremental resistance zones of the load VA characteristic.

5.2. Simulations with disturbance estimator

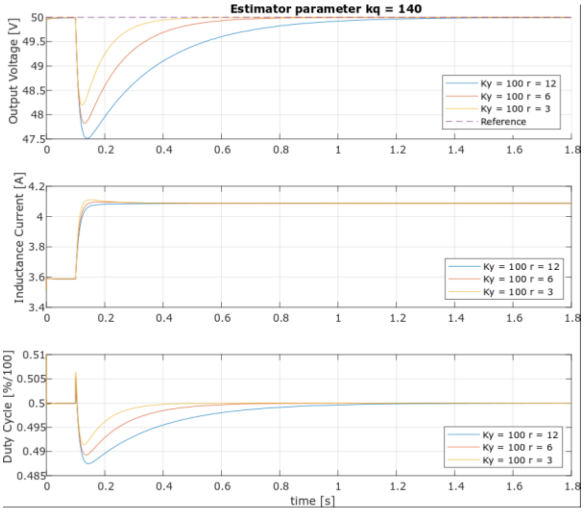
In this subsection the controller performance is tested with the disturbance estimator implemented. The first set of simulations shows only the disturbance rejection behaviour while the second set shows how the estimator degrades the performance of the controller presented in Subsection 5.1.

5.2.1. Disturbance rejection behaviour

Consider a disturbance of $\bar{\imath} = 0.25A$ applied at $t = 0.1s$ when the system is in steady state and regulating an output voltage of $50V$:



(a)



(b)

Figure 6: Disturbance rejection simulation results for estimator parameter $k_q = 10$ (a) and $k_q = 140$ (b).

The disturbance is satisfactorily rejected thanks to the estimator. Note that the response time of the system varies not only by means of k_q but also by r. This is because the estimator dynamics depends on the states (ψ, q) and on the duty cycle u.

5.2.2. Controller performance degradation

For this simulation set, consider the system in equilibrium corresponding to $\frac{q}{C} = 50V$ in presence of a disturbance of $\bar{\imath} = 0.25A$. Estimator parameter $k_q = 140$. The same voltage reference changes used in Subsection 5.1 are performed in this simulation set:

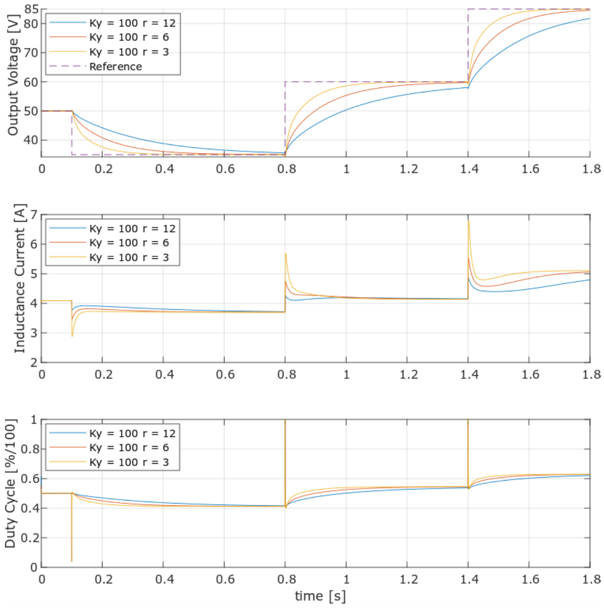


Figure 7: Controller performance with disturbance estimator implemented.

It is observed that the disturbance estimator degrades the transient performance of the whole system, as the response time increases and the inductance current overshoot is larger. On the other hand the estimator is needed to achieve the control objective in presence of a disturbance.

6. CONCLUSIONS

In this paper a control system design method for two DC-DC power electronics converters has been presented, where the control input acts through the system interconnection structure. Flatness–based coordinate transformation together with a state dependent input transformation allowed to write the open loop dynamics of both converters in linear affine form with a constant input matrix. This allow for an easy solution of the matching equation, usually the most difficult task in the IDA-PBC methodology, yielding a closed-loop in pH form with constant structure and dissipation matrices.

The main advantage of this solution is that it is not restricted to linear loads but works for a wide variety of nonlinear loads, including nonlinearities with negative incremental resistance in their Volt-Ampère characteristic. The resulting control law seems complicated in the transformed coordinates but in the original variables it can be easily understood as composed of a sum of power and energy terms. The controller implementation is also a simple task. The closed loop performance can be adjusted via experimental tuning (digital simulation) of a single parameter.

Instead of adding a PI for disturbance rejection - a technically difficult task in this case if intending to preserve the pH-form - the alternative of using a dynamic estimator was developed, which proved to preserve the equilibrium of the transformed variable, thus fulfilling the control objective. It is worthy to note that, adding the estimator has an impact on the original controller, increasing the response time and the current overshoot during transients. Increasing the speed of the estimator reduces the performance degradation of the controller, but demands a higher control effort. Depending on the application, a compromise relationship between response time and control effort must be found by the designer.

Future work will focus on extending the application of this control system design approach to DC electrical grids including energy storage and distributed sources, where the control of the power flow to satisfy the energy management requirements of the system is performed by these kind of converters.

ACKNOWLEDGEMENTS

The authors thank the National University of Rosario, Argentina, for the financial support to projects PID-UNR 1ING573 and ViTec "Red Eléctrica Inteligente Experimental …". J. Tomassini thanks Dr. M. Nacusse, a member of LAC for helpful discussions on pH modelling and IDA-PBC control methodology.

REFERENCES

He, W., Soriano-Rangel, C. A., Ortega, R., Astolfi, A., Mancilla-David, F., & Li, S. (2018). Energy shaping control for buck–boost converters with unknown constant power load. *Control Engineering Practice, 74, 33-43.*

Mohan, N., Undeland, T., & Robbins, W. (1995). *Power Electronics: converters, applications and design.* John Wiley & Sons, INC.

Ojo, J. O. (April de 2019). *Address from the Editor-In-Chief of the IEEE J. Emerging & Selected Topics in Power Electronics.* Obtenido de https://www.ieee-pels.org/publications/ieee-journal-of-emerging-and-selected-topics-in-power-electronics

Ortega, R., & García-Canseco, E. (2004). Interconnection and Damping Assignment Passivity-Based Control: A Survey. *European Journal of Control, 10*, 432-450. doi:http://dx.doi.org/10.3166/ejc.10.432-450

Sepulchre, R., Jankovic, M., & Kokotovic, P. V. (2012). *Constructive nonlinear control.* Springer Science & Business Media.

Tomassini, J., Donaire, A., & Junco, S. (2018). Energy-And flatness-based control of DC-DC converters with nonlinear load. *International Conference on Integrated Modeling and Analysis in Applied Control and Automation (IMAACA 2018).* Budapest, Hungary.

Tomassini, J., Donaire, A., Junco, S., & Pérez, T. (2017). A port-Hamiltonian approach to stabilization and disturbance rejection of DC-DC Buck converter with nonlinear load. *ASCC 2017 – the 2017 Asian Control Conference , 17 – 20 December 2017, Gold Coast, Australia.*

AUTHORS BIOGRAPHY

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 Argentine National Council of Scientific and Technical Research, CONICET. His main research interests are on IDA-PBC control, renewable energy and smart grids.

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 and Mechatronics.

Alejandro G. Donaire received the Electronic Engineering and PhD degrees in 2003 and 2009, respectively, from the Universidad Nacional de Rosario, 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. In 2017, he joined the Institute for Future Environments, School of Electrical Engineering and Computer Science, Queensland University of Technology, Australia. In 2019, he joined the School of Engineering, The University of Newcastle, Australia, where he conducts his academic activities 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.

COMPARISON OF TRAJECTORY TRACKING AND OBSTACLE AVOIDANCE STRATEGIES FOR A MULTI-AGENT DYNAMIC SYSTEM

Martín Crespo[a,b]**,Matías Nacusse**[a,b]**, Sergio Junco**[a]

[a] Laboratorio de Automatización y Control (LAC), Departamento de Control, FCEIA, UNR. Rosario, Argentina
[b] CONICET: Consejo Nacional de Investigaciones Científicas y Técnicas. Argentina

[a]crespom@fceia.unr.edu.ar, nacusse@fceia.unr.edu.ar, sjunco@fceia.unr.edu.ar

ABSTRACT

This paper designs different laws for formation control and obstacle avoidance for a group of robots with holonomic dynamics and presents a set of simulations that validate and compare them. The Bond Graph methodology, used to design the control laws, together with the physical interpretation of both the obstacles and the interaction between the robots, allows addressing the problem with an energy-based approach. A multi agent scheme is proposed where a leader drives a formation of agents through a desired path. The formation is organized in different hierarchy levels and the control laws for the robots arise from considering the interaction among them through virtual dampers and springs. Two different techniques are addressed for collision avoidance and three scenarios are presented to test the different techniques for coordinated tracking and obstacle collision avoidance. Simulation results are presented to show the good performance of the control system.

Keywords: formation control, obstacle avoidance, energy based methods, bond graphs.

1. INTRODUCTION

In the last decades the lower prices of the robots made feasible the utilization of large number of robots for several tasks such as surveillance, search and rescue or data acquisition. The problem of formation and movement in specified geometrical shapes has been widely studied.

The concept of coordinated tracking is an extension of the widely known problem of trajectory tracking (Egerstedt & Hu, 2001). That is, the objective is to find a coordinated control scheme for a group of robots that maintaining a desired scheme of formation could perform a desired task as a group.

This paper tackles the problems of Formation Control and Vehicle Following control (A. & H., 2018) for a group of holonomic vehicles which are represented as masses in the plane.

These problems were treated in the literature with multiple approaches, depending on the sensing capabilities of each agent and the desired topology, to mention: Leader Following, Predecessor-Following (also known as unidirectional connection), Leader-Predecessor follower, Predecessor-Successor (also known as bidirectional connection), etc. The reader must refer to (A. & H., 2018), (Zhang Z. & L., 2016), (R. & J., 2010), (S. & R., 2013) for a sound review of these topologies and others.

This paper uses a hierarchical unidirectional interconnection structure where each agent receives information about the relative position and the velocity of some of its surrounding robots. The robot acts as if it were connected with a spring and a damper with the surrounding agents but the reaction force of this interconnection may not affect the other agents.

Two interconnection structures are defined to generate multiple configurations. In the first one, called Scheme A, the agent receives information of only two surrounding robots, one from a superior level of hierarchy and the other from the same level. In the second scheme, called Scheme B, the robot receives information of four agents, two from a superior level of hierarchy and the others from the same level.

The problem of obtaining the obstacle avoidance law inspired in a physical phenomenon have been addressed in (Rezaee & Abdollahi, 2013) where each robot is modeled as an electric charge.

Several papers present obstacle avoidance approaches based on the artificial potential field concept (Khatib, 1986) (Alvarez D. & R., 2003). In (Matoui, Boussaid, & Abdelkrim, 2018) an attractive and a repulsive potential force are proposed to pull the robot toward the intended goal and to repel the robot from the zone of the obstacles, respectively, for real time obstacle avoidance. However, those methods suffer from points of local minima at which the robots become trapped. In (Park, Jeon, & Lee, 2001) a potential field approach with simulated annealing is proposed to avoid local minima. (Yun & Tan, 1997) proposes a switching control algorithm to avoid local minima. Even if successful, such kind of methods includes extra computation and is intrinsically less satisfactory than a method which avoids local minima from the outset (Graham & Buckingham, 1993).

The method proposed in (Connolly, Burns, & Weiss, 1990) utilizes a potential field characterized by a function that satisfies Laplace's Equation under

Dirichlet boundary conditions. The generated potential field does not contain any local minima over the region. The above techniques are presented mainly for the control of robot manipulators for both, obstacle avoidance and joint control in the case of redundant manipulators. Several other works propose potential fields for the coordination and obstacle avoidance of multi robots systems (Cai, Yang, Zhu, & Liang, 2007) (Leonard & Fiorelli, 2001).

The physical approach used here for both, formation control and obstacle avoidance call for multi-domain modeling frameworks. Besides the traditional Euler-Lagrange approach to modeling and control in robotics (Siciliano, Sciavicco, Villani, & Oriolo, 2010), the Bond Graph (BG) technique (Karnopp, Margolis, & Rosenberg, 2000) is increasingly gaining space as it is capable of representing the virtual interaction between the robots and the obstacles, very useful for analysis and simulation. The BG approach also provides methods to design control laws for physical systems (S. J. , 2004), (Dauphin-Tanguy G. & C., 1999).

The paper is organized as follows: in Section 2 the problem formulation is stated. Section 3 presents the solution to the formation control problem. In Section 4 two different control laws are designed to avoid obstacle collision of holonomic robots. In Section 5 simulation data is provided to illustrate the main results presented in the above sections. Finally, Section 6 draws some conclusions and provides future directions of research.

2. PROBLEM FORMULATION

The purpose of this paper is to present and compare different control strategies for a group of holonomic robots that ensure on the one hand a coordinated displacement from a starting location to a desired goal position and, on the other, obstacle collision avoidance. In the following sections three strategies are presented addressing these problems with different approaches. A control law that keeps the robots in a desired formation is common to all the three strategies. The objective of this common control law is to make the robots converge to the formation and to follow a robot leader while maintaining the stability of the formation.

2.1. First Strategy

In this strategy a rejection force based on a spring principle is proposed for obstacle avoidance. The control of the formation movement is done through a trajectory tracking control law applied to the leader robot.

2.2. Second Strategy

In this case, a control law based on a potential function is implemented in order to both avoid obstacle collision and drive the formation to the final position. The proposed potential function, that models the obstacles and the goal position with high and low values, respectively, satisfies the Laplace's equation in all the workspace. In that way, the gradient of the function drives the robots toward the goal while avoiding the obstacles.

2.3. Third Strategy

This strategy implements an obstacle avoidance law that results from modeling only the obstacles with a potential function. For its part, the movement of the formation is controlled by a trajectory tracking control law applied to the leader.

3. FORMATION CONTROL

In this section the formation control problem is presented as a coordinated tracking problem where the objective is to drive a group of robots in the plane while keeping a specific formation.

The formation, depicted in Figure 1, is composed by one leader and n agents organized in hierarchical levels. A pyramidal topology with a mesh of identical triangles is shown, but this does not imply a loss of generality as any other form can be specified through the adequate choice of the length of the triangles sides. The directions of the arrows represent the flow information that the robots exchange each other.

Thus, given a desired trajectory for the leader, the objective for the agents is to converge to the formation and to follow the leader while maintaining the stability of the formation and avoiding obstacle collision.

Notice that in the stationary state, with the leader located in its final position, the agent's formation may adapt any orientation around the leader. In this work, global orientation of the formation will not be controlled.

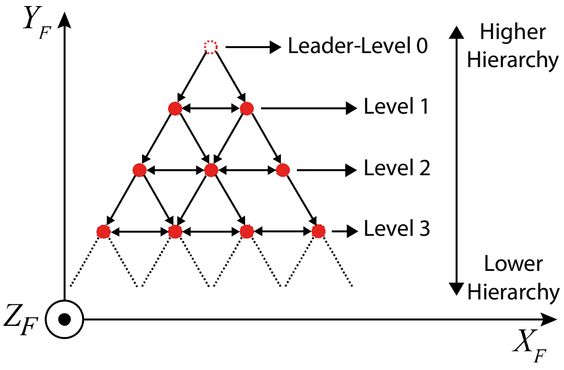


Figure 1: Interconnection network of the desired formation.

The control of the desired formation scheme is achieved through a decentralized and hierarchical architecture where each agent has its own control law. As the robots have holonomic dynamics, they may be treated as point masses. Thus, the dynamics of both the leader and the agents is simply given by $m\ddot{q} = u$, where $q = [q_x\ q_y]^T$ denotes the position of the robot in the plane and $u \epsilon \mathbb{R}^2$ is the control force vector expressed in Cartesian coordinates of an inertial reference system. The action exerted by the controller on the leader is

$$u_L = F_T + F_O \qquad (1)$$

where F_T and F_O represent the trajectory tracking control law and the obstacle avoidance force, respectively. In the case of the agents the control law of the robot is characterized by

$$u_A = F_c + F_O \qquad (2)$$

where F_c and F_O represent the convergence to the formation and the obstacle avoidance force of the agent, respectively.

This section deals with obtaining the feedback law F_C for the formation control of the agents.

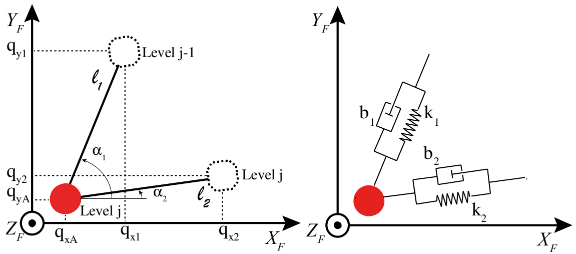
Each agent interacts with agents that belong to the same level of hierarchy and with those from the superior hierarchy. The proposed topology is of the type Predecessor-Following where the agents that belong to Level j receive position and velocity information from agents of level $j-1$ and are power coupled with those from Level j.

Taking into account the above idea, two kinds of interaction schemes arise considering the interconnection network of the formation shown in Figure 1. In the first one (Scheme A which corresponds to the agents on the periphery of the formation) the agent interacts with two other agents, one belonging to the same hierarchy level and the other to the superior level. In the second one (Scheme B which corresponds to the agents in the interior of the formation) the agent interacts with other four agents, two belong to the same hierarchy level and the other two to the superior one.

The control law for each agent comes from considering a virtual damper-spring connected between the agent and its interacting partners, as explained in the next sections. BG models of the formations are constructed ad hoc to obtain the control laws of the agents.

3.1. Scheme A

Consider an agent of Level j in the scheme A of interaction. As mentioned above, the agent interacts with two other agents as shown in Figure 2-a.



(a) (b)

Figure 2: Scheme A - (a) Geometric variables (b) Schematic diagram.

The schematic diagram in Figure 2-b shows the virtual dampers and springs connected with its two interacting agents. The formation control vector F_C for a peripheral agent, the red one at level j in Figure 2 for instance, arises from considering the virtual BG shown in Figure 3. There, sub index $i = 1$ indicates the agent in level $j-1$, while sub index $i = 2$ indicates the agent at the same level j. Notice that a dissipation term is added to the movement of the agents in the plane.

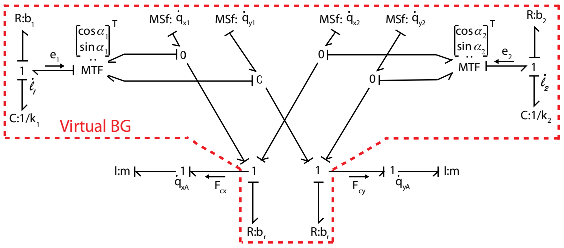


Figure 3: Scheme A – Virtual BG model.

The resulting control law is

$$F_c = \begin{bmatrix} F_{cx} \\ F_{cy} \end{bmatrix} = \begin{bmatrix} e_1 c_1 + e_2 c_2 - b_r \dot{q}_{xA} \\ e_1 s_1 + e_2 s_2 - b_r \dot{q}_{yA} \end{bmatrix} \qquad (3)$$

where b_r is the friction coefficient assigned to the movement of the agents in the plane, and $\mathrm{s_i}$ and $\mathrm{c_i}$ stand for $\sin \alpha_\mathrm{i} = \frac{q_{yi}-q_{yA}}{l_i}$ and $\cos \alpha_\mathrm{i} = \frac{q_{xi}-q_{xA}}{l_i}$, respectively for $i = 1{,}2$. The forces with the interacting agents are

$$e_i = k_i(l_i - l_{noi}) + b_i \dot{l}_i \qquad (4)$$

where k_i and b_i are the spring and damper coefficients of the interaction i, respectively. The distances between the agents are

$$l_i = \sqrt{(q_{xi} - q_{xA})^2 + \left(q_{yi} - q_{yA}\right)^2} \qquad (5)$$

And l_{noi} the natural length of the spring of the ith interaction. These lengths determine the topology of the mesh defining the geometric layout of the formation. The expression for $\dot{l}_i$ is

$$\dot{l}_i = cos\alpha_i(\dot{q}_{xi} - \dot{q}_{xA}) + sin\alpha_i(\dot{q}_{yi} - \dot{q}_{yA}) \qquad (6)$$

3.2. Scheme B

Consider an agent of Level j in the scheme B of interaction. As mentioned above, the agent interacts with other four agents as shown in Figure 4.

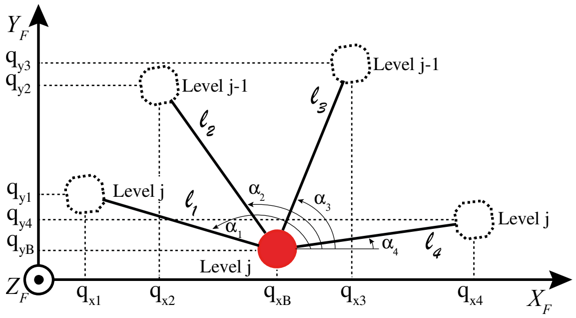


Figure 4: Scheme B - Geometric variables.

The distances l_i between the agents are

$$l_i = \sqrt{(q_{xi} - q_{xB})^2 + \left(q_{yi} - q_{yB}\right)^2} \tag{7}$$

while the forces e_i are the same as (5). The expression of $\dot{l}_i$ yields

$$\dot{l}_i = cos\alpha_i(\dot{q}_{xi} - \dot{q}_{xB}) + sin\alpha_i(\dot{q}_{yi} - \dot{q}_{yB}) \tag{8}$$

The schematic representation of the Scheme B is shown in Figure 5.

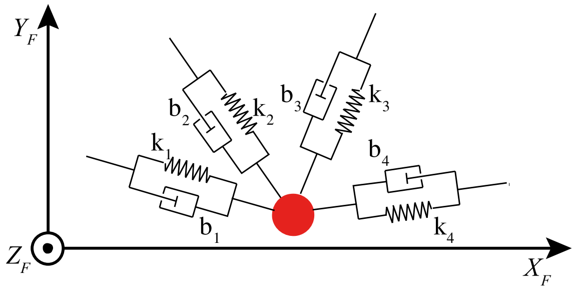


Figure 5: Scheme B - Schematic diagram.

In a similar way to scheme A the control vector F_c of the agent of scheme B, is obtained from the virtual BG depicted in Figure 6

$$F_c = \begin{bmatrix} F_{cx} \\ F_{cy} \end{bmatrix} = \begin{bmatrix} e_1c_1 + e_2c_2 + e_3c_3 + e_4\,c_4 - b_r\dot{q}_{xB} \\ e_1s_1 + e_2s_2 + e_3s_3 + e_4s_4 - b_r\dot{q}_{yB} \end{bmatrix} \tag{9}$$

where b_r is a friction coefficient, $c_i = \frac{q_{xi}-q_{xB}}{l_i}$, and $s_i = \frac{q_{yi}-q_{yB}}{l_i}$ for $i = 1,\cdots,4$.

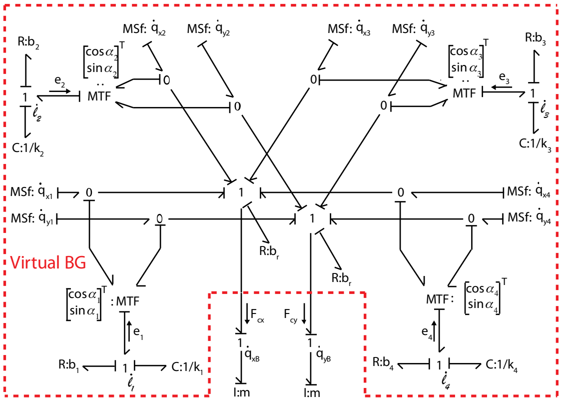


Figure 6: Scheme B - Bond Graph model.

Notice that the agents belonging to the same hierarchical level are power coupled. That is, the force of the link that joins two agents of the same level is the same and is computed by and applied to both of them.

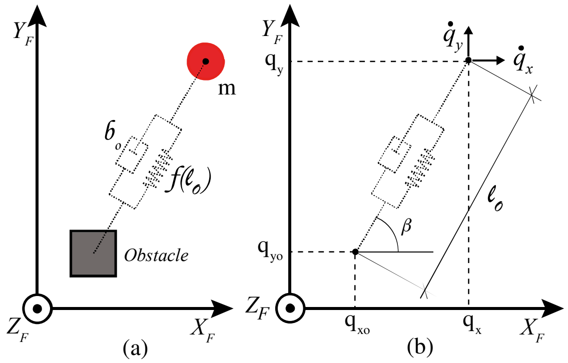
4. OBSTACLE AVOIDANCE & NAVIGATION

In this section the avoidance of both static and dynamic obstacles is addressed presenting two different techniques. The first one considers a repulsion force via a spring-damper based approach. The work (J. & Y., 1989) proposes a repulsive force for obstacle avoidance. In the second one the movement of the agents is inspired in the flow of a river in a valley. Thus, the workspace is modeled with a potential function where high values of potential are assigned to the obstacles (Connolly C. & R., 1990). Both techniques define the obstacle avoidance force F_O presented in (1) and (2) of the individual control laws of both, the leader and the agents, respectively.

Notice that in the former technique, the information about the obstacle becomes known to the robot through sensors during run-time whereas for the latter, the obstacle information is supposed to be known by the robot entirely before. This allows for considering dynamic obstacles in the first technique.

4.1. Spring-damper based approach

First the study of the obstacle avoidance problem is treated by means of a virtual spring-damper technique. This technique consists on applying a force to the center of the robot to move it away from the obstacle. The force is inversely proportional to the distance between the robot and the obstacle, and can be interpreted as the reaction force of a virtual spring attached to the obstacle and the center of the robot as shown in Figure 7-a. The virtual damper has been added to avoid oscillations during the rejection of the robots.



(a) (b)

Figure 7: (a) Schematic diagram (b) Geometric variables.

The distance l_o between the robot and the obstacle can be computed as

$$l_o = \sqrt{(q_x - q_{xo})^2 + (q_y - q_{yo})^2} \tag{10}$$

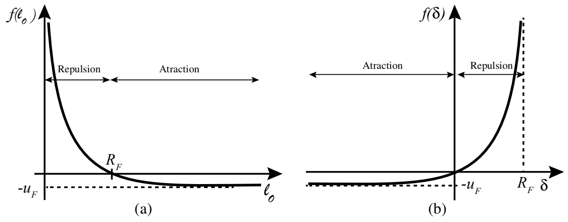
where q_{xo}, q_{yo} represent the position of the obstacle in the plane as depicted in Figure 7-b. The force of the virtual spring as a function of the distance l_o is chosen as

$$f(l_o) = \frac{k_o}{l_o^3} - \mu_F \tag{11}$$

where k_o is the spring constant and μ_F a positive constant value that allows negative excursion of the function $f(l_o)$ as shown in Figure 8-a. In this way, for distances l_o greater than

$$R_F = \sqrt[3]{k_o/\mu_F} \tag{12}$$

the force exerted by the spring is negative. This means that beyond a circumference of radius R_F centered in the obstacle the virtual spring exerts an attraction force to the robot.



(a) (b)

Figure 8: Avoidance function around the obstacle.

Even if in principle it is expected that the force applied to the agent by the spring vanishes as the robot moves away from the obstacle, it will be seen that this behavior allows treating the spring as a passive component. In fact, considering the change of coordinates $\delta = R_F - l_o$, equation (10) leads to

$$f(\delta) = \frac{k_o}{(R_F - \delta)^3} - \mu_F \tag{13}$$

This new function is evaluated in the first and third quadrant as shown in Figure 8-b. In that way, the addition of the term μ_F in the avoidance function and the change of coordinates ensure the passivity of the virtual spring component. The value that is assigned to μ_F is low and plays a secondary role.

Remark: *Notice that in an physical implementation the virtual spring may never work in traction since the sensing range of the sensors implemented in the robots is limited.*

Finally, reading the virtual BG shown in Figure 9, the implemented obstacle avoidance force vector F_o is

$$F_o = \begin{bmatrix} cos\beta \\ sin\beta \end{bmatrix} \left(f(\delta) + b_o \dot{\delta} \right) \tag{14}$$

and, considering a static obstacle

$$\dot{l}_o = cos\beta \dot{q}_x + sin\beta \, \dot{q}_y \tag{15}$$

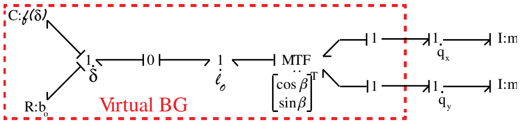


Figure 9: Virtual Spring – Bond Graph model.

However, as in (1) the force applied to the leader consists on the sum of two control laws obtained through different methods, the leader could be affected by local minima, becoming trapped in a position different from the final desired one. In this sense, to avoid local minima, an artificial potential method is presented.

4.2. Artificial potential

This method proposes a model of the environment in which each obstacle exerts a repulsive force while the goal position exerts an attractive force.

In this way a virtual potential field is generated in all the workspace. High and low values are assigned to the obstacles and goal position, respectively. As the generated field satisfies the Laplace's Equation under Dirichlet conditions, the resulting field is continuous in all the workspace and does not have any local minima (Connolly C. & R., 1990).

The generated potential field is a harmonic function $\emptyset$ in the domain $\Omega \epsilon \mathbb{R}^2$ that satisfies the Laplace's equation

$$\nabla^2 \emptyset = \frac{\partial^2 \emptyset}{\partial q_x^2} + \frac{\partial^2 \emptyset}{\partial q_y^2} = 0 \tag{16}$$

As it can be seen if the second derivatives are not zero, the curves of the function ϕ must have second derivatives with opposite signs, in that way if not plane, there is always a direction where the curve increases or decreases. Thus, the function ϕ does not have any local minima.

Numerical solutions for Laplace's Equation can be obtained from finite difference methods (Burden RL. & AC, 1981). Consider $u\left(q_{x_i}, q_{y_i}\right)$ a discretization of the function ϕ in the domain Ω. The second derivatives can be approximated by Taylor series as

$$\begin{aligned}\phi_{q_x q_x}\left(q_{x_i}, q_{y_i}\right) &= \frac{u\left(q_{x_{i+1}}, q_{y_i}\right) - 2u\left(q_{x_i}, q_{y_i}\right) + u\left(q_{x_{i-1}}, q_{y_i}\right)}{h^2} \\ \phi_{q_y q_y}\left(q_{x_i}, q_{y_i}\right) &= \frac{u\left(q_{x_i}, q_{y_{i+1}}\right) - 2u\left(q_{x_i}, q_{y_i}\right) + u\left(q_{x_i}, q_{y_{i-1}}\right)}{h^2}\end{aligned} \tag{17}$$

Where h is the step size to approximate the derivative. As ϕ satisfies the Laplace's Equation, $u\left(q_{x_i}, q_{y_i}\right)$ is expressed by

$$u\left(q_{x_i}, q_{y_i}\right) = \frac{u\left(q_{x_{i+1}}, q_{y_i}\right) + u\left(q_{x_{i-1}}, q_{y_i}\right) + u\left(q_{x_i}, q_{y_{i+1}}\right) + u\left(q_{x_i}, q_{y_{i-1}}\right)}{4} \tag{18}$$

To obtain the values of the harmonic function ϕ in the discretized points of the grid, the linear system is solved through iteration using the Gauss-Seidel method. This method consists in replacing repeatedly each element of the grid, using an iterative method, by the average of its adjacent elements. The boundary conditions are introduced specifying the values of the function $u\left(q_{x_i}, q_{y_i}\right)$ at the boundary of the domain $\partial\Omega$ (Dirichlet boundary conditions) and remain fixed.

The gradient of the potential function $\emptyset$ determines the vector force F_o that keeps away the robots from the obstacles and conducts them toward the goal.

$$F_o = \begin{bmatrix} \nabla_{q_x}\emptyset \\ \nabla_{q_y}\emptyset \end{bmatrix} \tag{19}$$

In this case the leader, and consequently the formation, moves toward the goal through a path determined by the gradient descent method. In this way, none tracking force is used in the control law of the leader, so that (1) becomes

$$u = F_O \tag{20}$$

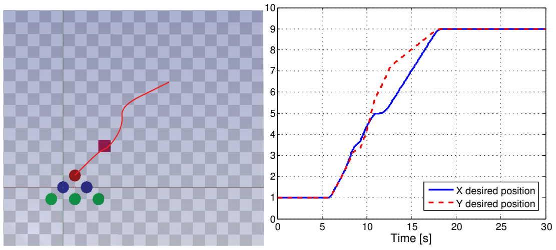
As can be noticed, F_O not only serves as an obstacle avoidance force in (2), in this case also conducts the agents toward the goal. Thus, the driving forces of the agents come not only from the formation control law but also from the obstacle avoidance force.

5. SIMULATION

In this section, the performance of the different strategies is assessed via simulations performed in the 20 Sim environment (20-Sim, Version 4.4, 2014).

In the simulation set $n = 5$ agents have been considered and, with the leader, all are distributed in three hierarchical levels.

The desired trajectory of the leader shown in Figure 10 starts at $\left(q_x, q_y\right) = (1,1)$ and finishes at $\left(q_x, q_y\right) = (9,9)$. As it can also be seen in Figure 10, the obstacle is a square of $(1m \times 1m)$ whose center is located at $\left(q_{xo}, q_{yo}\right) = (3.5,3.5)$. The formation, with the robots in their initial positions shown in Figure 10, is composed by the leader-Level 0 (red) and five agents distributed in Level 1–Level 2 (blue and green, respectively.)



(a) (b)

Figure 10: (a) Simulated scenario - desired path, obstacle and robots (b) Desired trajectory q_d.

Three different strategies are tested. In the first one the implemented obstacle avoidance technique is the virtual spring-damper approach seen in Section 4.1. In the second one, the avoidance force is implemented using the technique seen in Section 4.2. Finally, in the third strategy only the obstacle is modeled through a potential field, being the avoidance force the gradient of the potential function.

The trajectory tracking law of (1) is

$$F_T = m\ddot{q}_d + \mathrm{K_D}(\dot{q}_d - \dot{q}) + \mathrm{K_P}(q_d - q) \tag{21}$$

with K_P and K_D positive definite (diagonal) matrices, and $\dot{q}_d$ and q_d the desired velocity and position in the plane of the leader, respectively. In the second case, the tracking force is zero as explained in Section 4.2.

The mass of the robots is $m_i = 1.5\ [Kg]$. The parameters of the formation law presented in (4) are set as follows: for the first level $k_i = 20[N/m]$, $b_i = 9\ [Ns/m]$, for the second level $k_i = 200[N/m]$, $b_i = 20\ [Ns/m]$, and in both levels $l_{noi} = 2\ [m]$. The matrices of (21) are $K_P = [61, 0; 0,61]$, and $K_D = [193, 0; 0,193]$.

5.1. Strategy 1

The first simulation discussed is a coordinated tracking with obstacle avoidance strategy, implementing a virtual spring-damper approach as seen in Section 4.1. The parameters of (11) are $k_o = 337\ [N/m]$, $b_o = 5\ [Ns/m]$, $\mu_F = 0.1\ [N]$ and according to (12), $R_F = 15[m]$. In this way, as the robots are always located closer to the obstacle than this value, the virtual spring never attracts them.

In Figure 11, consecutive snapshots of the formation are shown. As it can be seen, the implemented approach keeps away the agents from the static obstacle.

In this case, as it is supposed that each robot measures the distance to the obstacle and computes the avoidance force (14), it is not necessary to specify the position of the obstacle in advance neither predefine the scenario workspace. This implies that, eventually, dynamics obstacles could be considered.

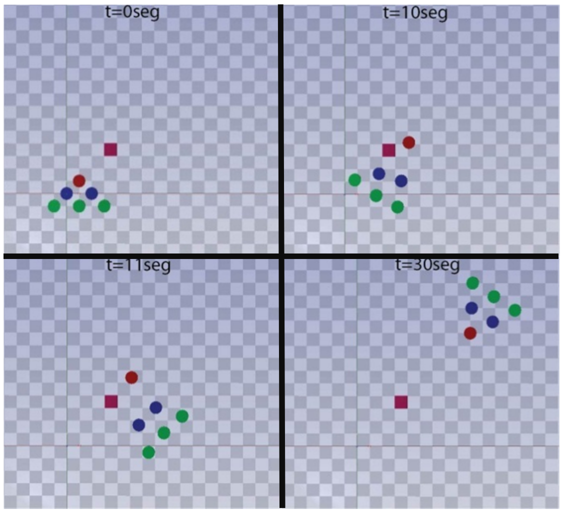
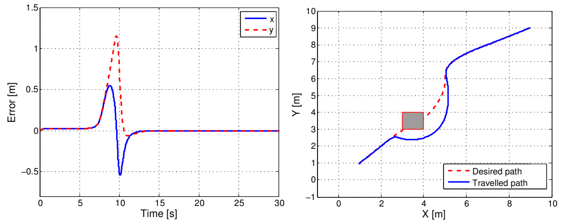


Figure 11: Strategy 1 - snapshots of the formation.

Figure 12-b depicts both, the desired and the real path travelled by the leader, while Figure 12-a shows the position error. Even if it cannot be distinguished in Figure 12-a, the final position error is not zero as, even if low, the virtual spring force is still presented when the leader arrives to its final position. This results in the agents remaining as far as possible from the obstacle, given to the formation the global orientation shown in Figure 11.



(a) (b)

Figure 12: (a) Tracking error – (b) Desired and travelled paths.

As it was mentioned above, one of the advantages of this strategy is that dynamic obstacles can be considered as it is not necessary to predefine the scenario. In that sense, under the same control strategy, a new scenario simulation was proposed in order to demonstrate this affirmation. In this scenario the static obstacle is replaced by a dynamic one that crosses the leader in his way to the goal. In this case, $\dot{l}_o$ is no longer (15), but

$$\dot{l}_o = cos\beta\,(\dot{q}_x - \dot{q}_{xo}) + sin\beta\,(\dot{q}_y - \dot{q}_{yo}) \qquad (22)$$

As it can be seen in Figure 13 all the agents avoid the obstacle, with the path done by leader depicted in Figure 14-b. The trajectory error of the leader, Figure 14-a, tends to zero with a permanent offset due to the force exerted by the virtual spring of the obstacle avoidance law.

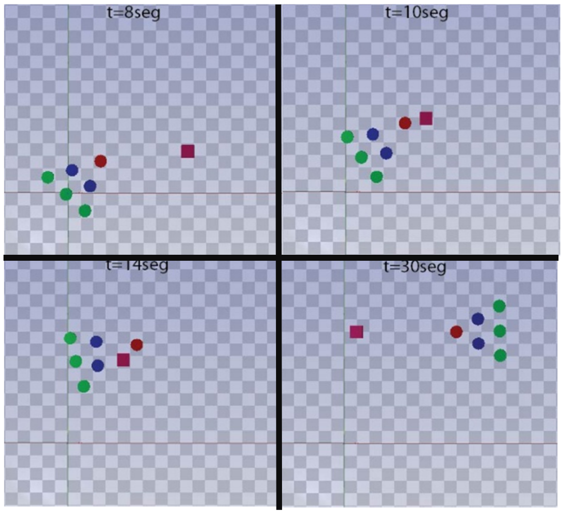
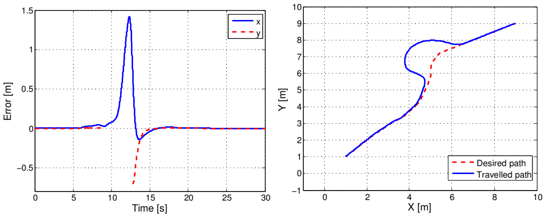


Figure 13: Strategy 1 with dynamic obstacle - snapshots of the formation.



(a) (b)

Figure 14: (a) Tracking error – (b) Desired and travelled paths.

Although the implemented control seems to be successful with both static and dynamic obstacle, the leader, as explained above, may get stuck into local minima, i.e. the sum of forces in (1) may remain zero with the leader in a final position different from the desired one. To avoid this, the strategy seen in Section 4.2 is tested.

5.2. Strategy 2

Next, the behavior of the six robots, each operating with the collision avoidance strategy seen in Section 4.2, is tested. The graph of the potential function that satisfies the Laplace's equation is shown in Figure 15-a where the obstacle and the goal position are represented with high and low potentials, respectively. The potential function is defined in a workspace of $(10m \times 10m)$. The algorithm's flow diagram is presented in the Appendix.

In this way, as is shown in the gradient trajectory map depicted in Figure 15-b, any trajectory initiated inside the workspace converges to the goal guaranteeing the absence of any local minima.

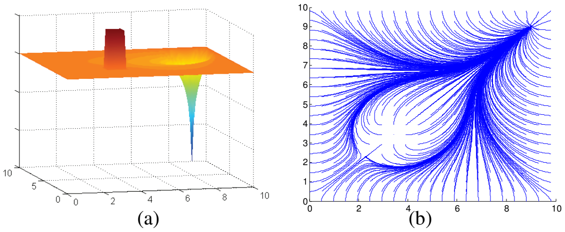

(a) (b)

Figure 15: (a) Potential function – (b) Field lines.

As mentioned above, in this case the control force F_o not only assures obstacle collision avoidance but also drives the leader toward the goal as shown in Figure 16. This means that the trajectory made by the leader depends on its initial position and $F_T = 0$ in (1). Furthermore, as this control law is also applied in (2), the agents are driven to the goal not just by its interactions but also by the avoidance force F_o.

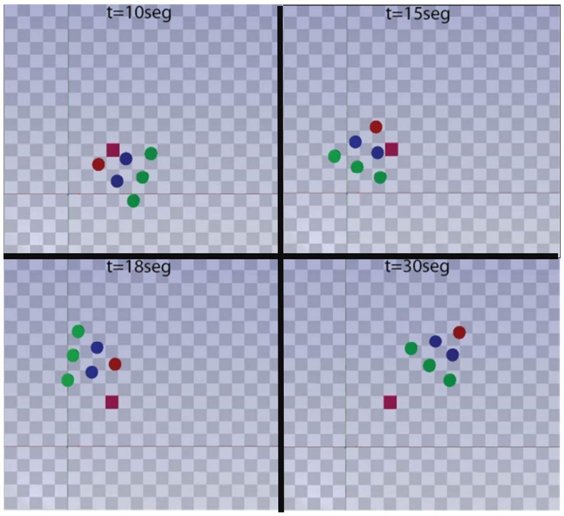


Figure 16: Strategy 2 - snapshots of the formation.

In the present case a previous study of the workspace is necessary in order to identify the position of the obstacle in advance. The final position error of the leader is zero and the global orientation of the formation remains determined by the control force F_o. However, desired trajectories can no longer be implemented through the control law F_T and the path realized by the leader depends on its initial position.

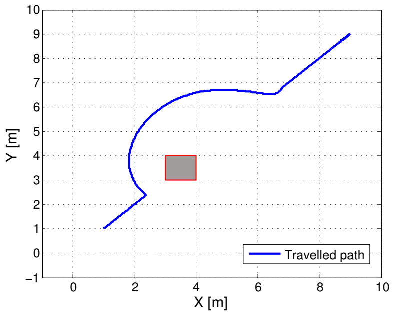


Figure 17: Path done by the leader.

To overcome this inconvenient, a third strategy is proposed in Section 5.3, where once again a potential field is created but now it only models the presence of the obstacles.

5.3. Strategy 3

Finally the behavior of the six robots modeling only the obstacle with a potential function is tested. The potential function that models the obstacle is depicted in Figure 18-a, and the corresponding field lines are shown in Figure 18-b.

In contrast to the previous case, the tracking trajectory law F_T is given as (21). As the obstacle avoidance force vanishes beyond an area of influence, the final position error of the leader is zero.

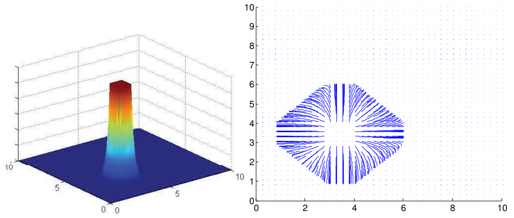

(a) (b)

Figure 18: (a) Potential function – (b) Field lines.

In the snapshots of the simulation depicted in Figure 19, it can be seen that even though the presence of the obstacle breaks the structure of the formation, the agents recover the desired interconnection once the obstacle is avoided and reach the goal with the desired formation. Notice that once the leader has reached the final position and the agents stand still, no control laws are applied to them and thus the formation can has any global orientation.

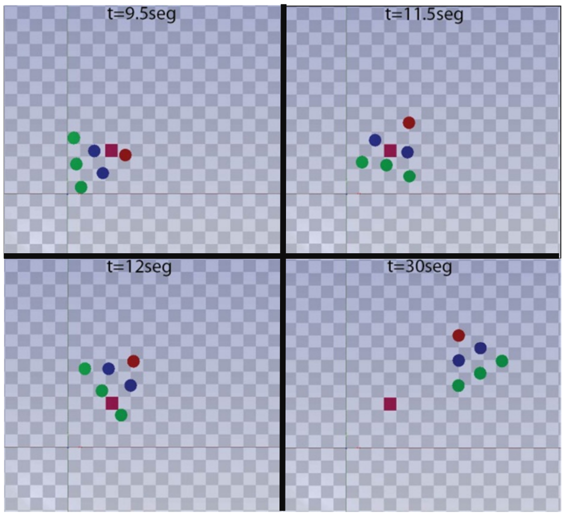
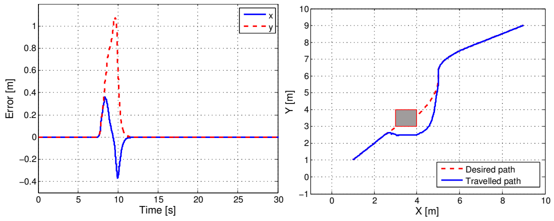


Figure 19: Strategy 3 - snapshots of the formation.

Figure 20-b depicts both, the desired and the real path made by the leader, while Figure 20-a shows the position error that is zero once the leader avoids the obstacle and reaches the final position goal.



(a) (b)

Figure 20: (a) Tracking error – (b) Desired and travelled paths.

Even if in this case a predefinition of the scenario has to be done and the leader may get stuck in local minima, unlike the other two strategies, a desired trajectory can be implemented and the final position error of the leader is zero.

Also it will be demonstrated, through simulation results and considering the same strategy, how the value of the virtual springs and dampers of the control law F_c of (2) affects the formation stability. Initially, whenever the hierarchy level increased, harder springs and dampers were considered. Supposing now that all the virtual dampers and springs are equal to $b_i = 9\,[Ns/m]$ and $k_i = 20[N/m]$, respectively, the formation movement of the robots can be appreciated in Figure 21. As it can be seen, even if the robots avoid the obstacle, the formation is no longer the desired one.

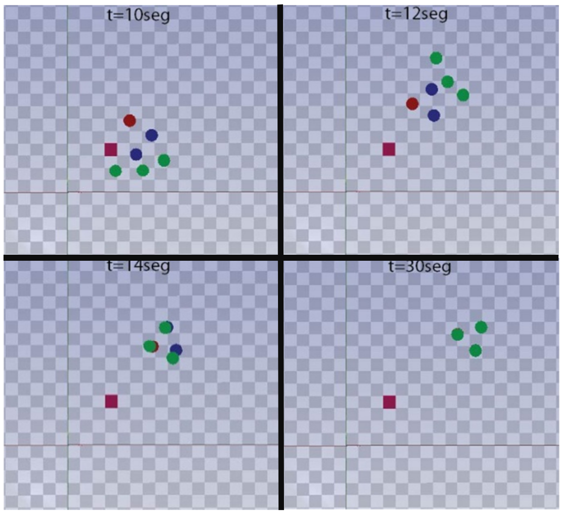


Figure 21: Strategy 3 - unstable formation - snapshots of the formation.

This happens because agents from high levels of hierarchy are subjected to higher rotational acceleration causing more instability in the formation. Even more, as the relation between the agents from different levels is unidirectional, agents from high levels does not sense position nor velocity from the lowest ones, so that the control laws do not interpret the changes in the formation as an error.

6. CONCLUSION

The problem of trajectory tracking and obstacle avoidance for a group of holonomic robots has been studied. The laws for formation control and obstacle avoidance have been obtained inspired on physical interpretations of the problem. Thus, the implemented control laws resulted in continues functions without commutation which implies less computation times.

Different variables have been analyzed in the simulations to study the advantages and disadvantages of each strategy: final position error of the leader, presence of local minima, formation orientation, trajectory tracking and previous analysis of the scenario.

Even if the formation was broken while it passes besides the obstacle, as shown in Section 5.3, the agents remain formed in the rest of the simulation sets and the control objectives have been achieved successfully.

Future directions include stability analysis of the formation considering different kinds of interconnections between the robots. Moreover, in order to improve the overall performance of the system, orientation control of both, the leader and the agents, will be taken into account.

ACKNOWLEDGMENTS

The authors wish to thank SeCyT-UNR for the support to this research through the financing of PID-UNR_ING502, as well as ANPCyC for the research project PICT 2017 No 3644.

APPENDIX

The following is the flow diagram of the iterative algorithm for the Laplace's Equation generation

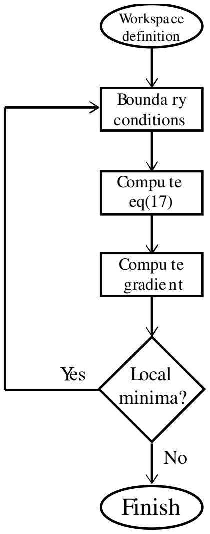


Figure 22: Laplace's equation generation.

The workspace is discretized in a mesh grid of $100x100$ equally spaced points. The definition of the Dirichlet boundary conditions consist in assigning to the function $u(q_{x_i}, q_{y_i})$ of (18) a high value to the borders of the workspace and the obstacle and a lower value to the goal position. This ensures that, after a determinate number of iterations, all the trajectories originated among the obstacle and the border finish in the goal position. Finally equation (18) and (19) are computed, in all the points of the workspace, to evaluate function ϕ and its gradient, respectively.

REFERENCES

A., G., & R., B. (1993). Real-time collision avoidance of manipulators with multiple redundancy. *Mechatronics, 3*(1), 89-106.

A., S., & H., H. (2018). Formation Control for a Fleet of Autonomous Ground Vehicles: A Survey. *Robotics, 7*(4), 67.

Alvarez D., A. J., & R., G. (2003). Online motion planning using Laplace potential fields. *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, *3*, págs. 3347-3352.

Burden RL., F. J., & AC, R. (1981). Numerical Analysis." 2nd. *Ed., Prindle, Weber, and Schmidt*, 458.

Cai C., Y. C., & Y., L. (2007). Collision avoidance in multi-robot systems. *2007 International Conference on Mechatronics and Automation*, (págs. 2795-2800).

Connolly C., B. J., & R., W. (1990). Path planning using Laplace's equation. *Proceedings., IEEE International Conference on Robotics and Automation*, (págs. 2102-2106).

Dauphin-Tanguy G., R. A., & C., S. (1999). Bond graph aided design of controlled systems. *Simulation Practice and Theory, 7*(5-6), 493-513.

H., R., & F., A. (2013). A decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots. *IEEE Transactions on Industrial Electronics, 61*(1), 347-354.

J., B., & Y., K. (1989). Real-time obstacle avoidance for fast mobile robots. *IEEE Transactions on systems, Man, and Cybernetics, 19*(5), 1179-1187.

Karnopp, D. C., Margolis, D. L., & Rosenberg, R. C. (2000). *System dynamics: modeling and simulation of mechatronic systems.* Wiley.

M., E., & X., H. (2001). Formation constrained multi-agent control.

Matoui F., B. B., & M., A. (2019). Distributed path planning of a multi-robot system based on the neighborhood artificial potential field approach. *SIMULATION, 95*(7), 637-657.

N., L., & E., F. (2001). Virtual leaders, artificial potentials and coordinated control of groups. *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, *3*, págs. 2968-2973.

O., K. (1986). Real-time obstacle avoidance for manipulators and mobile robots. En *Autonomous robot vehicles* (págs. 396-404). Springer.

Park M., J. J., & M., L. (2001). Obstacle avoidance for mobile robots using artificial potential field approach with simulated annealing. *ISIE 2001. 2001 IEEE International Symposium on Industrial Electronics Proceedings (Cat. No. 01TH8570)*, *3*, págs. 1530-1535.

R., M., & J., B. (2010). String instability in classes of linear time invariant formation control with limited communication range. *IEEE Transactions on Automatic Control, 55*(7), 1519-1530.

S., J. (2004). Virtual prototyping of bond graphs models for controller synthesis through energy and power shaping. *International Conference on Integrated Modeling and Analysis in Applied Control and Automation (IMAACA 2004).*

S., K., & R., M. (2013). Stability of two-dimensional linear systems with singularities on the stability boundary using LMIs. *IEEE Transactions on Automatic Control, 58*(10), 2579-2590.

Siciliano B., S. L., & G., O. (2010). *Robotics: modelling, planning and control.* Springer Science \& Business Media.

X., Y., & K., T. (1997). A wall-following method for escaping local minima in potential field based motion planning. *1997 8th International Conference on Advanced Robotics. Proceedings. ICAR'97*, (págs. 421-426).

Zhang Z., Z. L., & L., W. (2016). Leader-following consensus for linear and Lipschitz nonlinear multiagent systems with quantized communication. *IEEE Transactions on Cybernetics, 47*(8), 1970-1982.

Author's index

Ahmed 58
Benmoussa 11 32
Bensalah 48
Bessedik 24
Birame 24
Borutzky 1
Bouchareb 99
Crespo 38 124
Diaf 32
Djedidi 11 93
Djeziri 11 32
Donaire 115
Fukuoka 17
Glizer 65
Hemi 83
Ito 17
Junco 38 115 124
M'Sirdi 48 58 83 93 99
Mercantini 73
Naamane 58 24 48 83 93
Nacusse 38 124
Nemoto 17
Oida 17
Oudghiri 99
Saqli 99
Suzuki 17
Toguyéni 105
Tomassini 115
Turetsky 65