Lecture Notes in Electrical Engineering 174

Jean-Louis Ferrier • Alain Bernard
Oleg Gusikhin • Kurosh Madani
Editors

Informatics in Control, Automation and Robotics

8th International Conference, ICINCO 2011
Noordwijkerhout, The Netherlands, July 28-31, 2011
Revised Selected Papers

Springer

Lecture Notes in Electrical Engineering 174

For further volumes:
http://www.springer.com/series/7818

Jean-Louis Ferrier, Alain Bernard, Oleg Gusikhin,
and Kurosh Madani (Eds.)

Informatics in Control, Automation and Robotics

8th International Conference, ICINCO 2011
Noordwijkerhout, The Netherlands,
July 28–31, 2011 Revised Selected Papers

Springer

Editors

Jean-Louis Ferrier
University of Angers
France

Oleg Gusikhin
Ford Research & Adv. Engineering
USA

Alain Bernard
Ecole Centrale de Nantes
France

Kurosh Madani
University of PARIS-EST Créteil (UPEC)
France

ISSN 1876-1100 e-ISSN 1876-1119
ISBN 978-3-642-31352-3 e-ISBN 978-3-642-31353-0
DOI 10.1007/978-3-642-31353-0
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2012940860

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The present book includes extended and revised versions of a set of selected papers from the Eighth International Conference on Informatics in Control Automation and Robotics (ICINCO 2011), held in Noordwijkerhout, The Netherlands, from 28 to 31 July 2011. The conference was organized in four simultaneous tracks: Intelligent Control Systems and Optimization, Robotics and Automation, Systems Modelling, Signal Processing and Control and Industrial Engineering, Production and Management. The book is based on the same structure.

ICINCO 2011 received 322 paper submissions, from 52 countries in all continents. From these, after a blind review process, only 33 were accepted as full papers, of which 15 were selected for inclusion in this book, based on the classifications provided by the Program Committee. The selected papers reflect the interdisciplinary nature of the conference. The diversity of topics is an important feature of this conference, enabling an overall perception of several important scientific and technological trends. These high quality standards will be maintained and reinforced at ICINCO 2012, to be held in Rome, Italy, and in future editions of this conference.

Furthermore, ICINCO 2011 included 5 plenary keynote lectures given by Fumiya Iida (Institute of Robotics and Intelligent Systems, ETH Zurich, Switzerland), Jean-Marc Faure and Jean-Jacques Lesage (Ecole Normale Supérieure de Cachan, France), Okyay Kaynak (UNESCO Chair on Mechatronics, Bogazici University, Turkey), Bruno Siciliano (University of Naples Federico II, Italy) and Kurosh Madani (University of Paris-EST Créteil (UPEC), France). We would like to express our appreciation to all of them and in particular to those who took the time to contribute with a paper to this book.

On behalf of the conference organizing committee, we would like to thank all participants. First of all to the authors, whose quality work is the essence of the conference and to the members of the Program Committee, who helped us with their expertise and diligence in reviewing the papers. As we all know, producing a conference requires the effort of many individuals. We wish to thank also all the members of our organizing committee, whose work and commitment were invaluable.

March 2012

Jean-Louis Ferrier
Alain Bernard
Oleg Gusikhin
Kurosh Madani

Organization

Conference Chair

Jean-Louis Ferrier	University of Angers, France

Program Co-chairs

Alain Bernard	Ecole Centrale de Nantes, France
Oleg Gusikhin	Ford Research & Adv. Engineering, USA
Kurosh Madani	University of PARIS-EST Créteil (UPEC), France

Organizing Committee

Patrícia Alves	INSTICC, Portugal
Sérgio Brissos	INSTICC, Portugal
Marina Carvalho	INSTICC, Portugal
Helder Coelhas	INSTICC, Portugal
Vera Coelho	INSTICC, Portugal
Andreia Costa	INSTICC, Portugal
Patrícia Duarte	INSTICC, Portugal
Bruno Encarnação	INSTICC, Portugal
Liliana Medina	INSTICC, Portugal
Carla Mota	INSTICC, Portugal
Raquel Pedrosa	INSTICC, Portugal
Vitor Pedrosa	INSTICC, Portugal
Daniel Pereira	INSTICC, Portugal
Cláudia Pinto	INSTICC, Portugal
José Varela	INSTICC, Portugal
Pedro Varela	INSTICC, Portugal

Program Committee

Andrew Adamatzky, UK
Arvin Agah, USA
El-Houssaine Aghezzaf, Belgium
Arturo Hernandez Aguirre, Mexico
Hyo-Sung Ahn, Korea, Republic of
Hesham Alfares, Saudi Arabia
Frank Allgower, Germany
Job Van Amerongen, The Netherlands
Francesco Amigoni, Italy
Stefan Andrei, USA
Peter Arato, Hungary
Alejandro Hernandez Arieta, Switzerland
Carles Batlle Arnau, Spain
Marco Antonio Arteaga, Mexico
Vijanth Sagayan Asirvadam, Malaysia
Ronald Askin, USA
T. Asokan, India
Wudhichai Assawinchaichote, Thailand
Jacky Baltes, Canada
Ruth Bars, Hungary
Victor Becerra, UK
Laxmidhar Behera, India
Gerardo Beni, USA
Karsten Berns, Germany
Mauro Birattari, Belgium
Christian Blum, Spain
Jean-louis Boimond, France
Magnus Boman, Sweden
Thomas Braunl, Australia
Kevin Burn, UK
Dídac Busquets, Spain
Amaury Caballero, USA
Javier Fernandez de Canete, Spain
Giuseppe Carbone, Italy
Alessandro Casavola, Italy
Riccardo Cassinis, Italy
Ratchatin Chanchareon, Thailand
Antonio Chella, Italy
Peter C.Y. Chen, Singapore
Wen-Hua Chen, UK
Yangquan Chen, USA
Yuh-Min Chen, Taiwan
Graziano Chesi, China
Tsung-Che Chiang, Taiwan
Sung-Bae Cho, Korea, Republic of
Chengbin Chu, France
Carlos Coello Coello, Mexico
James M. Conrad, USA
José Boaventura Cunha, Portugal
Prithviraj (Raj) Dasgupta, USA
Mingcong Deng, Japan
Tony Dodd, UK
Frederick Ducatelle, Switzerland
Ashish Dutta, India
Petr Ekel, Brazil
Mohammed El-Abd, Kuwait
Ali Eydgahi, USA
Simon G. Fabri, Malta
Jean-Marc Faure, France
David Fernández-Llorca, Spain
Paolo Fiorini, Italy
Heinz Frank, Germany
Georg Frey, Germany
Manel Frigola, Spain
Ponnambalam S.G., Malaysia
Ryszard Gessing, Poland
Paulo Gil, Portugal
Maria Gini, USA
Alessandro Giua, Italy
Luis Gomes, Portugal
Bhaskaran Gopalakrishnan, USA
Ole-Christoffer Granmo, Norway
Frans C.A. Groen, The Netherlands
Da-Wei Gu, UK
Kevin Guelton, France
José J. Guerrero, Spain
Oleg Gusikhin, USA
Thomas Gustafsson, Sweden
Quang Phuc Ha, Australia
Maki K. Habib, Egypt
Juergen Hahn, USA
Wolfgang Halang, Germany
John Hallam, Denmark
Riad Hammoud, USA
Qing-Long Han, Australia
Jennifer Harding, UK

Inman Harvey, UK
Dominik Henrich, Germany
Suranga Hettiarachchi, USA
Victor Hinostroza, Mexico
Kaoru Hirota, Japan
M. Ani Hsieh, USA
Guoqiang Hu, USA
Biao Huang, Canada
Fumiya Iida, Switzerland
Giovanni Indiveri, Italy
Mirjana Ivanovic, Serbia
Sarangapani Jagannathan, USA
Myong K. Jeong, USA
Ivan Kalaykov, Sweden
Michail Kalogiannakis, Greece
Mansour Karkoub, Qatar
Dimitrios Karras, Greece
DaeEun Kim, Korea, Republic of
Jonghwa Kim, Germany
Ashok K. Kochhar, UK
Israel Koren, USA
Krzysztof Kozlowski, Poland
H.K. Lam, UK
Kauko Leiviskä, Finland
Tsai-Yen Li, Taiwan
Yangmin Li, China
Gordon Lightbody, Ireland
Huei-Yung Lin, Taiwan
Guoping Liu, UK
Honghai Liu, UK
Gonzalo Lopez, Spain
Iuan-Yuan Lu, Taiwan
Edwin Lughofer, Austria
Martin Lukac, Japan
José Tenreiro Machado, Portugal
Anthony Maciejewski, USA
Kurosh Madani, France
Chitralekha Mahanta, India
Bruno Maione, Italy
Frederic Maire, Australia
Om Malik, Canada
Hervé Marchand, France
Philippe Martinet, France
Sonia Martínez, USA
Aníbal Matos, Portugal
Seán McLoone, Ireland
Carlo Menon, Canada
Sanya Mitaim, Thailand
António Paulo Moreira, Portugal
Vladimir Mostyn, Czech Republic
Saeid Nahavandi, Australia
Andreas Nearchou, Greece
Maria Neves, Portugal
Hendrik Nijmeijer, The Netherlands
H.R.B. Orlande, Brazil
Manuel Ortigueira, Portugal
Christos Panayiotou, Cyprus
Michel Parent, France
Gwi Tae Park, Korea, Republic of
Igor Paromtchik, France
Pierre Payeur, Canada
Carlos Eduardo Pereira, Brazil
D.T. Pham, UK
Jeff Pieper, Canada
Angel P. Del Pobil, Spain
Marie-Noëlle Pons, France
Vittal Prabhu, USA
Raul Marin Prades, Spain
José Ragot, France
A. Fernando Ribeiro, Portugal
Mihailo Ristic, UK
Rodney Roberts, USA
Eduardo Rocon, Spain
Juha Röning, Finland
Mehdi Roopaei, Iran, Islamic Republic of
Agostinho Rosa, Portugal
João Mauricio Rosário, Brazil
Carlos Sagüés, Spain
Mehmet Sahinkaya, UK
Priti Srinivas Sajja, India
Antonio Sala, Spain
Abdel-badeeh Mohamed Salem, Egypt
Marcello Sanguineti, Italy
Elena De Santis, Italy
Srikanth Saripalli, USA
Nilanjan Sarkar, USA
Jurek Sasiadek, Canada
Daniel Sbarbaro, Chile
Klaus Schilling, Germany
Carla Seatzu, Italy

Ali Selamat, Malaysia
Michael Short, UK
Dan Simon, USA
Olivier Simonin, France
Joaquin Sitte, New Zealand
Adam Slowik, Poland
Andrzej Sluzek, Singapore
Safeeullah Soomro, Saudi Arabia
Stefano Squartini, Italy
Burkhard Stadlmann, Austria
Sergiu Stan, Romania
A. Frank Van Der Stappen, The Netherlands
J.D. Stigter, The Netherlands
Olaf Stursberg, Germany
Raúl Suárez, Spain
Kamesh Subbarao, USA
Kazuya Takeda, Japan
Choon Yik Tang, USA
József K. Tar, Hungary
Daniel Thalmann, Singapore
Haydn A. Thompson, UK
Gui Yun Tian, UK
N.G. Tsagarakis, Italy
Antonios Tsourdos, UK
Nikos Tsourveloudis, Greece
Masaru Uchiyama, Japan
Dariusz Ucinski, Poland
Angel Valera, Spain
Bram Vanderborght, Belgium
Ramiro Velazquez, Mexico
Damir Vrancic, Slovenia
Eric Wade, USA
Bernardo Wagner, Germany
James Whidborne, UK
Sangchul Won, Korea, Republic of
Qishi Wu, USA
Xun W. Xu, New Zealand
Bin Yao, USA
Janan Zaytoon, France
Du Zhang, USA
K. Zhou, USA
Loredana Zollo, Italy
Argyrios Zolotas, UK

Auxiliary Reviewers

Lounis Adouane, France
Jonathan Courbon, France
Felix Felgner, Germany
Tom Forster, UK
Giorgio Gnecco, Italy
Dani Juricic, Slovenia
Ganesh Kumar, USA
Shane Lynn, Ireland
Sinuhe Martinez-Martinez, France
Konstantinos Michail, Cyprus
Luis Palma, Portugal
Alessandro Rizzo, Italy
Xuan Zhang, Norway

Invited Speakers

Fumiya Iida	Institute of Robotics and Intelligent Systems, ETH Zurich, Switzerland
Bruno Siciliano	University of Naples Federico II, Italy
Jay Lee	University of Cincinnati, USA
Okyay Kaynak	UNESCO Chair on Mechatronics, Bogazici University, Turkey
Jean-Marc Faure	École Normale Supérieure de Cachan, France
Jean-Jacques Lesage	École Normale Supérieure de Cachan, France

Contents

Invited Paper

Part I: Intelligent Control Systems and Optimization

Part II: Robotics and Automation

Part III: Signal Processing, Sensors, Systems Modeling and Control

Part IV: Industrial Engineering, Production and Management

Invited Paper

Perception and Cognition: Two Foremost Ingredients toward Autonomous Intelligent Robots

Kurosh Madani

Images, Signals and Intelligence Systems Laboratory (LISSI / EA 3956)
University PARIS-EST Creteil (UPEC), Senart-FB Institute of Technology
Bât. A, Av. Pierre Point, F-77127 Lieusaint, France
madani@univ-paris12.fr

Abstract. Inspired by early-ages human's skills developments, the present paper accosts the robots' intelligence from a different slant directing the attention to both "cognitive" and "perceptual" abilities. the machine's (robot's) shrewdness is constructed on the basis of a Multi-level cognitive concept attempting to handle complex artificial behaviors. The intended complex behavior is the autonomous discovering of objects by robot exploring an unknown environment.

1 Introduction and Problem Stating

The term "cognition", refers to the ability for the processing of information applying knowledge. If the word "cognition" has been and continues to be used within quite a large number of different contexts, in the field of computer science, it often intends artificial intellectual activities and processes relating "machine learning" and accomplishment of knowledge-based "intelligent" artificial functions. However, cognitive process of "knowledge construction" (and in more general way "intelligence") requires ability to perceive information from surrounding environment. Thus, "cognition" and "perception" remain inseparable ingredients toward machines' intelligence and thus toward machines' (robots', etc…) autonomy.

Concerning most of works relating modern robotics, and especially humanoid robots, have concerned either the design of controllers controlling different devices of such machines ([1] and [2]) or the navigation aspects of such robots ([4] to [6]). In the same way, major part of works dealing with human-like or in more general terms intelligent behavior is connected with abstract tasks, as those relating reasoning inference, interactive deduction mechanisms, etc….([7] to [11]) Inspired by early-ages human's skills developments ([12] to [16]) and especially human's early ages walking ([17] to [20]), the present work accosts the robots' intelligence from a different slant directing the attention on both "cognitive" and "perceptual" traits.

Combining cognitive and perceptual abilities, the machine's (robot's) shrewdness is constructed on the basis of two kinds of functions: "Unconscious Cognitive Functions" (UCF) and "Conscious Cognitive Functions" (CCF). We identify UCF as activities belonging to the "instinctive" cognition level handling reflexive abilities. Beside this, we distinguish CCF as functions belonging to the "intentional" cognition level handling thought-out abilities. The two above-mentioned kinds of functions

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 3–18.
springerlink.com

have been used as basis of a Multi-level cognitive concept attempting to handle complex artificial behaviors. The intended complex behavior is the autonomous discovering of objects by robot exploring an unknown environment. The present article will not itemize the motion related aspect which has been widely presented, analyzed, discussed and validated (on different examples) in [21]. Regarding perceptual skill, it is developed on the basis of artificial vision and "salient" object detection. The paper will center this foremost skill.

The paper is organized in six sections. The next section briefly introduces the Multi-level cognitive concept. Section 3 describes the general structure of a cognitive function. In section 4, the suggested Motion-Perception control strategy is presented. Validation's results, obtained from implementation on a real humanoid-like robot, are reported in this section. Finally, the last section concludes the paper.

2 Brief Overview of Multi-level Cognitive Concept

The concept considers a process (mainly a complex process) as a multi-model structure where involved component (models), constructed as a result of Machine Learning (ML), handle two categories of operational levels: reflexive and intentional [21]. So, ML and related techniques play a central role in this concept and the issued architectures. According to what has been mentioned in introductory section, two kinds of functions, so-called UCF and CCF, will build-up functional elements ruling the task or complex behavior. Figure 1 illustrates the bloc diagram of the proposed cognitive conception. As it is noticeable from this figure, within the proposed concept, the overall architecture is obtained by building up cognitive layers (levels) corresponding to different skills fashioning the complex task. As well UCF as CCF enclose a number of "Elementary Functions" (EF). Within such a scheme, a cognitive layer may fulfil a skill either independently from other layers (typically, the case of unconscious cognitive levels) or using one or several talents developed by other layers (characteristically, the case of conscious cognitive levels) [21].

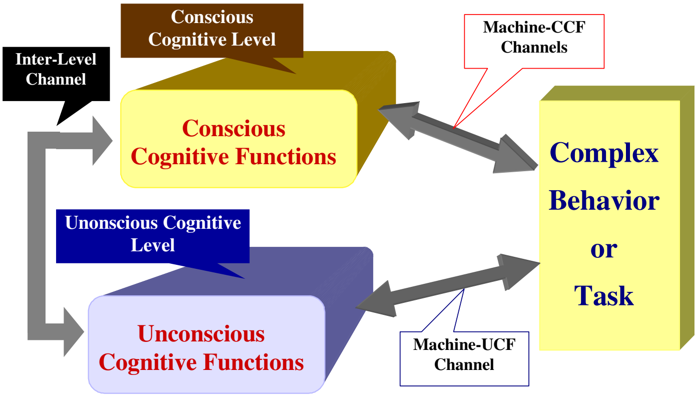

Conscious Cognitive Level
Inter-Level Channel
Machine-CCF Channels
Conscious Cognitive Functions
Complex Behavior or Task
Unonscious Cognitive Level
Unconscious Cognitive Functions
Machine-UCF Channel

Fig. 1. Robot's coordinates described by a triplet as P(x , y, θ)

The first key-advantage of conceptualizing the problem within such incline is to detach the build-up of artificial complex behaviour's modelling from the type of robot (machine): as the early-age human's abilities development which in its global achievement doesn't depend on kind of "baby" (early ages human walking development is a typical example). The second chief-benefit of the concept is that the issued artificial structures are "Machine Learning" (Artificial Neural Networks, Fuzzy logic, reinforcement learning, etc…) based, taking advantage from "learning" capacity and "generalization" propensity of such models: allowing a precious potential to deal with high dimensionality, nonlinearity and empirical (non-analytical) proprioceptive or exteroceptive information.

3 Cognitive Function

As it has been mentioned-above, a cognitive function (CUF or CCF) is constructed by a number of EF. EF is defined as a function (learning-based or conventional) realizing an operational aptitude composing (necessary for) the skill accomplished by concerned cognitive function. An EF is composed of "Elementary Components" (EC). An EC is the lowest level component (module, transfer function, etc…) realizing some elementary aptitude contributing in EF's operational aptitude. Two kinds of EC could be defined (identified): the first corresponding to elementary action that we call "Action Elementary Component" (AEC) and the second corresponding to elementary decision that we call "Decision Elementary Component" (DEC). An EF may include one or both two kinds of the above-defined EC. Finally, an EF may include one or several EC. If Fig. 2 gives the general structure of a cognitive function, it is pertinent to notice that there is any limitation to include an EC (individually) in composition (e.g. in structure) of a cognitive function. In other words, when it may be necessary, an EC could play the role of an EF.

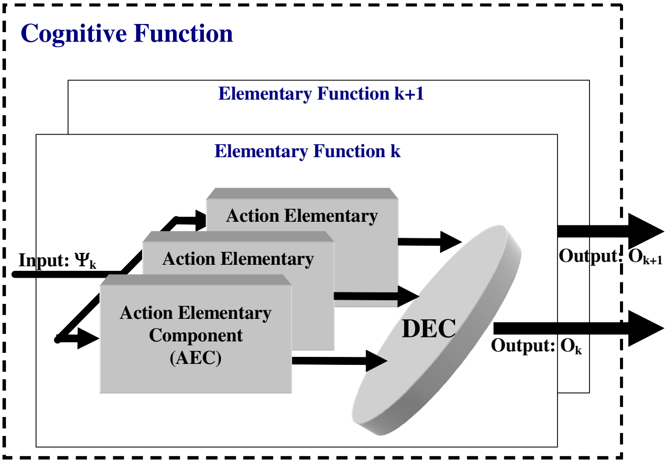


Fig. 2. General Bloc-diagram of a cognitive function

Supposing that a given cognitive function (either conscious or unconscious) includes K (K ∈ N where N represents the "Natural numbers' ensemble) elementary functions, considering the k-th EF (with k ∈ N and k ≤K) composing this cognitive function, we define the following notations:

Ψ_k the input of k-th EF: $\Psi_k = [\psi_1,\ldots,\psi_j,\ldots,\psi_M]^T$ where ψ_j represents the input component of the j-th EC of this EF, j≤M and M the total number of elementary components composing this EF

O_k the output of k-th EF.

o_j the output of the j-th EC of the k-th EF, with j≤M and M the total number of elementary components composing the k-th EF.

F_k (.) the skill performed by the k-th EF.

$f_j^A(\cdot)$ the function (transformation, etc…) performed by j-th AEC.

$f^D(\cdot)$ the decision (matching, rule, etc…) performed by DEC.

Within the above-defined notation, the output of k-th EF is formalized as shown in (1) with o_j given by (2). In a general case, the output of an EC may also depend to some internal (specific) parameters particular to that EC [21].

$$O_k = F_k(\Psi_k) = f^D(\Psi_k, o_1, \ldots, o_j, \ldots, o_M) \tag{1}$$

$$o_j = f_j^A(\psi_j) \tag{2}$$

4 Motion-Perception Architecture for Cognitive Robots' Control

Based on the aforementioned cognitive concept, the control scheme of a robot could be considered within the frame of "Motion-Perception" (MP) based architecture. Consequently, as well robot's motions as its perception of the environment are obtained combining UCF and CCF. Robot's sway is achieved combining unconscious and conscious cognitive motion functions (UCMF and CCMF respectively). In the same way, essentially based on vision, robot's perceptual ability is constructed combining unconscious and conscious cognitive visual functions (UCVF and CCVF respectively). Fig. 3 shows such an MP based robot's cognitive control scheme.

It is pertinent to notice that the proposed control scheme takes advantage from some universality, conceptualizing the build-up of both robot's motion and perception abilities independently from the type of robot. It is also relevant to emphasize that the proposed cognitive scheme links the behavior's control construction to perception constructing the robot's action from and with perceptual data and interaction with the context. This way of doing lays the robot's way of doing (knowledge construction) to the Human's way of learning and knowledge's construction: humans or animals learn and construct the knowledge by interacting with the environment. In other words, "smart" systems operate using "awareness" about its environment.

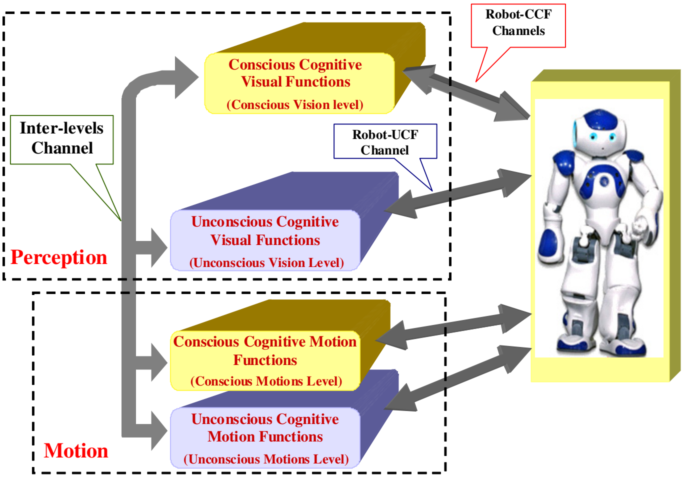

Robot-CCF Channels
Conscious Cognitive Visual Functions
(Conscious Vision level)
Inter-levels Channel
Robot-UCF Channel
Unconscious Cognitive Visual Functions
(Unconscious Vision Level)
Perception
Conscious Cognitive Motion Functions
(Conscious Motions Level)
Unconscious Cognitive Motion Functions
(Unconscious Motions Level)
Motion

Fig. 3. Bloc-diagram of Motion-perception based robot's cognitive control scheme

4.1 Visual Perception

As it has been argued in introductory section, the perceptual function (and thus its design) is a major ingredient in proffering robot's autonomy. If the question of how humans learn, represent, and recognize objects under a wide variety of viewing conditions is still a great challenge to both neurophysiology and cognitive researchers [22], a number of works relating the human's early-ages cognitive walking ability's construction process highlight a number of key mechanisms. As shows clinical experiments (as those shown by [23]), one them is the strong linkage between visual and motor mechanisms. This corroborates the pertinence of the suggested cognitive MP based scheme. Beside this, [24] and [25] show that apart of shaping the way we (human) **see** the world by bringing our attention to visually important objects first, the visual attention mechanism plays also one of the key roles in human infants learning of the encountered objects. Thus, it appears appropriate to draw inspiration from studies on human infants and robots learning by demonstration. The present sub-section focuses on the robot's perception skill: a learning process based visual perception.

Making an intelligent system to perceive the environment in which it evolves and constructing the knowledge, by learning unknown objects present in that environment, makes appear a clear need relating the ability to select from the overwhelming flow of sensory information only the pertinent ones. This foremost ability is known as "visual saliency" sometimes called in literature as visual attention, unpredictability or surprise. It is described as a perceptual quality that makes a part of image stand out relative to the rest of the image and to capture attention of observer ([26]). It may be generalized, that it is the saliency (in terms of motion, colors, etc.) that lets the pertinent information "stand-out" from the context [27]. We argue that in this context visual saliency may be helpful to enable unsupervised extraction and subsequent learning of a previously unknown object by a machine: in other words, proffering to the machine (robot) the awareness about its environment.

Referring to perception bloc of Fig.3, the visual perception architecture is composed of one unconscious visual including UCVF and one conscious visual level containing CCVF. Unconscious visual level handles reflexive visual tasks, namely the pre-processing of acquired images, the salient objects detection and the detected salient objects' storage. If the pre-processing could appear as an independent UCVF, it also may be an EF of one of UCVF composing the unconscious visual level. In this second way of organizing the unconscious visual level, the UCVF including the pre-processing task will deliver the pre-processing results (as those relating image's segmentation, different extracted features, etc…) as well to other UCVF composing the unconscious level as to those CCVF of conscious level which need the aforementioned results, using the inter-levels channel. Conscious visual level conducts intentional visual tasks, namely the objects' learning (including learning detected salient objects), the knowledge construction by carrying out an intentional storage (in unconscious visual level) of new detected salient objects, the detected salient objects recognition in robot's surrounding environment (those already known and the visual target (recognized salient object) tracking allowing the robot's self-orientation and motion toward a desired recognized salient object. Consequently, the conscious level communicates (e.g. delivers the outputs of concerned CCVF) as well to unconscious level (e.g. to the concerned UCVF) as to unconscious motion and conscious motion levels (e.g. the bloc in MP based robot's cognitive control scheme in charge of robot's motions).

Next sub-sections are devoted to description of two principle cognitive visual functions. The first one will detail the main UCVF, called "salient vision" which allows robot to self-discover (automatically detect) pertinent objects within the surrounding environment. While, the second sub-section will spell out one of the core CCVF, called "visual intention" which proffers the robot artificial visual intention ability and allow it to construct the knowledge about the surrounding environment.

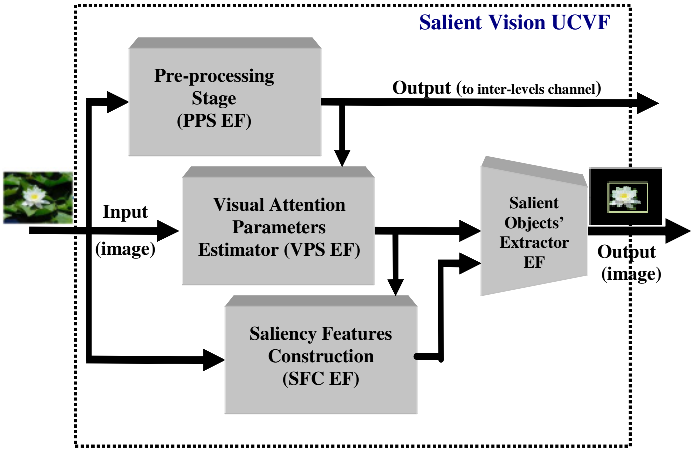

Salient Vision UCVF
Pre-processing Stage (PPS EF)
Output (to inter-levels channel)
Input (image)
Visual Attention Parameters Estimator (VPS EF)
Salient Objects' Extractor EF
Output (image)
Saliency Features Construction (SFC EF)

Fig. 4. Bloc-diagram of Salient Vision UCVF, handling the automated detection of salient objects

4.2 Salient Vision UCVF and Salient Objects Detection

The bloc-diagram detailing the structure of "Salient Vision" UCVF is given by Fig. 4. As it is visible from this figure, the "Salient Vision" UCVF includes also the pre-processing stage (defined as one of its constituting EF), meaning that this UCVC handles the image's segmentation and common image's features' extraction tasks, delivering the issued results as well to other UCVF as to conscious visual level. Beside this EF, it includes three other EF: the "Visual Attention Parameters' Estimator" (VAPE) EF, the "Salient Features' Construction" (SFC) EF and the "Salient Objects' Detection" (SOD) EF. This last EF plays the role of a Decision-like elementary component, implemented as an independent EF.

4.2.1 Visual Attention Parameters' Estimation Elementary Function

The VAPE elementary function determines what could be assimilated to some kind of "visual attention degree". Computed on the basis of pre-processing bloc's issued results and controlling local salient features, visual attention parameter p constructs a top-down control of the attention and of the sensitivity of the feature in scale space. High value of p (resulting in a large sliding window size) with respect to the image's size will make the local saliency feature more sensitive to large objects. In the same way, low values of p allow focusing the visual attention to smaller objects and details. The value of visual attention parameter p can be hard-set to a fixed value based on a heuristic according to [28]. However, as different images usually present salient objects in different scales, this way of doing will limit the performance of the system. Thus a new automated cognitive estimation of the parameter p has been designed. The estimation is based on the one hand, on calculation (inspired from the work presented in [29]) of the histogram of segment sizes from the input image and on the other hand, on using of an Artificial Neural Network (ANN). The ANN receives (as input) the feature vector issued from the above-mentioned histogram and provides the sliding window value. The weights of the neural network are adapted in training stage using a genetic algorithm.

To obtain the feature vector, the input image is segmented into n segments $(S_1, S_2, \cdots, S_n)$. For each one of the found segments S_i (where $S_i \in \{S_1, S_2, \cdots, S_n\}$), its size $|S_i|$ (measured in number of pixels) is divided by the overall image size $|I|$. An absolute-histogram H_{SA} of segments' sizes is constructed according to (3), avoiding leading to a too sparse histogram. This ensures that the first histogram bin contains the number of segments with area larger that 1/10 of the image size, the second contains segments from 1/10 to 1/100 of the image size etc. For practical reasons we use a 4-bin histogram. Then this absolute-histogram leads to a relative-histogram H_{SR} computed according to the relation (4).

$$H_{SA}(i) = \sum_{j=1}^{n} \begin{cases} 1 & if\, 10^{i-1} \le \left(\frac{|S_i|}{|I|} \right) \le 10^{i} \\ 0 & otherwise \end{cases} \tag{3}$$

$$H_{SR}(i) = \frac{H_{SA}(i)}{\sum_j H_{SA}(j)} \tag{4}$$

The core of the proposed visual attention parameter estimator is a fully connected three-layer feed-forward MLP-like ANN, with a sigmoidal activation function, including: 4 input nodes, 3 hidden neurons and 1 output neuron. The four input nodes are connected each to its respective bin from the H_{SR} histogram. The value of the output node, belonging to the continuous interval $[0,1]$, could be interpreted as the ratio of the estimated sliding window's size p and the long side's size of the image. The ANN is trained making use of a genetic algorithm described in [30]. Each organism in the population consists of a genome representing an array of floating point numbers whose length corresponds with the number of weights in MLP. To calculate the fitness of each organism, the MLP weights are set according to its current genome. Once visual attention parameter p is available (according to the MLP output) saliency is computed over the image and salient objects are extracted. The result is compared with ground truth and the precision, the recall and the F-ratio (representing the overall quality of the extraction) are calculated (according to [31] and using the measures proposed in the same work to evaluate quantitatively the salient object extraction). The F-ratio is then used as the measure of fitness. In each generation, the elitism rule is used to explicitly preserve the best solution found so far. Organisms are mutated with 5% of probability. As learning data-set, we use 10% of the MSRA-B data-set (described in [31]). The resting 90% of the above-indicated data-set has been used for validation.

4.2.2 Salient Features' Construction Elementary Function

The Salient Features' Construction EF performs two kind of features (both used for salient objects' detection). The first kind is global saliency features and the second local saliency features.

Global saliency features capture global properties of image in terms of distribution of colors. The global saliency is obtained combining "intensity saliency" and the "chromatic saliency". Intensity saliency $M_l(x)$, given by relation (5), is defined as Euclidean distance of intensity I to the mean of the entire image. Index l stands for intensity channel of the image, $I_{\mu l}$ is the average intensity of the channel. In the same way, chromatic saliency, given by relation (6), is defined as Euclidean distance of azimuth and zenith components' intensities (e.g. azimuth ϕ and zenith θ respectively) to their means ($I_{\mu\phi}$ and $I_{\mu\theta}$ respectively) in the entire image. Term *(x)* denotes coordinates of a given pixel on the image.

$$M_l(x) = \left\| I_{\mu l} - I_l(x) \right\| \tag{5}$$

$$M_{\phi\theta}(x) = \sqrt{\left(I_{\mu\phi} - I_{\phi}(x)\right)^2 + \left(I_{\mu\theta} - I_{\theta}(x)\right)^2} \tag{6}$$

The global saliency map $M(x)$, given by relation (7) is a hybrid result of combination of maps resulted from (1) and (2) according to logistic sigmoid blending function. Blending of the two saliency maps together is driven by a function of color saturation C of each pixel. It is calculated from RGB color model for each pixel as pseudo-norm, given by $C = Max[R,G,B] - Min[R,G,B]$. When C is low, importance is given to intensity saliency. When C is high, chromatic saliency is emphasized.

$$M(x) = \frac{1}{1-e^{-C}} M_{\phi\theta}(x) + \left(1 - \frac{1}{1+e^{-C}}\right) M_l(x) \tag{7}$$

The global saliency (and related features) captures the visual saliency with respect to the colors. However, in real cases, the object's visual saliency may also consist in its particular shape or texture, distinct to its surroundings, either beside or rather than simply in its color. To capture this aspect of visual saliency, a local feature over the image is determined. Inspired from a similar kind of feature introduced in [31], the local saliency it is a centre-surround difference of histograms, which was loosely. The idea relating the local saliency is to go through the entire image and to compare the content of a sliding window with its surroundings to determine, how similar the two are. If similarity is low, it may be a sign of a salient region within the sliding window.

To formalize this idea leading local saliency features, let us have a sliding window P of size p, centred over pixel *(x)*. Define a (centre) histogram H_C of pixel intensities inside it. Then let us define a (surround) histogram H_S as histogram of intensities in a window Q surrounding P in a manner that the area of $(Q-P) = p^2$. The centre-surround feature $d(x)$ is then given as (8) over all histogram bins (i).

$$d(x) = \sum_i \frac{|H_C(i) - H_S(i)|}{p^2} \tag{8}$$

Resulting from computation of the $d(x)$throughout all the l, ϕ and θ channels, the centre-surround saliency $D(x)$ on a given position *(x)* is defined according to (9). Similarly to (7), a logistic sigmoid blending function has been used to combine chromaticity and intensity in order to improve the performance of this feature on images with mixed achromatic and chromatic content. However, here the color saturation C refers to average saturation of the content of the sliding window P.

$$D(x) = \frac{1}{1-e^{-C}} d_l(x) + \left(1 - \frac{1}{1+e^{-C}}\right) Max\left(d_\phi(x), d_\theta(x)\right) \tag{9}$$

Fig.5 shows examples of global and local saliency features extracted from two images. In the first image the global salient feature (upper image of column b) is enough to track salient objects, while for the second, where the salient object (leopard) is partially available, chromatic saliency is not enough to extract the object.

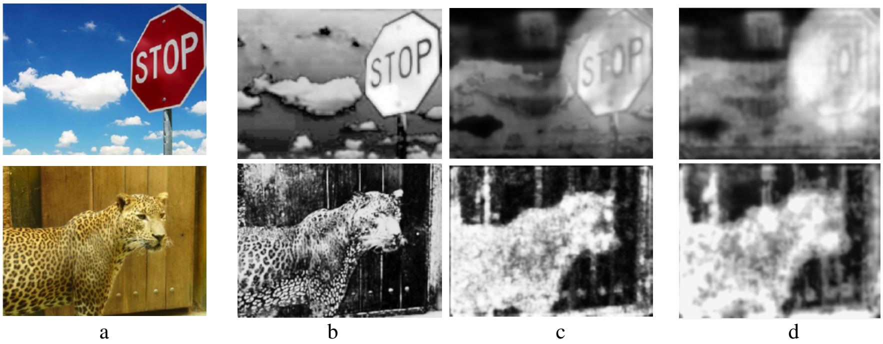


Fig. 5. Examples of global and local saliency features: original image (a), global saliency map (b), local saliency map (c) and final saliency map

4.2.3 Salient Objects' Detection Elementary Function

Salient object's detection EF acts as the last step of saliency map calculation and salient objects' detection. The extracted global and local salient features (e.g. $M(x)$ and $D(x)$, respectively) are combined together by application of (10), resulting in final saliency map $M_{final}(x)$, which is then smoothed by Gaussian filter. The upper part of the condition in (10) describes a particular case, where a part of image consists of a color, that is not considered salient (i.e. pixels with low $M(x)$ measure) but which is distinct to the surroundings by virtue of its shape. The final saliency map samples are shown on the column d of Fig. 5

$$M_{final}(x) = \begin{cases} D(x) & \text{if } M(x) < D(x) \\ \sqrt{M(x)\,D(x)} & \text{otherwise} \end{cases} \tag{10}$$

Accordingly to segmentation and detection algorithms described in [29] and [32], the segmentation splits an image into a set of chromatically coherent regions. Objects present on the scene are composed of one or multiple such segments. For visually salient objects, the segments forming them should cover areas of saliency map with high overall saliency, while visually unimportant objects and background should have this measure comparatively low. Conformably to [32], input image is thus segmented into connected subsets of pixels or segments $(S_1, S_2, \cdots, S_n)$. For each one of the found segments S_i (where $S_i \in \{S_1, S_2, \cdots, S_n\}$), its average saliency $\overline{S_i}$ and variance (of saliency values) $Var(S_i)$ are computed over the final saliency map $M_{final}(x)$. All the pixel values $p(x, y) \in S_i$ p (z, y) of the segment are then set following (11), where $\tau_{\overline{S_i}}$ and τ_{Var} are thresholds for average saliency and its variance respectively. The result is a binary map containing a set of connected components $C = \{C_1, C_2, \cdots, C_n\}$ formed by adjacent segments S_i evaluated by (11) as binary value "1". To remove of noise, a membership condition is imposed that any $C_i \in C$ has its area larger than a given threshold. Finally, the binary map is projected on the original image leading to a result which is parts (areas) of the original image containing its salient objects. [32] and [33] give different values for aforementioned parameters and thresholds.

$$p(x,y)=\begin{cases}1 & if\ \overline{S}_i > \tau_{\overline{S}_i}\ and\ Var(S_i) > \tau_{Var} \\ 0 & \qquad otherwise\end{cases} \tag{11}$$

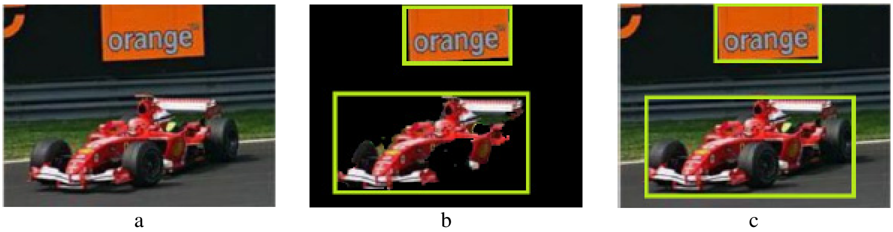


Fig. 6. Examples of salient object detection: input image (a), detected salient objects (b) and ground truth salient objects (c)

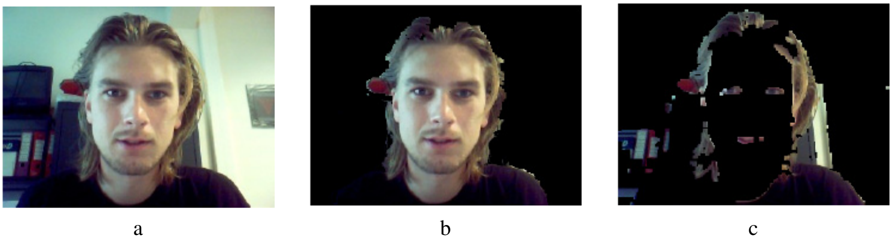


Fig. 7. Effect of the visual attention parameter p : input image (a), detected salient objects with high values of p (b) and small values of p (c)

Fig. 6 and Fig.7 show examples of salient object detection as well as effect of the visual attention parameter p on extracted salient regions, respectively.

4.3 Visual Intention CCVF

As it has previously been stated, composed of Conscious Cognitive Visual Functions (CCVF), the conscious visual level conducts intentional visual tasks. One of the core functions of this level is "visual intention" CCVF, proffering the robot some kind of "artificial visual intention ability" and allows the machine to construct its first knowledge about the surrounding environment. Fig. 8 gives the bloc-diagram of visional intention CCVF. As it could be seen from this figure, this CCVF is composed of four elementary functions: "Short-term Salient objects' Visual Memory" (SSVM) EF, "Unsupervised Learning Module" (ULM) EF, "Salient Objects' Classifier" (SOC) EF and "Object Detector" (OD) EF.

The main task of Short-term Salient objects' Visual Memory (SSVM) EF is to provide already known objects and storage currently recognized or detected salient objects. It could also be seen as the first knowledge construction of surrounding environment, because it contains the clusters of salient objects resulting from unsupervised learning. Its content (e.g. stored salient objects or groups of salient objects) could supply the main knowledge base (a long-term memory). That is why its output is also connected to inter-levels channel.

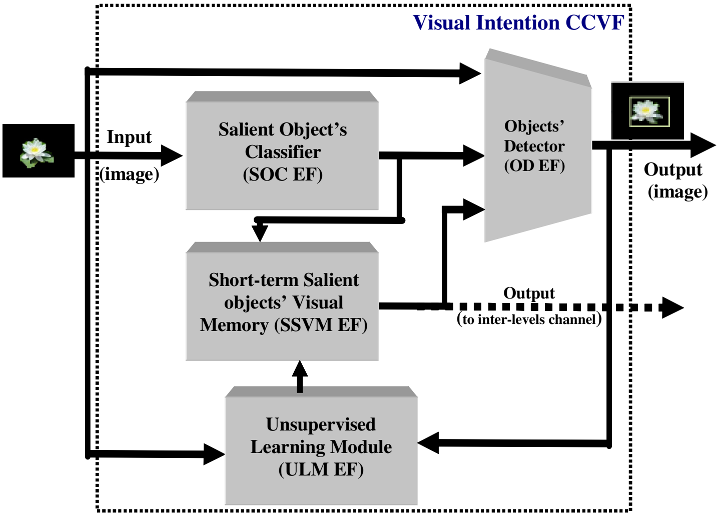

Visual Intention CCVF
Input
(image)
Salient Object's Classifier (SOC EF)
Objects' Detector (OD EF)
Output
(image)
Short-term Salient objects' Visual Memory (SSVM EF)
Output
(to inter-levels channel)
Unsupervised Learning Module (ULM EF)

Fig. 8. Bloc-diagram of visional intention CCVF

The role of Unsupervised Learning (performed by ULM EF) is to cluster the detected (new) salient objects. The learning process is carried out on-line. When an agent (e.g. robot) takes images while it encounters a new object, if the objects are recognized to be salient (e.g. extracted) they are grouped incrementally while new images are acquired. The action-flow of the learning process is given here-bellow. In the first time, the algorithm classifies each found fragment, and in a second time, the learning process is updated (on-line learning)

```
acquire image
extract fragments by salient object detector
for each fragment F
   if(F is classified into one group)
       populate the group by F
   if(F is classified into multiple groups)
        populate by F the closest group by Euclidian distance of
    features
    if(F is not classified to any group)
        create a new group and place F inside
select the most populated group G
use fragments from G as learning samples for object detection
algorithm
```

The Salient Object's Classifier in a combination of four weak classifiers $\{w_1, w_2, w_3, w_4\}$, each classifying a fragment as belonging or not belonging to a certain class. F denotes the currently processed fragment, G denotes an instance of the group in question. The first classifier w_1, defined by (12), separates fragments with too different areas. In experiments $t_{area} = 10$. The w_2 , defined by (13) separates fragments, whose aspect are too different to belong to the same object. In experiments, t_{aspect} has been set to 0.3. The classifier w_3 , defined by (14), separates fragments with clearly different chromaticity. It works over 2D normalized histograms of ϕ and θ component denoted by $G_{\phi\theta}$ and $F_{\phi\theta}$ respectively with L bins, calculating their intersection. We use L = 32 to avoid too sparse histogram and $t_{\phi\theta}$ equal to 0.35. Finally, w_4 (defined by (15)) separates fragments, whose texture is too different. We use the measure of texture uniformity calculated over the l channel of fragment. $p(z_i)$ where $i \in \{0, 1, 2, \cdots, L-1\}$ is a normalized histogram of l channel of the fragment and L is the number of histogram bins. In experiments, 32 histogram bins has been used to avoid too sparse histogram and value $t_{uniformity}$ of 0.02. A fragment belongs to a class if $\prod_{i=1}^{n} w_i$

$$w_1 = \begin{cases} 1 & if\ c_{W1} < t_{area} \\ 0 & otherwise \end{cases} ; c_{w1} = \frac{\max(G_{area}, F_{area})}{\min(G_{area}, F_{area})} \tag{12}$$

$$w_2 = \begin{cases} 1 & if\ c_{W2} < t_{aspect} \\ 0 & otherwise \end{cases} ; c_{w2} = \left\| \log\left(\frac{G_{width}}{G_{hight}}\right) - \log\left(\frac{F_{width}}{F_{hight}}\right) \right\| \tag{13}$$

$$w_3 = \begin{cases} 1 & if\ c_{W3} < t_{\phi\theta} \\ 0 & otherwise \end{cases} ; \quad with \quad c_{w3} = \frac{\sum_{j=1}^{L-1} \sum_{k=1}^{L-1} \min\left(G_{\phi\theta}(j,k) - F_{\phi\theta}(j,k)\right)}{L^2} \tag{14}$$

$$w_4 = \begin{cases} 1 & if\ c_{W4} < t_{uniformity} \\ 0 & otherwise \end{cases} ; c_{w4} = \left\| \sum_{j=0}^{L-1} p_G^2(z_j) - \sum_{k=0}^{L-1} p_F^2(z_k) \right\| \tag{15}$$

4.4 Implementation on Real Robot and Experimental Validation

The above-described concept has been implemented on NAO robot, which includes vision devices and a number of onboard pre-implemented motion skills. For experimental verification, the robot has been introduced in a real environment with different common objects (with different surface, shapes and properties). Several objects were exposed in robots field of view, presented in visual context different to the learning time. The number of images acquired for each object varied between 100 and 600 for learning image sequences and between 50 and 300 for testing sequences, with multiple objects occurring on the same scene. During learning, objects were correctly extracted from 82% of the images acquired by the robot. The subsequent grouping of fragments has achieved on the same dataset success rate of 96%, i.e. only 4% of image fragments (usually bearing close visual resemblance to other similar objects on the scene) were placed into a wrong groups.

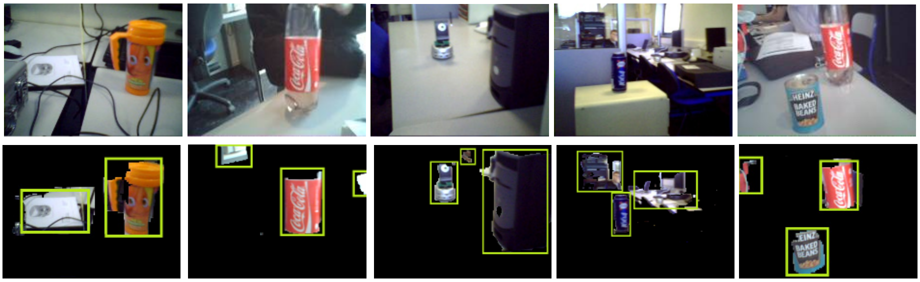


Fig. 9. NAO robot's camera issued images (upper images) and salient objects found and segmented by NAO (lower images)

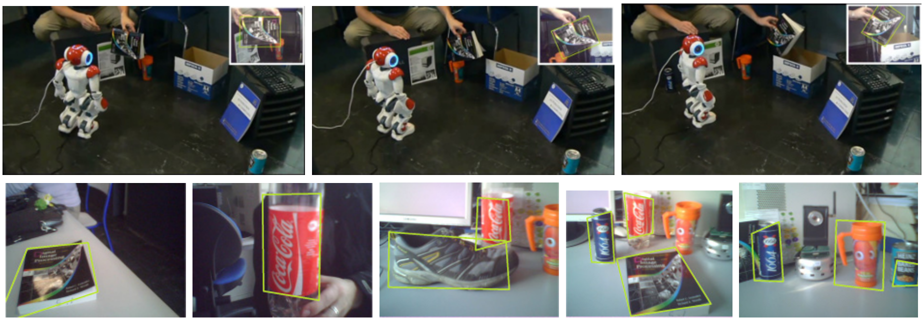


Fig. 10. Tracking a previously learned moving object (upper images: video http://www.youtube.com/watch?v=xxz3wm3L1pE). The upper right corner of each image shows robot camera's picture. The test's results relative to a set objects' detection by robot (lower images).

To demonstrate real-time abilities the system, the NAO robot was required to learn objects and then to find them in its environment, to track and to follow them. Some results of those experiments are shown on Fig. 9 and Fig. 10.

5 Conclusions

By supplanting the modeling of robots' complex behavior from the "control theory" backdrop to the "cognitive machine learning" backcloth, the proposed machine-learning based multi-level cognitive Motion-Perception concept attempts to offer a unified model of robot's autonomous evolution, slotting in two kinds of cognitive levels: "unconscious" and "conscious" cognitive levels, answerable of its reflexive and intentional visual and motor skills, respectively.

The first key-advantage of conceptualizing the problem within such incline is to detach the build-up of robot's perception and motion from the type of machine (robot). The second chief-benefit of the concept is that the issued structure is "Machine Learning" based foundation taking advantage from "learning" capacity and "generalization" propensity of such models. The validation has been performed considering a real biped robot. The issued results show effectiveness of the proposed cognitive multi-level architecture.

References

1. Westervelt, E.R., Buche, G., Grizzle, J.W.: Experimental validation of a framework for the design of controllers that induce stable walking in planar bipeds. International J. of Robotics Research 23(6), 559–582 (2004)
2. Park, J.H., Kwon, O.: Reflex Control of Biped Robot Locomotion on a Slippery Surface. In: Proc. IEEE Conf. on Robotics and Automation, pp. 4134–4139 (2001)
3. Kuffner, K., Nishiwaki, S., Kagami, M., Inaba, H.: Inoue. Footstep Planning Among Obstacles for Biped Robots. In: Proceedings of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 500–505 (2001)
4. Chestnutt, J., Kuffner, J.J.: A Tiered Planning Strategy for Biped Navigation. In: Int. Conf. on Humanoid Robots (Humanoids 2004), Proceedings, vol. 1, pp. 422–436 (2004)
5. Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N., Tanie, K.: Planning walking patterns for a biped robot. IEEE Transac. on Robotics and Automation 17(3), 280–289 (2001)
6. Sabe, K., Fukuchi, M., Gutmann, J., Ohashi, T., Kawamoto, K., Yoshigahara, T.: Obstacle Avoidance and Path Planning for Humanoid Robots using Stereo Vision. In: Int. Conf. on Robotics Automation (ICRA), pp. 592–597 (2004)
7. Holmes, R.: Acts of War: The Behavior of Men in Battle (First American Edition). The Free Press, New York (1985)
8. Tambe, M., Johnson, W., Jones, R., Koss, F., Laird, J., Rosenbloom, P., Schwamb, K.: Intelligent Agents for Interactive Simulation Environments. AI Magazine 16(1), 15–40 (1995)
9. Langley, P.: An abstract computational model of learning selective sensing skills. In: Proceedings of the 18th Conference of the Cognitive Science Society, pp. 385–390 (1996)
10. Bauckhage, C., Thurau, C., Sagerer, G.: Learning Human-Like Opponent Behavior for Interactive Computer Games. In: Michaelis, B., Krell, G. (eds.) DAGM 2003. LNCS, vol. 2781, pp. 148–155. Springer, Heidelberg (2003)
11. Potkonjak, V., Kostic, D., Tzafestas, S., Popovic, M., Lazarevic, M., Djordjevic, G.: Human-like behavior of robot arms: general considerations and the handwriting task. Robotics and Computer-Integrated Manufacturing 17(4), 317–327 (2001)
12. Edlund, J., Gustafson, J., Heldner, M., Hjalmarsson, A.: Towards human-like spoken dialogue systems. J. Speech Communication 50(8-9), 630–645 (2008)
13. Lubin, A., Poirel, N., Rossi, S., Pineau, A., Houdé, O.: Math in actions: Actor mode reveals the true arithmetic abilities of French-speaking two-year-olds in a magic task. J. of Experimental Child Psychology (103), 376–385 (2009)
14. Campbell, F.A., Pungello, E.P., Miller-Johnson, S., Burchinal, M., Ramey, C.T.: The development of cognitive and academic abilities: growth curves from an early childhood educational experiment. Dev. Psychol. 37(2), 231–242 (2001)
15. Leroux, G., Joliot, M., Dubal, S., Mazoyer, B., Tzourio-Mazoyer, N., Houdé, O.: Cognitive inhibition of number/length interference in a Piaget-like task: Evidence from ERP and fMRI. Human Brain Mapping (27), 498–509 (2006)
16. Lubin, A., Poirel, N., Rossi, S., Lanoé, C., Pineau, A., Houdé, O.: Pedagogical effect of action on arithmetic performances in Wynn-like tasks solved by 2-year-olds. Experimental Psychology (2010)
17. Cassell, O.C., Hubble, M., Milling, M.A., Dickson, W.A.: Baby walkers; still a major cause of infant burns. Burns 23, 451–453 (1997)
18. Crouchman, M.: The effects of babywalkers on early locomotor development. Developmental Medicine and Child Neurology (8), 757–761 (1986)

19. Siegel, A., Burton, R.: Effects of babywalkers on early locomotor development in human infants. Dev. Behav. Pediatr. (20), 355–361 (1999)
20. Kauffmann, I., Ridenour, M.: Influence of an infant walker on onset and quality of walking pattern of locomotion: an electromyographic investigation. Percept Motor Skills (45), 1323–1329 (1987)
21. Madani, K., Sabourin, C.: Multi-level cognitive machine-learning based concept for human-like "artificial" walking: application to autonomous stroll of humanoid robots. Neurocomuting 74, 1213–1228 (2011)
22. Bülthoff, H., Wallraven, C., Giese, M.: Perceptual Robotic. In: Siciliano, B., Khatib, O. (eds.) Handbook of Robotics. Springer (2007)
23. http://www.universcience-vod.fr/media/577/la-marche-des-bebes.html
24. Zukow-Goldring, P., Arbib, M.A.: Affordances, effectivities, and assisted imitation: Caregivers and the directing of attention. Neurocomputing 70, 2181–2193 (2007)
25. Brand, R.J., Baldwin, D.A., Ashburn, L.A.: Evidence for 'motionese': modifications in mothers infant-directed action. Developmental Science 5, 72–83 (2002)
26. Achanta, R., Hemami, S., Estrada, E., Susstrunk, S.: Frequency-tuned Salient Region Detection. In: IEEE Inrernat. Conf. on Compurer Vision & Pattern Recognition, CVPR (2009)
27. Wolfe, J.M., Horowitz, T.S.: What attributes guide the deployment of visual attention and how do they do it. Nature Reviews Neuroscience 5, 495–501 (2004)
28. Hou, X., Zhang, L.: Saliency detection: A spectral residual approach. IEEE Conference on Computer Vision and Pattern Recognition 2(800), 1–8 (2007)
29. Moreno, R., Graña, M., Ramik, D.M., Madani, K.: Image segmentation by spherical coordinates. In: Proc. of 11th Internat. Conf. on Pattern Recognition and Information Processing (PRIP 2011), pp. 112–115 (2011)
30. Holland, J.H.: Adaptation in Natural anti Artificial Systems: An introductory Analysis with Applications to Biology. In: Control and Artificial Intelligence. MIT Press (1992)
31. Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.-Y.: Learning to detect a salient object. JEEE Trans. Pattern Anal. Mach. Intell. 33(2), 353–367 (2011)
32. Ramík, D.M., Sabourin, C., Madani, K.: Hybrid Salient Object Extraction Approach with Automatic Estimation of Visual Attention Scale. In: Proc. 7th Internat. Conf. on Signal Image Technology & Internet-Based Systems (IEEE – SITIS 2011), pp. 438–445 (2011)
33. Ramík, D.M., Sabourin, C., Madani, K.: A Cognitive Approach for Robots' Vision Using Unsupervised Learning and Visual Saliency. In: Cabestany, J., Rojas, I., Joya, G. (eds.) IWANN 2011, Part I. LNCS, vol. 6691, pp. 81–88. Springer, Heidelberg (2011)

Part I
Intelligent Control Systems and Optimization

Robust Output Feedback Interpolation Based Control for Constrained Linear Systems

Hoai-Nam Nguyen[1], Sorin Olaru[1,⋆], Per-Olof Gutman[2], and Morten Hovd[3]

[1] SUPELEC Systems Sciences (E3S) - Automatic Control Department, Gif sur Yvette, France
[2] Faculty of Civil and Environmental Engineering, Technion - Israel Institute of Technology, Haifa, Israel
[3] Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway
{hoainam.nguyen,sorin.olaru}@supelec.fr
peo@technion.ac.il, morten.hovd@itk.ntnu.no

Abstract. In this paper, we consider the robust output feedback problem for constrained linear systems. A novel interpolation based control scheme is introduced, which guarantees feasibility and robust asymptotically stable closed loop behavior despite the presence of constraints on the input and output variables and the presence of the additive and bounded disturbances. A solution to the problem of the state representation is provided through the use of stored input values and the measured past outputs.

1 Introduction

This paper considers the problem of output feedback control design for a class of linear discrete time systems in presence of output and control constraints and subject to bounded disturbance. The boundedness assumptions on the different manipulated signals will be modeled by means of polyhedral constraints which assure a global linear system description (linear difference equation and linear equalities/inequalities).

There are several papers in the literature dealing with the output feedback synthesis problem. Due to the presence of input and state constraints, the robust model predictive control (MPC) design seems to best fit our objectives. Indeed, based on a Luenberger observer, an approach that incorporates the error on the state estimation as an additive bounded disturbance has been proposed in [1]. The estimation error is then taken in to account in the classical design of the constrained controller. A different approach is taken in [2], where the authors include the observer dynamics in the computation of the domain of attraction of the closed loop system.

The main drawback of the observer-based approaches is that, when the constraints become active, the nonlinearity dominates the properties of the state feedback control system and one cannot expect the separation principle to hold. Moreover there is no guarantee that the constraints will be satisfied along the closed-loop trajectories.

The work of [3] proposed an approach to MPC based on a non-minimal state space model, in which the states are represented by measured past inputs and outputs. This

⋆ The second author wish to acknowledge support from the RNC programme Yggdrasil.

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 21–36.
springerlink.com

approach eliminates the need of an observer. However the resulting state space model is unobservable and the state dimension may be large.

The main aim of the present paper is twofold. In the first part, we revisit the problem of state construction through measurement and storage of appropriate previous measurements. We recall that, there exists a *minimal* state space model with the structural constraints of having a state variable vector available though measurement and storage of appropriate previous measurements. Even if this model might be *non-minimal* from the classical state space representation point of view, it is directly measurable and will provide an appropriate model for the control design with constraints handling guarantees.

In the second part, starting from this state space model, we consider the robust control problem of constrained discrete-time linear invariant systems with disturbance and bounded input. For this purpose, two types of controller will be used in this paper. The first one is the global vertex controller [4]. The second one is the local unconstrained robust optimal control. Based on an interpolation technique and by minimizing an appropriate objective function, feasibility and a robustly asymptotically stable closed loop behavior are achieved.

The following notations will be used throughout the paper. We call a C-set a convex and compact set and containing the origin as an interior point. A polyhedron, or a polyhedral set, is the intersection of a finite number of half spaces. A polytope is a closed and bounded polyhedral set. Given two sets $X_1 \subset R^n$ and $X_2 \subset R^n$, the Minkowski sum of the sets X_1 and X_2 is defined by $X_1 \oplus X_2 \triangleq \{x_1 + x_2 | \, x_1 \in X_1, x_2 \in X_2\}$. The set X_1 is a proper subset of the set X_2 if and only if X_1 lies strictly inside X_2. For the set X, let Fr(X) be the boundary of X, Int(X) be the interior of X.

The paper is organized as follows. Section 2 is concerned with the problem statement. Section 3 is dedicated to the state space realization. Section 4 deals with the problem of computing an invariant set, while Section 5 is concerned with an interpolation technique. The simulation results are evaluated in Section 6 before drawing the conclusions.

2 Problem Statement

Consider the regulation problem for the following discrete linear time-invariant system, described by the input-output relationship

$$\begin{aligned} & y(t+1) + D_1 y(t) + D_2 y(t-1) + \ldots + D_n y(t-n+1) \\ & = N_1 u(t) + N_2 u(t-1) + \ldots + N_m u(t-m+1) + w(t) \end{aligned} \tag{1}$$

where: $y(t) \in R^q$, $u(t) \in R^p$, $w(t) \in R^q$ and $D_i, i = 1, \ldots, n$ and $N_i, i = 1, \ldots, m$ are matrices of suitable dimension.

It is assumed that $m \leq n$. The output and control are subject to the following hard constraints

$$y(t) \in Y, \; u(t) \in U \tag{2}$$

where $Y = \{y : F_y y \leq g_y\}$ and $U = \{u : F_u u \leq g_u\}$ are polyhedral sets and contain the origin in their interior.

The signal $w(t)$ represents the disturbance input. In this paper, we assume that the disturbance $w(t)$ is unknown, additive and lie in the polytope W, i.e. $w(t) \in W$, where $W = \{w : F_w w \leq g_w\}$ is a C-set.

3 State Space Model

In this section, the measured plant input, output and their past measured values are used to represent the states of the plant.

To simplify the description, it is assumed that $m = n$. Note that this assumption is always true, by supposing $N_{m+1} = N_{m+2} = \ldots = N_n = 0$.

The state space model of the system is constructed along the lines of [5], but the details of the construction are included here for the presentation to be self contained.

$$x(t) = \left(x_1(t)^T \; x_2(t)^T \; \ldots \; x_n(t)^T \right)^T \tag{3}$$

where $(*)^T$ denotes the transpose of matrix $(*)$ and

$$\begin{cases} x_1(t) = y(t) \\ x_2(t) = -D_n x_1(t-1) + N_n u(t-1) \\ x_3(t) = -D_{n-1} x_1(t-1) + x_2(t-1) + N_{n-1} u(t-1) \\ x_4(t) = -D_{n-2} x_1(t-1) + x_3(t-1) + N_{n-2} u(t-1) \\ \vdots \\ x_n(t) = -D_2 x_1(t-1) + x_{n-1}(t-1) + N_2 u(t-1) \end{cases} \tag{4}$$

It is clear that

$$\begin{aligned} x_2(t) &= -D_n y(t-1) + N_n u(t-1) \\ x_3(t) &= -D_{n-1} y(t-1) - D_n y(t-2) + N_{n-1} u(t-1) + N_n u(t-2) \\ &\vdots \\ x_n(t) &= -D_2 y(t-1) - D_3 y(t-2) - \ldots - D_n y(t-n+1) + \\ &\quad + N_2 u(t-1) + N_3 u(t-2) + \ldots + N_n u(t-n+1) \end{aligned}$$

One has

$$\begin{aligned} y(t+1) = -D_1 y(t) - D_2 y(t-1) - \ldots - D_n y(t-n+1) \\ +N_1 u(t) + N_2 u(t-1) + \ldots + N_n u(t-n+1) + w(t) \end{aligned}$$

or

$$x_1(t+1) = -D_1 x_1(t) + x_n(t) + N_1 u(t) + w(t)$$

The state space model is then defined as follows

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) + Ew(t) \\ y(t) = Cx(t) \end{cases} \tag{5}$$

where

$$A = \begin{pmatrix} -D_1 & 0_q & 0_q & \dots & 0_q & I_q \\ -D_n & 0_q & 0_q & \dots & 0_q & 0_q \\ -D_{n-1} & I_q & 0_q & \dots & 0_q & 0_q \\ -D_{n-2} & 0_q & I_q & \dots & 0_q & 0_q \\ \dots & \dots & \dots & \dots & \dots & \dots \\ -D_2 & 0_q & 0_q & \dots & I_q & 0_q \end{pmatrix}, \quad B = \begin{pmatrix} N_1 \\ N_n \\ N_{n-1} \\ N_{n-2} \\ \dots \\ N_2 \end{pmatrix}, \quad E = \begin{pmatrix} I_q \\ 0_q \\ 0_q \\ 0_q \\ \dots \\ 0_q \end{pmatrix},$$

$$C = \begin{pmatrix} I_q \; 0_q \; 0_q \; 0_q \; \dots \; 0_q \end{pmatrix}.$$

Here I_q, 0_q denote the identity and zeros matrices of dimension $q \times q$, respectively.

It should be noted that, the state space realization (5) is minimal in the single input single output case, but might not be minimal for multiple inputs and/or multiple outputs, as showing in the following example. Consider the SIMO discrete time system

$$y(t+1) + \begin{pmatrix} -2 & 0 \\ 0 & -2 \end{pmatrix} y(t) + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} y(t-1) = \begin{pmatrix} 0.5 \\ 2 \end{pmatrix} u(t) + \begin{pmatrix} 0.5 \\ 1 \end{pmatrix} u(t-1) + w(t) \tag{6}$$

Using the above construction, the state space model is given as follows

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

where

$$A = \begin{pmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \\ -1.5 \end{pmatrix}, \quad E = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

It is obvious that this realization is not minimal. One of the minimal realizations of the system is given by

$$A = \begin{pmatrix} 0 & -1 \\ 1 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}, \quad E = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad C = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

□

Denote

$$z(t) = (y(t)^T \; y(t-1)^T \; \dots \; y(t-n+1)^T \; u(t-1)^T \; u(t-2)^T \; \dots \; u(t-n+1)^T)^T \tag{7}$$

The state vector $x(t)$ (3) is related to the vector $z(t)$ as follows

$$x(t) = Tz(t) \tag{8}$$

where

$$T = (T_1 \quad T_2)$$

$$T_1 = \begin{pmatrix} I_q & 0_q & 0_q & \dots & 0_q \\ 0_q & -D_n & 0_q & \dots & 0_q \\ 0_q & -D_{n-1} & -D_n & \dots & 0_q \\ \dots & \dots & \dots & \dots & \dots \\ 0_q & -D_2 & -D_3 & \dots & -D_n \end{pmatrix}, \quad T_2 = \begin{pmatrix} 0_{q\times p} & 0_{q\times p} & 0_{q\times p} & \dots & 0_{q\times p} \\ N_n & 0_{q\times p} & 0_{q\times p} & \dots & 0_{q\times p} \\ N_{n-1} & N_n & 0_{q\times p} & \dots & 0_{q\times p} \\ \dots & \dots & \dots & \dots & \dots \\ N_2 & N_3 & N_4 & \dots & N_n \end{pmatrix}$$

From the formula (8), it is apparent that at any time instant t, the state variable vector is available though measurement and storage of appropriate previous measurements.

4 Invariant Set Construction

Using (4), it is clear that $x_i(t) \in X_i$ where X_i is given by

$$\begin{aligned}
&X_1 = Y \\
&X_2 = D_n(-X_1) \oplus N_n(U) \\
&X_i = D_{n+2-i}(-X_1) \oplus X_{i-1} \oplus N_{n+2-i}U, \forall i = 3, \ldots, n
\end{aligned}$$

In summary, the constraints on the state are $x \in X$, where the set X is expressed as a collection of linear inequalities describing a polyhedron $X = \{x : \ F_x x \leq g_x\}$.

4.1 Maximal Robustly Admissible Set for $\boldsymbol{u = Kx}$

Using established results in control theory (LQR, LQG, LMI based, ...), one can find a feedback gain K, that quadratically stabilizes the system (5) with some desired properties. The details of such a synthesis procedure are not reproduced here, but we assume that the feasibility of such an optimization based robust control design is guaranteed, leading to a closed loop transition matrix $A_c = A + BK$.

Definition 1 (Robustly Positively Invariant Set). The set $\Omega \subseteq X$ is a robustly positively invariant (RPI) set with respect to $x(t+1) = A_c x(t) + Ew(t)$ if and only if

$$\forall x \in \Omega \Rightarrow A_c x + Ew \in \Omega \tag{9}$$

for any $w \in W$.

Definition 2 (Minimal RPI). The set $\Omega_\infty \subseteq X$ is a minimal RPI (mRPI) set with respect to $x(t+1) = A_c x(t) + Ew(t)$ if and only if Ω_∞ is an RPI and contained in any RPI set.

It is possible to show that if the mRPI set Ω_∞ exists, then it is unique, bounded and contains the origin in its interior [6], [7]. Moreover, all trajectories of the system $x(t+1) = A_c x(t) + Ew(t)$ starting from the origin, are bounded by Ω_∞. It follows from linearity and asymptotic stability of A_c, that Ω_∞ is the limit set of all trajectories of the system $x(t+1) = A_c x(t) + Ew(t)$.

It is clear that, it is impossible to devise a controller $u(t) = Kx(t)$ such that $x(t) \to 0$ as $t \to \infty$. The best that can be hoped for is that the controller steers any initial state to the mRPI set Ω_∞, and maintains the state in this set once it is reached. In other words, the set Ω_∞ can be considered as the limit set for the trajectories of the system (5).

In the sequel, it is assumed that the set Ω_∞ is a proper subset of the set Y.

Definition 3 (Maximal RPI). The set $O_\infty \in X$ is a maximal RPI (MRPI) set with respect to $x(t+1) = A_x x(t) + Ew(t)$ and constraints on the state $x \in X$ if and only if $O_\infty \subset X$ is an RPI and contains every RPI set.

If the MRPI set is non-empty, then it is unique. Furthermore if X, U and W are C-sets, then the MRPI set O_∞ is also a C-set.

The mRPI set Ω_∞ and the MRPI set O_∞ are connected be the following theorem:

Theorem 1. The following statements are equivalent:

1. The MRPI set O_∞ is non-empty.
2. $\Omega_\infty \subset X$

Proof. Interested readers are referred to [6] for the details of the proof. □

Define the polytope P_{xu} as follows

$$P_{xu} = \{x : \ F_{xu}x \le g_{xu}\} \tag{10}$$

where

$$F_{xu} = \begin{pmatrix} F_x \\ F_u K \end{pmatrix}, \, g_{xu} = \begin{pmatrix} g_x \\ g_u \end{pmatrix}$$

Under the assumption that the Ω_∞ is a proper subset of X, a constructive procedure is used to compute the MRPI set, as follows [8].

Procedure 1. Maximal robustly positively invariant set computation.

1. Set $t = 0$, $F_t = F_{xu}$, $g_t = g_{xu}$ and $P_t = P_{xu}$.
2. Set $P_t^1 = P_t$
3. Solve the following set of linear programs
$$d = \max F_t E w, \text{ s.t. } w \in W$$
4. Compute a polytope
$$P_t^2 = \{x : F_t A_c x \le g_t - d\}$$
5. Set P_t as an intersection
$$P_t = P_t^1 \cap P_t^2$$
6. If $P_t = P_t^1$ then stop and set $O_\infty = P_t$. Else continue.
7. Set $t = t + 1$, go to step 2.

Non-emptiness property of the MRPI set O_∞ assures that the above procedure terminates in finite time and lead to the MRPI in form of a polytope

$$O_\infty = \{x : F_o x \le g_o\} \tag{11}$$

4.2 Robustly Positively Controlled Invariant Set for $u \in U$

Recall the following definitions related to the control of systems under constraints [8].

Definition 4 (Robust Controlled Positively Invariant Set). Given the system (5), the set $\Psi \subseteq X$ is robust controlled positively invariant if and only if for any $x(t) \in \Psi$, there exists a control action $u(t) \in U$ such that for any $w(t) \in W$, one has $x(t+1) = Ax(t) + Bu(t) + Ew(t) \in \Psi$.

Definition 5 (Pre-image Set). Given the polytopic system (1), the one-step pre-image set of the set $P_0 = \{x : F_0 x \le g_0\}$ is given by all states that can robustly be steered in one step in P_0 when a suitable control is applied. The pre-image set, called $P_1 = Pre(P_0)$ can be shown to be:

$$P_1 = \{x \in R^n : \exists u \in U : F_0(Ax + Bu) \leq g_0 - \max F_0 Ew\} \qquad (12)$$

where $w \in W$.

Remark 1. It is clear that if the set Ψ is contained in its pre-image set the Ψ is invariant.

Recall that the set O_∞ is the MRPI. Define P_N as the set of states, that can be steered to the O_∞ in no more that N steps along an admissible trajectory, i.e. a trajectory satisfying control, state and disturbance constraints. This set can be generated recursively by the following procedure:

Procedure 2. Invariant set computation

1. Set $k = 0$ and $P_0 = O_\infty$.
2. Define
$$P_{k+1} = \text{Pre}(P_k) \bigcap X$$
3. If $P_{k+1} = P_k$, then stop and set $P_N = P_k$. Else continue.
4. If $k = N$, then stop else continue.
5. Set $k = k + 1$ and go to the step 2.

A a consequence of the fact that O_∞ is an invariant set, it follows that for each k, $P_{k-1} \subset P_k$ and therefore P_k is an invariant set and a sequence of nested polytopes. Note that the complexity of the set P_N does not have an analytic dependence on N and may increase without bound, thus placing a practical limitation on the choice of N.

For further use, the controlled invariant set resulting from the Procedure 2 is denoted

$$P_N = \{x : F_N x \leq g_N\} \qquad (13)$$

5 Interpolation Based Controller with Linear Programming

The purpose of this section is to show how an interpolation technique can be used within a linear programming based control law design.

5.1 Vertex Control Law

Given a positively invariant polytope $P_N \in R^n$, this polytope can be decomposed in a sequence of simplices P_N^k each formed by n vertices $x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)}$ and the origin. These simplices have following properties:

- Int$(P_N^k) \neq \emptyset$,
- Int$(P_N^k \cap P_N^l) = \emptyset$ if $k \neq l$,
- $\bigcup_k P_N^k = P_N$,

Denote by $X^{(k)} = (x_1^{(k)}\ x_2^{(k)}\ \ldots\ x_n^{(k)})$ the square matrix defined by the vertices generating P_N^k. Since P_N^k has nonempty interior, $X^{(k)}$ is invertible. Let $U^{(k)} = (u_1^{(k)}\ u_2^{(k)}\ \ldots\ u_n^{(k)})$ be the matrix defined by the *admissible control* values at these vertices. For $x \in P_N^k$ consider the following linear gain K^k:

$$K^k = U^{(k)}(X^{(k)})^{-1} \tag{14}$$

Here by admissible control values we understand any control actions that steer the state x into the P_N in *finite time*. The reader is referred to [4] for more details.

Remark 2: Maximizing the control action at the vertices of P_N can be achieved by using the following program.

$$J = \max \|u\|_p \text{ s.t. } \begin{cases} F_N(Ax + Bu) \leq g_N - \max F_N Ew \\ F_u u \leq g_u. \end{cases} \tag{15}$$

where $\|u\|_p$ is a $p-$ norm of vector u and $w \in W$.

Due to the properties of the positive invariant set, the above program is always feasible.

Theorem 2. The piecewise linear control $u = K^k x$ is feasible and asymptotically stable for all $x \in P_N$.

The proof of this theorem is not reported here, the interested reader being referred to [4] and [9] for the necessary elements.

5.2 Interpolation Via Linear Programming

Any state $x(t)$ in P_N can be decomposed as follows

$$x(t) = cx_v(t) + (1-c)x_o(t) \tag{16}$$

where $x_v(t) \in P_N$, $x_o(t) \in O_\infty$ and $0 \leq c \leq 1$.

Consider the following control law

$$u(t) = cu_v(t) + (1-c)u_o(t) \tag{17}$$

where $u_v(t)$ is obtained by applying the vertex control law and $u_o(t) = Kx_o(t)$ is the control law, that is feasible in O_∞.

Theorem 3. The control law (17) is feasible for all $x \in P_N$.

Proof. Corresponding to the decomposition, the control law is given by (17).

One has to prove that $F_u u(t) \leq g_u$ and $x(t+1) = Ax(t) + Bu(t) + Ew(t) \in P_N$ for all $x(t) \in P_N$ and for any $w(t) \in W$.

One has

$$\begin{aligned} F_u u(t) &= F_u(cu_v(t) + (1-c)u_o(t)) = cF_u u_v(t) + (1-c)F_u u_o(t) \\ &\leq cg_u + (1-c)g_u = g_u \end{aligned}$$

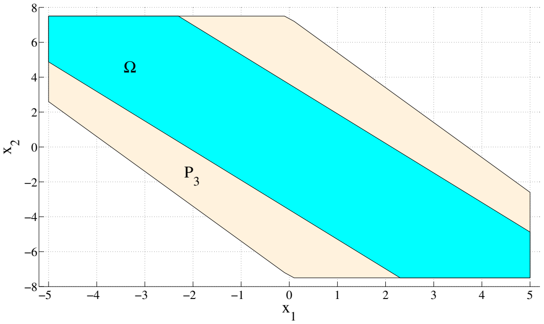


Fig. 1. Feasible regions for example 1. The blue set is the MRPI O_∞, when applying the control law $u = Kx$. The red set is the positive invariant set P_N.

and

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) + Ew(t) \\ &= A(cx_v(t) + (1-c)x_o(t)) + B(cu_v(t) + (1-c)u_o(t)) + Ew(t) \\ &= c(Ax_v(t) + Bu_v(t) + Ew(t)) + (1-c)(Ax_o(t) + Bu_o(t) + Ew(t)) \end{aligned}$$

We have $Ax_v(t)+Bu_v(t)+Ew(t) \in P_N$ and $Ax_o(t)+Bu_o(t)+Ew(t) \in O_\infty \subset P_N$, it follows that $x(t+1) \in P_N$. □

In order to give a maximal control action, one would like to minimize c, so the following program is given:

$$c^*(x) = \min_{c,x_v,x_o} c, \text{ s.t. } \begin{cases} F_N x_v \leq g_N, \\ F_o x_o \leq g_o, \\ cx_v + (1-c)x_o = x, \\ 0 \leq c \leq 1 \end{cases} \tag{18}$$

Denote $r_v = cx_v$, $r_o = (1-c)x_o$. It is clear that $r_v \in cP_N$ and $r_o \in (1-c)O_\infty$ or equivalently $F_N r_v \leq cg_N$ and $F_w r_o \leq (1-c)g_w$. The above non-linear program is translated into a linear program as follows.

Interpolation based on Linear Programming

$$c^*(x) = \min_{c,r_v} c, \text{ s.t. } \begin{cases} F_N r_v \leq cg_N \\ F_o(x - r_v) \leq (1-c)g_o \\ 0 \leq c \leq 1 \end{cases} \tag{19}$$

Remark 3. If one would like to maximize c, it is obvious that $c = 1$ for all $x \in P_N$. In this case the controller turns out to be the vertex controller.

Theorem 4. The control law using interpolation based on linear programming (16), (17), (19) guarantees robustly asymptotic stability for all initial state $x(0) \in P_N$.

Proof. The complete proof of this theorem is given in [10]. □

6 Examples

To show the effectiveness of the proposed approach, two examples will be presented in this section. For both of these examples, to solve linear programs and to implement polyhedral operations, the Multi-parametric toolbox [11] has beed used.

6.1 Example 1

Consider the following discrete-time system

$$y(t+1) - 2y(t) + y(t-1) = 0.5u(t) + 0.5u(t-1) + w(t) \tag{20}$$

subject to the following constraints

$$-5 \leq y(t) \leq 5, \quad -5 \leq u(t) \leq 5, \quad -0.1 \leq w(t) \leq 0.1$$

The state space model is given by

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

where

$$A = \begin{pmatrix} 2 & 1 \\ -1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}, \quad E = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix}$$

The state $x(t)$ is available though the measured plant input, output and their past measured values as $x(t) = Tz(t)$, where

$$z(t) = \begin{pmatrix} y(t) & y(t-1) & u(t-1) \end{pmatrix}^T,$$
$$T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0.5 \end{pmatrix}$$

The constraints on the state are: $-5 \leq x_1 \leq 5, \quad -7.5 \leq x_2 \leq 7.5$.

Using the linear quadratic regulator with weighting matrices $Q = C'C$ and $R = 0.1$ the feedback gain is obtained $K = \begin{pmatrix} -2.3548 & -1.3895 \end{pmatrix}$

Using procedures 1 and 2 one obtains the set O_∞ and P_N as shown in Figure 1. Note that $P_3 = P_4$, in this case P_3 is a maximal invariant set for system (20).

The set of vertices of P_N is given by the matrix $V(P_N)$ below, together with the control matrix U_v

$$V(P_N) = \begin{pmatrix} -5 & -0.1 & 5 & 0.1 & -0.1 & -5 & 0.1 & 5 \\ 7.5 & 7.5 & -2.6 & 7.2 & -7.2 & 2.6 & -7.5 & -7.5 \end{pmatrix}$$

and

$$U_v = \begin{pmatrix} -5 & -5 & -5 & -4.9 & 5 & 5 & 5 & 4.9 \end{pmatrix}$$

Figure 2 shows the state space partition and 6 different trajectories of the closed loop system.

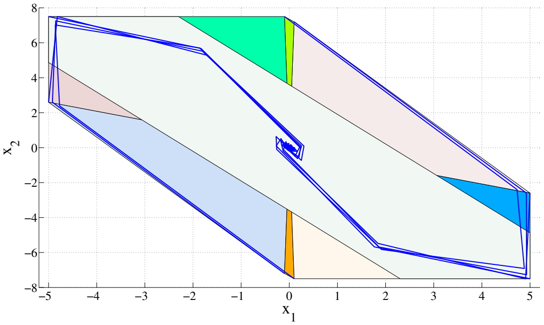


Fig. 2. State space partition and trajectories of the closed loop system for example 1

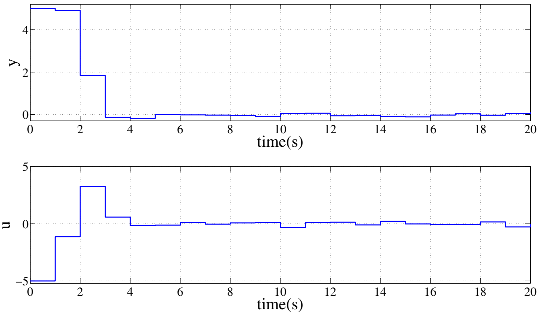


Fig. 3. Output and input trajectory for example 1

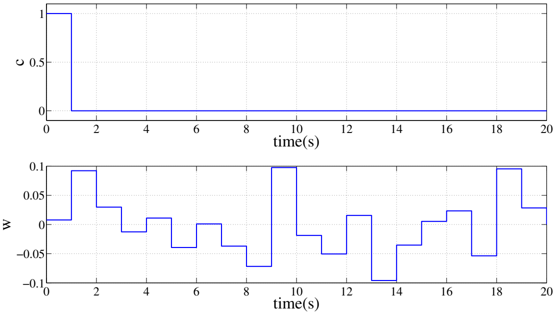


Fig. 4. The interpolating coefficient and the disturbance input for example 1

Corresponding to the initial condition $x_0 = (5.0000 \quad -2.6000)^T$, Figure 3 shows the output and input trajectory.

Figure 4 shows the disturbance input and the interpolating coefficient $c^*(t)$ as a function of t. As expected this function is positive and non-increasing.

In a comparison with the approach, that based on using the Kalman filter, Figure 5 shows the output trajectories using our approach and the Kalman filter based approach.

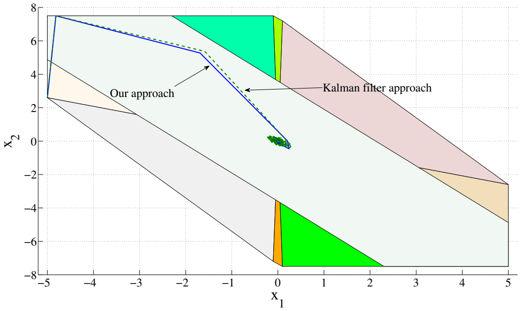


Fig. 5. The state trajectory of our approach and the Kalman filter based approach for example 1. The mRPI set of the Kalman filter based approach is bigger than the mRPI set of our approach.

It is obvious that, the mRPI set of the Kalman filter based approach is bigger than the mRPI set of our approach.

The Matlab routine with the command *'kalman'* is used for designing the Kalman filter. The process noise is a white noise with an uniform distribution and there is no measurement noise. w is a random number with an uniform distribution, $w_l \le w \le w_u$. The variance of w is given as follows:

$$C_w = \frac{(w_u - w_l + 1)^2 - 1}{12} = 0.0367$$

The estimator gain of the Kalman filter is obtained: $L = (2 \quad -1)^T$

The initial condition is $x_0 = (-4 \quad 6)^T$. The Kalman filter is used to estimate the state of the system and then this estimation is used to close the loop with the interpolated control law. In contrast to our approach, where the state is exact, in the Kalman filter approach, the state is not exact and moreover, there is no guarantee that the constraints are satisfied.

Figure 6 shows the output trajectories of our approach and the Kalman filter approach.

In Figure 7 it is showed that, the constraints might be violated where the Kalman filter is used to estimate the state of the system. Alternatively, if the estimator error is included, the feasible set is smaller.

6.2 Example 2

Consider the following discrete-time system

$$\begin{aligned} &y(t+1) + \begin{pmatrix} -1.8787 & 0 \\ 0 & -1.8964 \end{pmatrix} y(t) + \begin{pmatrix} 0.8787 & 0 \\ 0 & 0.8964 \end{pmatrix} y(t-1) = \\ &= \begin{pmatrix} -0.3800 & -0.5679 \\ -0.2176 & 0.4700 \end{pmatrix} u(t) + \begin{pmatrix} 0.3339 & 0.5679 \\ 0.2176 & -0.4213 \end{pmatrix} u(t-1) + w(t) \end{aligned} \tag{21}$$

The constraints are

$$\begin{aligned} -2 \le y_1 \le 2, &\quad -2 \le y_2 \le 2 \\ -10 \le u_1 \le 10, &\quad -10 \le u_2 \le 10 \\ -0.1 \le w_1 \le 0.1, &\quad -0.1 \le w_2 \le 0.1 \end{aligned}$$

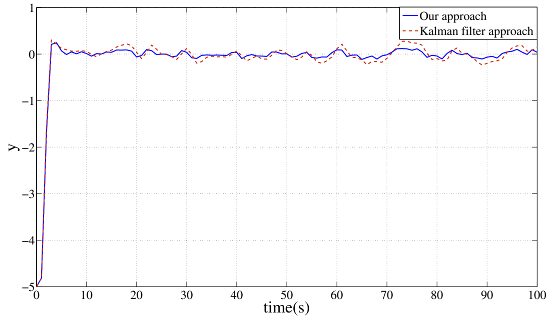


Fig. 6. The output trajectories of our approach and the Kalman filter approach for example 1

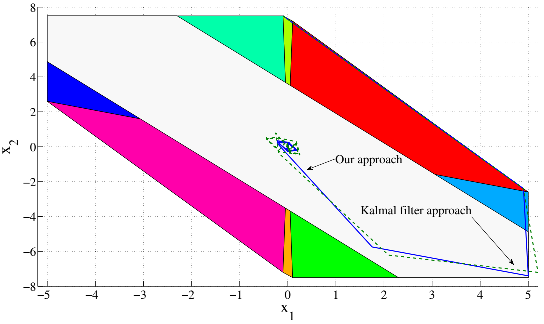


Fig. 7. Constraints violation for example 1

The state space model is

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

where

$$A = \begin{pmatrix} 1.8787 & 0 & 1.000 & 0 \\ 0 & 1.8964 & 0 & 1 \\ -0.8787 & 0 & 0 & 0 \\ 0 & -0.8964 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} -0.3800 & -0.5679 \\ -0.2176 & 0.4700 \\ 0.3339 & 0.5679 \\ 0.2176 & -0.4213 \end{pmatrix}, \quad E = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

It is worth noticing that, the above state space realization is minimal. The state $x(t)$ is available though the measured plant input, output and their past measured values as $x(t) = Tz(t)$, where

$$z(t) = \left(y(t)^T \; y(t-1)^T \; u(t-1)^T \right)^T,$$

$$T = \begin{pmatrix} 1.0000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.8787 & 0 & 0.3339 & 0.5679 \\ 0 & 0 & 0 & -0.8964 & 0.2176 & -0.4213 \end{pmatrix}$$

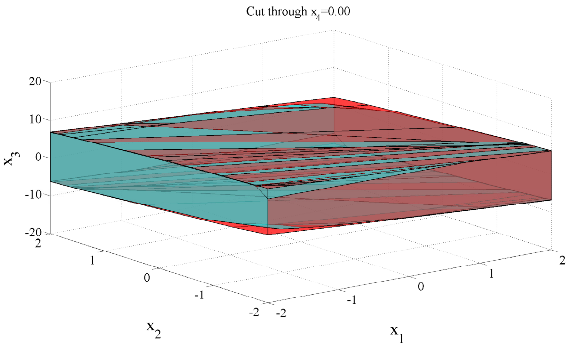


Fig. 8. Feasible regions for example 2, cut through $x_4 = 0$. The blue set is the MRPI set O_∞, when applying the control law $u = Kx$. The red set is the positive controlled invariant set P_3.

The constraints on the state are

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0.5460 & -0.8378 \\ 0 & 0 & -0.5958 & -0.8031 \\ 0 & 0 & -1.0000 & 0 \\ 0 & 0 & -0.5460 & 0.8378 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 0 & 0.0000 & -1.0000 \\ 0 & 0 & 0.5958 & 0.8031 \\ 0 & 0 & 1.0000 & 0.0000 \end{pmatrix} x \leq \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 9.0918 \\ 6.2239 \\ 10.7754 \\ 9.0918 \\ 8.1818 \\ 8.1818 \\ 6.2239 \\ 10.7754 \end{pmatrix}$$

Using the linear quadratic regulator with weighting matrices $Q = C'C$ and $R = I$, the feedback gain is obtained

$$K = \begin{pmatrix} 1.9459 & 1.7552 & 1.4968 & 1.3775 \\ 0.8935 & -1.7212 & 0.5524 & -1.2704 \end{pmatrix}$$

Using procedures 1 and 2, one obtains the set O_∞ and P_3 as illustrated in Figure 8. The number of vertices of the set P_3 is 1030 and these are not reported here. The control values at the vertices of the set P_3 are found by applying the program (15). Corresponding to the initial condition $x_0 = \begin{pmatrix} -1.67 & 0.21 & 10.78 & -3.83 \end{pmatrix}^T$, Figure 9 presents the output and input trajectories.

Figure 10 shows the disturbance inputs $w_1(t), w_2(t)$ and the interpolating coefficient $c^*(t)$ as a function of t. As expected, this function is positive and non-increasing.

In a comparison with the Kalman filter based approach, Figure 11 shows the output trajectories using our approach and the Kalman filter based approach.

The initial condition is $x_0 = \begin{pmatrix} -1.3378 & 0.1670 & 8.6203 & -3.0637 \end{pmatrix}^T$.

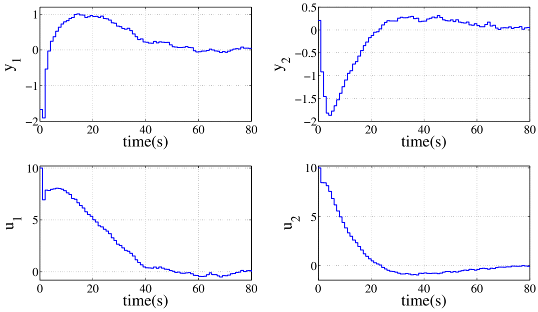


Fig. 9. Output and input trajectory for example 2

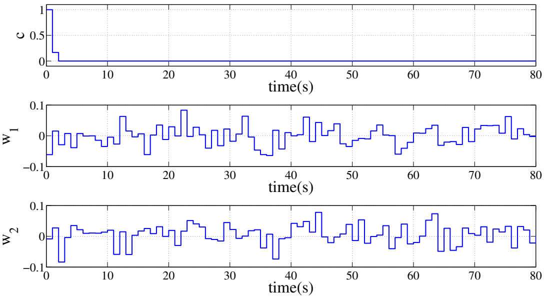


Fig. 10. The interpolating coefficient and the disturbance input for example 2

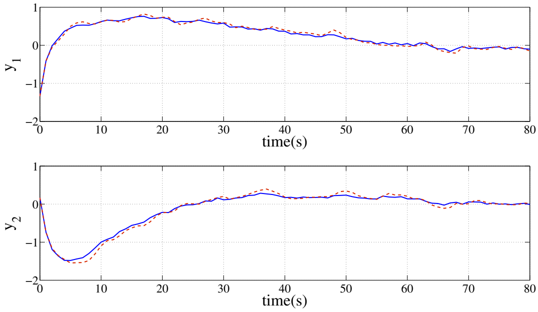


Fig. 11. The output trajectories of our approach and the Kalman filter approach for example 2. The blue line is our approach, and the red dotted line is the Kalman filter-based approach.

The Matlab routine with the command *'kalman'* is used for designing the Kalman filter. The process noise is a white noise with an uniform distribution and there is no measurement noise. w is a random vector with an uniform distribution, $w_l \leq w \leq w_u$. The covariance matrix of w is given as follows:

$$C_w = \frac{(w_u - w_l + 1)^2 - 1}{12} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0.0367 & 0 \\ 0 & 0.0367 \end{pmatrix}$$

The estimator gain of the Kalman filter is obtained:

$$L = \begin{pmatrix} 1.8787 & 0 \\ 0 & 1.8964 \\ -0.8787 & 0 \\ 0 & -0.8964 \end{pmatrix}$$

7 Conclusions

In this paper, a state space realization is described for discrete-time linear time invariant systems, with the particularity that the state variable vector is available through measurement and storage of appropriate previous measurements.

A robust control problem is solved based on the interpolation technique and using linear programming. Practically, the interpolation is done between a global vertex controller and a local unconstrained robust optimal control law.

Several simulation examples are presented including a comparison with an earlier solution from the literature and a multi-input multi-output system.

References

1. Mayne, D., Rakovic, S., Findeisen, R., Allgower, F.: Robust output feedback model predictive control of constrained linear systems. Automatica 42, 1217–1222 (2006)
2. Goulart, P., Kerrigan, E.: A method for robust receding horizon output feedback control of constrained systems. In: 2006 45th IEEE Conference on Decision and Control, pp. 5471–5476. IEEE (2007)
3. Wang, L., Young, P.: An improved structure for model predictive control using non-minimal state space realisation. Journal of Process Control 16, 355–371 (2006)
4. Gutman, P., Cwikel, M.: Admissible sets and feedback control for discrete-time linear dynamical systems with bounded controls and states. IEEE Transactions on Automatic Control 31, 373–376 (1986)
5. Taylor, C., Chotai, A., Young, P.: State space control system design based on non-minimal state-variable feedback: further generalization and unification results. International Journal of Control 73, 1329–1345 (2000)
6. Kolmanovsky, I., Gilbert, E.: Theory and computation of disturbance invariant sets for discrete-time linear systems. Mathematical Problems in Engineering 4, 317–363 (1998)
7. Rakovic, S., Kerrigan, E., Kouramas, K., Mayne, D.: Invariant approximations of the minimal robust positively invariant set. IEEE Transactions on Automatic Control 50, 406–410 (2005)
8. Blanchini, F., Miani, S.: Set-theoretic methods in control. Springer (2008)
9. Blanchini, F.: Minimum-time control for uncertain discrete-time linear systems. In: Proceedings of the 31st IEEE Conference on Decision and Control, pp. 2629–2634 (1992)
10. Nguyen, H.N., Gutman, P.O., Olaru, S., Hovd, M., Colledani, F.: Improved vertex control for uncertain linear discrete-time systems with control and state constraints. In: American Control Conference. IEEE (2011)
11. Kvasnica, M., Grieder, P., Baotić, M., Morari, M.: Multi-Parametric Toolbox (MPT). In: Alur, R., Pappas, G.J. (eds.) HSCC 2004. LNCS, vol. 2993, pp. 448–462. Springer, Heidelberg (2004)

SVM Approximation of Value Function Contours in Target Hitting Problems

Laetitia Chapel[1] and Guillaume Deffuant[2]

[1] Lab-STICC, Université Européenne de Bretagne, Université de Bretagne Sud
56017 Vannes Cedex, France
laetitia.chapel@univ-ubs.fr

[2] Laboratoire d'Ingénierie pour les Systèmes Complexes, Cemagref
63172 Aubière Cedex, France
guillaume.deffuant@clermont.cemagref.fr

Abstract. In a problem of target hitting, the capture basin at cost c is the set of states that can reach the target with a cost lower or equal than c, without breaking the viability constraints. The boundary of a c-capture basin is the c-contour of the problem value function. In this paper, we propose a new algorithm that solves target hitting problems, by iteratively approximating capture basins at successive costs. We show that, by a simple change of variables, minimising a cost may be reduced to the problem of time minimisation, and hence a recursive backward procedure can be set. Two variants of the algorithm are derived, one providing an approximation from inside (the approximation is included in the actual capture basin) and one providing a outer approximation, which allows one to assess the approximation error. We use a machine learning algorithm (as a particular case, we consider Support Vector Machines) trained on points of a grid with boolean labels, and we state the conditions on the machine learning procedure that guarantee the convergence of the approximations towards the actual capture basin when the resolution of the grid decreases to 0. Moreover, we define a control procedure which uses the set of capture basin approximations to drive a point into the target. When using the inner approximation, the procedure guarantees to hit the target, and when the resolution of the grid tends to 0, the controller tends to the optimal one (minimizing the cost to hit the target). We illustrate the method on two simple examples, Zermelo and car on the hill problems.

Keywords: Viability theory, Capture basin, Optimal control, Support vector machines.

1 Introduction

We consider a dynamical system, described by the evolution of its state variable $x \in \mathbb{R}^n$:

$$x(t)' \in F(x(t)) = \{\varphi(x(t), u(t)) \mid u(t) \in \mathcal{U}\}\,, \tag{1}$$

where $x(t)$ is the state at time t, F is a set-valued map and $\mathcal{U}$ the set of admissible control. We assume that there is a strictly positive cost $\ell(x, u)$ per unit of time for taking control u in state x.

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 37–48.
springerlink.com

We focus on the problem of defining the control function that drives the dynamical system inside a given target compact set $\mathcal{C} \subset \mathcal{K}$ without going out from a given set $\mathcal{K}$ (called the viability constraint set), and that minimises the cost functional

$$\inf_{u \in \mathcal{U}} \int_0^{+\infty} \ell(x(\tau), u(\tau)) \cdot d\tau. \tag{2}$$

This problem, often called the reachability problem, can be addressed by optimal control methods, solving Hamilton-Jacobi-Bellman (HJB) or Isaacs (HJI) equations. Several numerical techniques are available; for example, [10] propose an algorithm that computes an approximation for the backward reachable set of a system using a time dependent HJI partial differential equation, [9] builds the value function of the problem which can be then used to choose the best action at each step when the cost function represents the time.

Reachability problem can also be addressed in the viability theory framework [1]. To apply viability theory to target hitting problems when the cost to minimise is the time elapsed to reach the target, one must add an auxiliary dimension to the system, representing the time. The approach computes an approximation of the envelopes of all t-capture basins, the sets of points for which there exists a control function that allows the system to reach the target in a time less than t. [7] shows that the boundary of this set is the value function in the dynamical programming perspective. Hence, solving this extended viability problem also provides the minimal time for a state x to reach the target $\mathcal{C}$ while always staying in $\mathcal{K}$ (minimal time function $\vartheta_{\mathcal{C}}^{\mathcal{K}}(x)$ [4]). This function can then be used to define controllers that drive the system into the target. The same approach can be used for cost minimisation; in that case, the extra-dimension represents the cost-to-go to the target [3].

Several numerical algorithms [12,4] provide an over-approximation of capture basins. [2] implement an algorithm proposed by [13] that computes a discrete under-approximation of continuous minimal time functions (and thus an over-approximation of capture basins), without adding an additional dimension. [8] present an algorithm, based on interval analysis, that provides inner and outer approximations of the capture basin. In general, capture basin and minimal time function approximation algorithms face the curse of dimensionality, which limits their use to problems of low dimension (in the state and control space).

This paper proposes a new method to solve target hitting problems, inspired by our work on viability kernel approximation [6], that minimises the cost to reach the target. The principle is to approximate iteratively the capture basins at successive costs c. To compute cost c-capture basin approximation, we use a discrete grid of points covering set $\mathcal{K}$, and label $+1$ the points for which there exists a control leading the point into the $c - \delta c$-capture basin approximation, and -1 otherwise. Then, we use a machine learning method to compute a continuous boundary between +1 and -1 points of the grid. We state the conditions the learning method should fulfil (they are similar to the one established to approximate viability kernels [6]) in order to prove the convergence toward the actual capture basins.

We consider two variants of the algorithm: one provides an approximation that converges from outside, and the other from inside. Although no convergence rate is provided, the comparison of the two approximations gives an assessment of the approximation error for a given problem. Moreover, we define a controller that guarantees to reach the target when derived from the inner approximation.

We consider Support Vector Machines (SVMs [15,14]) as a relevant machine learning technique in this context. Indeed, SVMs provide parsimonious approximations of capture basins, that allow the definition of compact controllers. Moreover, they make possible to use optimisation techniques to find the controls, hence problems with control spaces in more dimensions become tractable. We can also more easily compute the control on several time steps, which improves the quality of the solution for a given resolution of the grid.

We illustrate our approach with some experiments on two simple examples. Finally, we draw some perspectives.

2 Problem Definition

We consider a controlled dynamical system in discrete time (Euler approximation), described by the evolution of its state variable $x \in \mathcal{K} \subset \mathbb{R}^n$. We aim at defining the set of controls to apply to the system starting from point x in order to reach the target $\mathcal{C} \subset \mathcal{K}$:

$$\begin{cases} x(t+dt) = x(t) + \varphi\left(x(t), u(t)\right) \cdot dt, & \text{if } x(t) \notin \mathcal{C} \\ x(t+dt) = x(t), & \text{if } x(t) \in \mathcal{C} \\ u(t) \in \mathcal{U}, \end{cases} \tag{3}$$

where φ is a continuous and derivable function of x and u. The control u must be chosen at each time step in the set $\mathcal{U}$ of admissible controls. We first consider the cost functional $\ell(x(t), u(t)) = 1$, which means that we would like to minimise the time to reach the target. We shall show at the end of this section how the problem can be easily extended to the case of any strictly positive function as a cost.

The capture basin of the system is the set of states for which there exists at least one series of controls such that the system reaches the target in finite time, without leaving $\mathcal{K}$. Let $G\left(x, (u_1, .., u_n)\right)$ be the point reached when applying successively during n time steps the controls $(u_1, .., u_n)$, starting from point x. Let the *minimal time function* (or hitting time function) be the function that associates to a state $x \in \mathcal{K}$ the minimum time to reach $\mathcal{C}$:

$$\vartheta_{\mathcal{C}}^{\mathcal{K}}(x) = \inf\left\{n | \exists (u_1, .., u_n) \in \mathcal{U}^n \text{ such that } G\left(x, (u_1, .., u_n)\right) \in \mathcal{C} \right. \\ \left. \text{and for } 1 \leq j \leq n, G\left(x, (u_1, .., u_j)\right) \in \mathcal{K}\right\} . \tag{4}$$

This is the value function obtained when solving HJB equations in a dynamic programming approach. It takes values in $\mathbb{R}^+ \cup +\infty$, specifically $\vartheta_{\mathcal{C}}^{\mathcal{K}}(x) = 0$ if $x \in \mathcal{C}$ and $\vartheta_{\mathcal{C}}^{\mathcal{K}}(x) = +\infty$ if no trajectory included in $\mathcal{K}$ can reach $\mathcal{C}$. The *capture basin* of $\mathcal{C}$ viable in $\mathcal{K}$ is then defined as:

$$Capt(\mathcal{K}, \mathcal{C}) = \left\{x \in \mathcal{K} | \vartheta_{\mathcal{C}}^{\mathcal{K}}(x) < +\infty\right\} , \tag{5}$$

and we can also define the capture basin in finite time n:

$$Capt(\mathcal{K}, \mathcal{C}, dt, n) = \left\{x \in \mathcal{K} | \vartheta_{\mathcal{C}}^{\mathcal{K}}(x) \leq n\right\}. \tag{6}$$

To solve a target hitting problem in the viability perspective, one must consider the following extended dynamical system $(x(t), y(t))$ when $x(t) \notin \mathcal{C}$, adding variable $y(t)$ which represents the cumulated time over a trajectory:

$$\begin{cases} x(t+dt) = x(t) + \varphi\left(x(t), u(t)\right) \cdot dt \\ y(t+dt) = y(t) + dt. \end{cases} \tag{7}$$

and $(x(t+dt) = x(t), y(t+dt) = y(t))$ when $x(t) \in \mathcal{C}$. [4] prove that approximating minimal time function comes down to a viability kernel approximation problem of this extended dynamical problem. In a viability problem, one must find the rule of controls for keeping a system indefinitely within a constraint set. [2,13] give examples of such an application of viability approach to solve a target hitting problem.

If one wants to minimise a strictly positive cost over the trajectory, variable y becomes the cumulated cost over the trajectory:

$$y(t+dt) = y(t) + \ell(x(t), u(t)) \cdot dt. \tag{8}$$

[3] show that the problem of minimising the cost functional comes down to a viability problem of this extended dynamics for variable y, when $\ell(x(t), u(t)) > 0$. If one defines a variable value for the time step, as follows:

$$dt_\ell = dt \cdot \ell(x(t), u(t)), \tag{9}$$

then, we have:

$$\begin{cases} x(t+dt) = x(t) + \varphi\left(x(t), u(t)\right) \cdot dt = x(t) + \varphi^*\left(x(t), u(t)\right) \cdot dt_\ell \\ y(t+dt) = y(t) + \ell(x(t), u(t)) \cdot dt = y(t) + dt_\ell. \end{cases} \tag{10}$$

where

$$\varphi^*(x, u) = \frac{\varphi(x, u)}{\ell(x, u)}. \tag{11}$$

Hence the problem of minimising a stricly positive cost is equivalent to minimising the time for a slightly different function.

[6] proposed an algorithm, based on [12], that uses a machine learning procedure to approximate viability kernels. The main advantage of this algorithm is that it provides continuous approximations that enable to find the controls with standard optimization techniques, and then to tackle problems with control in large dimensional space. The aim of this paper is to adapt [6] to compute directly an approximation of the capture basin limits, without adding the auxiliary dimension, and then to use these approximations to define the sequence of controls.

3 Machine Learning Approximation of Value Function Contours and Optimal Control

For simplicity, we denote $Capt(\mathcal{K}, \mathcal{C}, dt, n \cdot dt) = Capt^n$. In all the following, continuous sets are denoted by rounded letters and discrete sets in capital letters.

We consider function G:

$$G(x,u) = \begin{cases} x + \varphi(x,u)^* \cdot dt_\ell & \text{if } x \notin \mathcal{C} \\ x & \text{if } x \in \mathcal{C}. \end{cases} \tag{12}$$

We suppose that G is μ-Lipschitz with respect to x:

$$\forall (x,y) \in \mathcal{K}^2, \forall u \in \mathcal{U}, |G(x,u) - G(y,u)| < \mu |x-y|. \tag{13}$$

We define a grid K_h as a discrete subset of $\mathcal{K}$, such that:

$$\forall x \in \mathcal{K}, \exists x_h \in K_h \text{ such that } |x - x_h| \leq h. \tag{14}$$

Moreover, we define an algebraic distance $d_a(x, \partial\mathcal{E})$ of a point x to the boundary $\partial\mathcal{E}$ of a continuous closed set $\mathcal{E}$, as the distance to the boundary when x is located inside $\mathcal{E}$, and this distance negated when x is located outside $\mathcal{E}$:

$$\text{if } x \in \mathcal{E}, d_a(x, \partial\mathcal{E}) = d(x, \partial\mathcal{E}), \tag{15}$$

$$\text{if } x \notin \mathcal{E}, d_a(x, \partial\mathcal{E}) = -d(x, \partial\mathcal{E}). \tag{16}$$

3.1 Capture Basin Approximation Algorithms

In this section, we describe two variants of an algorithm that provides an approximation $\mathcal{C}_h^n$ of the capture basin at time $n \cdot dt_\ell$ of system 12 (and hence the capture basin of cost $n \cdot dt_\ell$ of the original problem), one variant approximates the capture basins from outside and the other one from inside. At each step n of the algorithm, we first build a discrete approximation $C_h^n \subset K_h$ of the capture basin $Capt^n$, and then we use a learning procedure L (for instance Support Vector Machines, as shown below) to generalise this discrete set into a continuous one $\mathcal{C}_h^n$:

$$\mathcal{C}_h^n = L(C_h^n) \tag{17}$$

To simplify the writing, we first define:

$$\overline{C_h^n} = \{x_h \in K_h \text{ s.t. } x_h \notin C_h^n\}, \tag{18}$$

$$\overline{\mathcal{C}_h^n} = \{x \in \mathcal{K} \text{ s.t. } x \notin \mathcal{C}_h^n\}. \tag{19}$$

The two variants differ on the conditions for defining the discrete set C_h^{n+1} from C_h^n, and on the conditions the learning procedure L must fulfil. For both variant, we construct an increasing sequence of approximations at time $n \cdot dt_\ell$, by adding the points of the grid for which there exists at least one control that drives the system not too far away (in an algebraic sense – negative distance in the outer case and positive distance in the inner one) from the boundary of the previous approximation. They also both require conditions on the learning procedure, in order to guarantee the convergence toward the actual capture basin when the step of the grid h decreases to 0. In the inner approximation case, the condition is stricter on set $\mathcal{C}_h^n$ and more relaxed on set $\overline{\mathcal{C}_h^n}$, while the outer case requires converse conditions. We now describe in more details both variants and conditions.

Outer Approximation. The algorithm recursively constructs discrete sets $C_h^{n-1} \subseteq C_h^n \subseteq C_h$ and their continuous generalisation $\mathcal{C}_h^n$ as follows:

Algorithm 1. Outer capture basin approximation algorithm.

```
$n \leftarrow 0$
$C_h^0 \leftarrow \mathcal{C} \cap K_h$
$\mathcal{C}_h^0 \leftarrow \mathcal{C}$
repeat
    $n \leftarrow n + 1$
    $C_h^n \leftarrow C_h^{n-1} \bigcup \left\{ x_h \in \overline{C_h^{n-1}} \text{ such that } \exists u \in \mathcal{U}, d_a(G(x_h, u), \partial\mathcal{C}_h^{n-1}) \geq -\mu h \right\}$
    $\mathcal{C}_h^n \leftarrow L(C_h^n)$
until $C_h^n = C_h^{n-1}$
return $\left\{ \mathcal{C}_h^i \right\}_{0 \leq i \leq n}$
```

Proposition 1. *If the learning procedure L respects the following conditions:*

$$\forall x \in \overline{\mathcal{C}_h^n}, \exists x_h \in \overline{C_h^n} \text{ such that } |x - x_h| \leq h \tag{20}$$

$$\exists \lambda \geq 1 \text{ such that } \forall h, \forall x \in \mathcal{C}_h^n, \exists x_h \in C_h^n \text{ such that } |x - x_h| \leq \lambda h \tag{21}$$

then the convergence of the approximation from outside defined on algorithm 1 is guaranteed:

$$\forall n, Capt^n \subset \mathcal{C}_h^n, \tag{22}$$

$$\mathcal{C}_h^n \to Capt^n \text{ when } h \to 0. \tag{23}$$

Proof. The proof involves two parts.

Part I. First, let us prove by induction that $\forall h > 0, Capt^n \subset \mathcal{C}_h^n$.
By definition, $Capt^0 = \mathcal{C} = \mathcal{C}_h^0$. Suppose that at step n, $Capt^n \subset \mathcal{C}_h^n$. Consider $x \in Capt^{n+1}$. Let us recall that $G(x, u) = x + \varphi(x, u)^* \cdot dt_\ell$ when $x \notin \mathcal{C}$.

Defining $B_h(x, d)$ the set of points of K_h such that $|x - x_h| \leq d$, we can easily show that condition (21) can be rewritten as:

$$B_h(x, h) \subset C_h^n \Rightarrow x \in \mathcal{C}_h^n. \tag{24}$$

By definition, we know that there exists $u \in \mathcal{U}$ such that $G(x, u) \in Capt^n$, which implies that for all $x_h \in B_h(x, h)$, $d\left(G(x_h, u), Capt^n\right) \leq \mu h$, because G is μ-Lipschitz.

Moreover, for all $x_h \in B_h(x, h)$, $d(G(x_h, u), \mathcal{C}_h^n) \leq \mu h$, because, by hypothesis, $Capt^n \subset \mathcal{C}_h^n$. Thus $x_h \in C_h^{n+1}$. Therefore, $x \in \mathcal{C}_h^{n+1}$ (because of condition (24)).

We can thus conclude $Capt^{n+1} \subset \mathcal{C}_h^{n+1}$.

Part II. Now, we prove by induction that for any n, $\mathcal{C}_h^n \to Capt^n$ when $h \to 0$.
Suppose now that for a given value n, $\mathcal{C}_h^n \to Capt^n$ when $h \to 0$.
Because $Capt^n \subset \mathcal{C}_h^n$, we have:

$$\forall x \in \mathcal{K} \mid x \notin Capt^n, \exists h > 0 \mid x \notin \mathcal{C}_h^n. \tag{25}$$

Now, consider $x \in \mathcal{K}$ such that $x \notin Capt^{n+1}$.

This implies that for all $u \in \mathcal{U}$ such that $d\left(G(x, u), Capt^n\right) > 0$. One can choose $h' > 0$ and h such that for all $u \in \mathcal{U}$, $d\left(G(x, u), Capt^n\right) > h' + \mu\lambda h$.
Condition (20) can be rewritten as:

$$B_h(x, \lambda h) \subset \overline{C_h^n} \Rightarrow x \in \overline{\mathcal{C}_h^n}. \tag{26}$$

In this case, for all $x_h \in B_h(x, \lambda h)$, all $u \in \mathcal{U}$, $d\left(G(x_h, u), Capt^n\right) > h'$, because G is μ-Lipschitz.

Since $\mathcal{C}_h^n \to Capt^n$ when $h \to 0$, there exists h, such that, for all $x_h \in B_h(x, \lambda h)$, and all $u \in \mathcal{U}$, $G(x_h, u) \in \overline{\mathcal{C}_h^n}$, hence $x_h \in \overline{C_h^n}$. Hence, there exists h such that $x \notin \mathcal{C}_h^n$ (because of condition 24).

Therefore $\mathcal{C}_h^{n+1} \to Capt^{n+1}$ when $h \to 0$.

Conclusion. $Capt^n \subset \mathcal{C}_h^n$ and $\mathcal{C}_h^n \to Capt^n$ then $\mathcal{C}_h^n$ is an outer approximation of the capture basin at time $n \cdot dt_\ell$, which tends to the actual capture basin when the resolution of the grid h tends to 0. □

Inner Approximation. We consider the following algorithm:

Algorithm 2. Inner capture basin approximation algorithm.

```
$n \leftarrow 0$
$C_h^0 \leftarrow \mathcal{C} \cap K_h$
$\mathcal{C}_h^0 \leftarrow \mathcal{C}$
repeat
  $n \leftarrow n + 1$
  $C_h^n = C_h^{n-1} \bigcup \left\{ x_h \in \overline{C_h^{n-1}} \text{ such that } \exists u \in \mathcal{U}, d_a(G(x_h, u), \partial\mathcal{C}_h^n) \geq \mu h \right\}$
  $\mathcal{C}_h^n \leftarrow L(C_h^n)$
until $C_h^n = C_h^{n-1}$
return $\left\{\mathcal{C}_h^i\right\}_{0 \leq i \leq n}$
```

Proposition 2. *If the learning procedure L respects the following conditions:*

$$\forall x \in \mathcal{C}_h^n, \exists x_h \in C_h^n \textit{ such that } |x - x_h| \leq h \tag{27}$$

$$\exists \lambda \geq 1 \textit{ such that } \forall h, \forall x \in \overline{\mathcal{C}_h^n}, \exists x_h \in \overline{C_h^n} \textit{ such that } |x - x_h| \leq \lambda h \tag{28}$$

and that the dynamics are such that:

$$\forall x \in \mathcal{K} \textit{ with } d_a\left(x, \partial Capt^n\right) > 0, \exists u \in \mathcal{U} \textit{ such that } d_a\left(G(x,u), \partial Capt^{n-1}\right) > 0 \tag{29}$$

then the convergence of the approximation from inside defined on algorithm 2 is guaranteed:

$$\forall n, \mathcal{C}_h^n \subset Capt^n, \tag{30}$$

$$\mathcal{C}_h^n \to Capt^n \textit{ when } h \to 0. \tag{31}$$

Proof. Convergence proof of the algorithm from inside requires an additional condition on the dynamics (eq. (29)): a point x of the interior of capture basin at time $n \cdot dt_\ell$, should be such that there exists $y \in G(x)$ belonging to the interior of capture basin at time $(n-1) \cdot dt_\ell$ (and not on $\partial Capt^{n-1}$).

Part I. We begin to show by induction that $\mathcal{C}_h^n \subset Capt^n$.
Suppose that $\mathcal{C}_h^n \subset Capt^n$ and consider $x \in \mathcal{C}_h^{n+1}$.
Because of condition (27):

$$\exists x_h \in C_h^{n+1} \text{ such that } |x - x_h| < h.$$

By definition of C_h^{n+1}:

$$\exists u \in U \text{ such that } d_a\left(G(x_h,u), \mathcal{C}_h^n\right) > \mu h.$$

By hypothesis of induction $\mathcal{C}_h^n \subset Capt^n$, hence : $d_a\left(G(x_h,u), Capt^n\right) > \mu h$. By hypothesis on G, $|G(x_h,u) - G(x,u)| < \mu |x_h - x|$, hence $d_a\left(G(x,u), Capt^n\right) > 0$. Therefore $x \in Capt^{n+1}$. Thus $\mathcal{C}_h^{n+1} \subset Capt^{n+1}$.

Part II. We prove by induction that, when $h \to 0$, $\mathcal{C}_h^n \to Capt^n$.
Suppose that $\mathcal{C}_h^n \to Capt^n$ when $h \to 0$.
Because $\mathcal{C}_h^n \subset Capt^n$, we have:

$$\forall x \in Capt^n \mid d_a(x, \partial Capt^n) > 0, \exists h > 0 \mid x \in \mathcal{C}_h^n.$$

We use the rewriting of condition (28):

$$B_h(x, \lambda h) \cap C_h^n \Rightarrow x \in \mathcal{C}_h^n. \tag{32}$$

Consider $x \in Capt^{n+1}$ such that $d_a(x, \partial Capt^{n+1}) > 0$. One can choose h such that $d_a(x, \partial Capt^{n+1}) > (\mu + \lambda)h$. With such a choice, for each $x_h \in B_h(x, \lambda h)$, $d_a(x_h, \partial Capt^{n+1}) > \mu h$, hence, there exists $u \in \mathcal{U}$ such that $d_a\left(G(x_h,u), Capt^n)\right) > 0$ (because G is μ-Lipschitz).

By induction hypothesis, there exists h such that $d_a\left(G(x_h,u), \mathcal{C}_h^n\right) > \mu h$, hence $x_h \in C_h^{n+1}$. Taking the smallest value of h this is true for all $x_h \in B_h(x, \lambda h)$. Therefore $x \in \mathcal{C}_h^{n+1}$ (because of condition (32)).
Therefore $\mathcal{C}_h^{n+1} \to Capt^{n+1}$ when $h \to 0$.

Conclusion. $\mathcal{C}_h^n \subset Capt^n$ and $\mathcal{C}_h^n \to Capt^n$ then $\mathcal{C}_h^n$ is an inner approximation of the capture basin in finite time $n \cdot dt_\ell$, which tends to the actual capture basin when the resolution of the grid tends to 0. □

3.2 Optimal Control

The aim of the optimal controller is to choose a control function that reaches the target in minimal time (or cost), without breaking the viability constraints. The idea is to choose the controls which drive the system to cross $\mathcal{C}_h^n$ boundaries in a descending order.

Algorithm 3. Optimal controller.

```
Require: $x(0) \in \mathcal{K}$ and $\notin \mathcal{C}$
Require: $n$ such that $x(0) \in \mathcal{C}_h^n$ and $\notin \mathcal{C}_h^{n-1}$.
  for $i = 1$ to $n$ do
    compute $u(i)^*$ such that $G(x(i-1), u(i)^*) \in \mathcal{C}_h^{n-i}$
    $x(i) = G(x(i-1), u(i)^*)$
  end for
  return $\{u^*(i)\}_{1 \leq i \leq n}$
```

Proposition 3. *The procedure described in algorithm 3 converges towards the control policy minimizing the hitting time, when h and dt tend to 0.*

Proof. By construction, if $x(i) \in \mathcal{C}_h^{n-i}$ and $x(i) \notin \mathcal{C}_h^{n-i-1}$, there exists a control value $u^*(i+1)$ such that $x(i+1) = G(x(i), u^*(i+1)) \in \mathcal{C}_h^{n-i-1}$ (see proof of the convergence of the inner algorithm, part I.). Therefore, the procedure leads to the target in $n+1$ time steps, i.e. with a time $(n+1) \cdot dt_\ell$. Moreover, by definition, the optimum time for reaching the target from a point x located on the boundary of capture basin $Capt^n$ is $n \cdot dt_\ell$. Hence, the optimum time for reaching the target from point x such that $x \in Capt^{n+1}$ and $x \notin Capt^n$, with the dynamics defined by φ, is between $n \cdot dt_\ell$ and $(n+1) \cdot dt_\ell$. Then, the fact that $\mathcal{C}_h^n$ converges to $Capt^n$ when h tends to 0, ensures that the number of steps needed by the procedure applied to this point x will tend to $(n+1) \cdot dt_\ell$. When dt_ℓ tends to 0, the difference with the optimum time to reach the target, which is smaller than dt_ℓ, tends to 0. □

4 Experiments

4.1 SVM as a Learning Procedure

We use Support Vector Machines [15,14] as the learning procedure L to define capture basin approximations $\mathcal{C}_h^n = L(C_h^n)$. At each iteration, we construct a learning set: let S_h^n be a set of couples (x_h, y_h), where $x_h \in K_h$ and $y_h = +1$ if $x_h \in C_h^n$ and -1 otherwise. Running a SVM on learning sets S_h^n provides a separating function f_h^n between points of different labels and hence, allows us to define a continuous approximation $\mathcal{C}_h^n$ of C_h^n as follows:

$$\mathcal{C}_h^n = \{x \in K \text{ such that } f_h^n(x) \geq 0\}\,. \tag{33}$$

Points x of the boundary $\partial\mathcal{C}_h^n$ are those such that $f_h^n(x) = 0$. The fulfilment of the conditions guaranteeing convergence is discussed in [6] and the same arguments hold in both variants of the algorithm.

In the following examples, we use libSVM [5] to train the SVMs. As we did in [6], we use the SVM function as a proxy for the distance to the boundary, in order to simplify the computations.

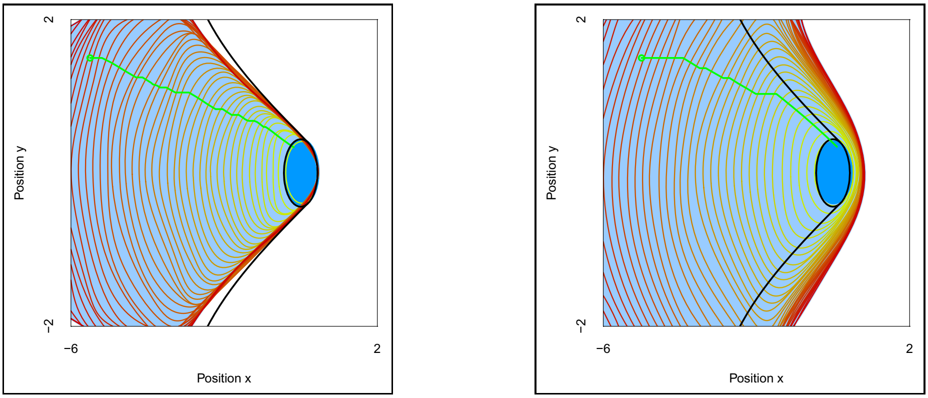


Fig. 1. Approximation from inside (left) and from outside (right) for Zermelo problem. The horizontal axis represents the position x and the vertical one position y. $\mathcal{K}$ is the rectangle. The capture basin is represented in blue. The black thick line limits the boundary of the actual capture basin. The level lines represent approximation of the contours of the capture basins for successive time steps. The grid contains 41 points by dimension. The optimisation is made on 4 time steps, with $dt = 0.05$. Each figure presents an example of trajectory (in green) using the SVM optimal controller.

4.2 Zermelo Problem

The state $(x(t), y(t))$ of the system represents the position of a boat in a river. There are two controls: the thrust u and the direction θ of the boat. The system in discrete time defined by a time interval dt can be written as follows:

$$\begin{cases} x(t+dt) = x(t) + (1 - 0.1y(t)^2 + u\cos\theta) \cdot dt \\ y(t+dt) = y(t) + (u\sin\theta) \cdot dt, \end{cases} \tag{34}$$

where $u \in [0; 0.44]$ and $\theta \in [0; 2\pi]$. The boat must remain in a given interval $\mathcal{K} = [-6; 2] \times [-2; 2]$, and reach a round island $\mathcal{C} = B(0; 0.44)$ in minimal time. We suppose that the boat must reach the island before time $T = 7$.

For this simple system, it is possible to derive analytically the capture basin, hence we can compare the approximations given by the two algorithms with the actual capture basin. Figure 1 compares some results obtained with the outer and inner approximation. In any cases, the quality of the approximation can be assessed by comparing both approximations: by construction, the contour of the actual capture basin is surrounded by inner and outer approximations. A example of a optimal trajectory defined with the optimal controller is also presented: with the inner approximation, the trajectory will enable the boat to reach the target, while it is not guaranteed in the outer case.

4.3 Mountain Car

We consider the well-known car on the hill problem. The state is two-dimensional (position and velocity) and the system can be controlled with a continuous one-dimensional

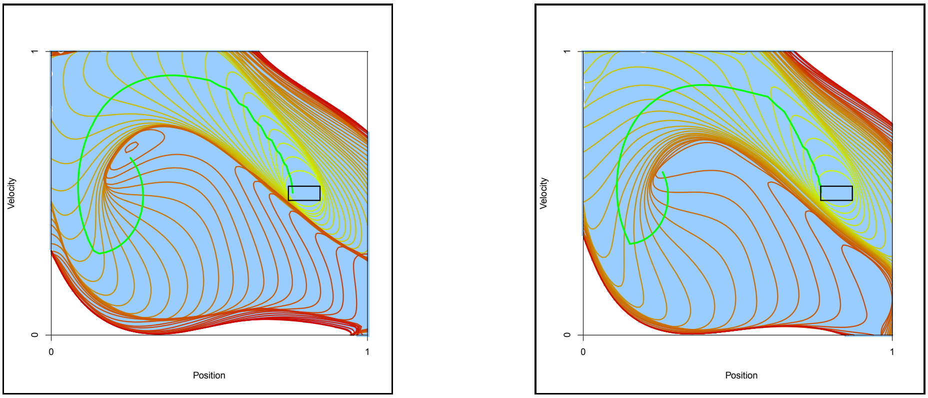


Fig. 2. Inner (left) and outer (right) approximation for the car on the hill problem. The grid contains 61 points by dimension, the optimisation is made on 2 time steps, and $dt_\ell = 0.04$. Example of optimal trajectories are depicted in green.

control (thrust). For a description of the dynamics and the state space constraints, one can refers to [11]. The aim of the car on the hill system is to keep the car inside a given set of constraints, and to reach a target (the top of the hill) as fast as possible. The interesting characteristics of the problem is that the car has limited power, the acceleration can not overcome the gravitational force and hence the car may have to first go away from the solution before reverse up the hill. We add a cost function to the definition of the problem, that contains two parts: the first one represents the time to reach the target, the second part is function of the control (we make the simple assumption that we want to minimise the logarithm of the thrust a as a high thrust may lead to an increase of the fuel consumption for instance):

$$\ell(x(t), u(t)) = 1 + \log(|a(t)| + 1). \tag{35}$$

Figure 2 shows the approximation of the contours of the value function using outer and inner variants of the algorithm, with examples of optimal trajectories.

5 Discussion

We proposed an algorithm that approximates capture basins and contours of value function, using a classification procedure, in two variants (inner and outer). The inner approximation can be used to define a optimal controller that guarantees to find a series of controls that allows the system to reach the target. SVMs appear as a particularly relevant classification procedure for this approach, because they provide parsimonious representations of the capture basins and enable to use optimization techniques to compute the controls. This latter point is particularly important to deal with high dimensional control space. The parsimonious property may allow to define compact and fast controller, even for high dimensional state space. However, although we generally manage to find parameters in which the result respect the conditions of convergence, this is not guaranteed. Therefore, considering other learning algorithms that would be even more appropriate seems a relevant research direction.

For now, the method proposed here can only be used to solve problems with deterministic dynamics. A second direction of research is to investigate the behaviour of the optimal controller when there is some uncertainty on the state or the control.

References

1. Aubin, J.P.: Viability theory. Birkhäuser (1991)
2. Bayen, A.M., Crück, E., Tomlin, C.J.: Guaranteed Overapproximations of Unsafe Sets for Continuous and Hybrid Systems: Solving the Hamilton-Jacobi Equation Using Viability Techniques. In: Tomlin, C.J., Greenstreet, M.R. (eds.) HSCC 2002. LNCS, vol. 2289, pp. 90–104. Springer, Heidelberg (2002)
3. Cardaliaguet, P., Quincampoix, M., Saint-Pierre, P.: Optimal times for constrained nonlinear control problems without local optimality. Applied Mathematics & Optimization 36, 21–42 (1997)
4. Cardaliaguet, P., Quincampoix, M., Saint-Pierre, P.: Set-Valued Numerical Analysis for Optimal control and Differential Games. Annals of the International Society of Dynamic Games (1998)
5. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines (2001)
6. Deffuant, G., Chapel, L., Martin, S.: Approximating viability kernels with support vector machines. IEEE Transactions on Automatic Control 52(5), 933–937 (2007)
7. Frankowska, H.: Optimal trajectories associated with a solution of the contingent hamilton-jacobi equation. Applied Mathematics and Optimization 19(1), 291–311 (1989)
8. Lhommeau, M., Jaulin, L., Hardouin, L.: Inner and outer approximation of capture basin using interval analysis. In: 4th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2007 (2007)
9. Lygeros, J.: On reachability and minimum cost optimal control. Automatica 40, 917–927 (2004)
10. Mitchell, I., Bayen, A., Tomlin, C.: A time-dependent Hamilton-Jacobi formulation for reachable sets for continuous dynamic games. IEEE Transactions on Automatic Control 50(7), 947–957 (2005)
11. Moore, A., Atkeson, C.: The parti-game algorithm for variable resolution reinforcement learning in multidimensional state-spaces. Machine Learning 21, 199–233 (1995)
12. Saint-Pierre, P.: Approximation of the viability kernel. Applied Mathematics & Optimization 29(2), 187–209 (1994)
13. Saint-Pierre, P.: Approche ensembliste des systèmes dynamiques, regards qualitatifs et quantitatifs. Société de Mathématiques Appliquées et Industrielles, 66 (2001)
14. Scholkopf, B., Smola, A.: Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond. MIT Press, Cambridge (2002)
15. Vapnik, V.: The nature of statistical learning theory. Springer (1995)

Part II
Robotics and Automation

Dynamic Obstacle Avoidance with Simultaneous Translational and Rotational Motion Control for Autonomous Mobile Robot

Masaki Takahashi, Takafumi Suzuki, Tetsuya Matsumura, and Ayanori Yorozu

Dept. of System Design Engineering,
Keio University
3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan
takahashi@sd.keio.ac.jp

Abstract. This paper presents a real-time collision avoidance method with simultaneous control of both translational and rotational motion with consideration of a robot width for an autonomous omni-directional mobile robot. In the method, to take into consideration the robot's size, a wide robot is regarded as a capsule-shaped case not a circle. With the proposed method, the wide robot can decide the direction of translational motion to avoid obstacles safely. In addition, the robot can decide the direction of the rotational motion in real time according to the situation to perform smooth motion. As an example of design method of the proposed method, novel control method based on the fuzzy potential method is proposed. To verify its effectiveness, several experiments using a real robot are carried out.

Keywords: Service Robot, Obstacle Avoidance, Omni-directional Platform, Fuzzy Potential Method.

1 Introduction

Various obstacle avoidance methods and their availabilities for mobile robots have described [1]-[8]. Most of these studies regard the robots as points or circles and discuss control methods of translational motion. In these studies, a non-circle robot is regarded as a circle robot with consideration of maximum size of the robot. The effectiveness of avoiding obstacles by this approach has been confirmed. However, depending on the shape of the robot, this approach reduces and wastes available free space and can decrease the possibility that the robot reaches the goal. If wide robots, which are horizontally long, are regarded as circles in accordance with conventional approaches, they may not be able to go between two objects due to the largest radius of the robot, even if they ought to be able to go through by using their shortest radius. This suggests the necessity of a suitable orientation angle at the moment of avoidance. Consequently, to enable wide robots to avoid obstacles safely and efficiently, it is necessary to control not only a translational motion but also a rotational motion. In our current research, a wide robot with omni-directional platforms shown in Fig.1 is developed.

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 51–64.
springerlink.com

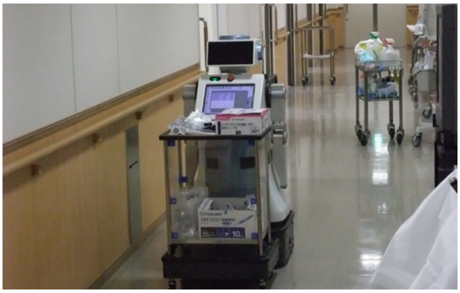


Fig. 1. An autonomous robot for hospital use

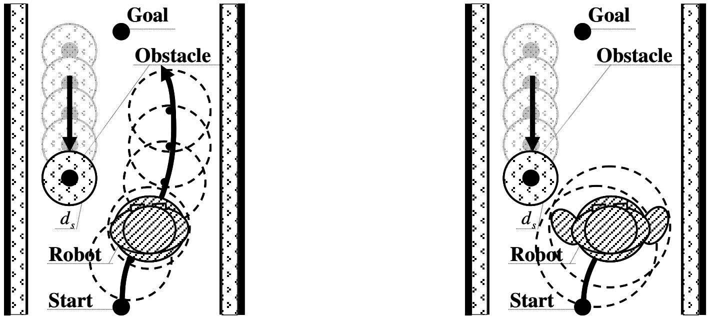
Several studies have focused on the orientation angle of the robot [9]; [10]. In these studies, by convolving the robot and the obstacle at every orientation and constructing the C-space, the suitable orientation angles of the robot for path planning are decided. However, these methods require an environmental map and the studies have not shown the effectiveness for avoidance of unknown obstacles by autonomous mobile robots. Therefore, to avoid unknown obstacles reactively with consideration of the orientation angle, wide robots need an algorithm that can decide the orientation angle and rotational velocity command in real time based on current obstacle information.

This study proposes a control method of both translational and rotational motion with consideration of a robot width in order to achieve a smooth motion. With the proposed method, the orientation angle is controlled easily in real time. To verify the effectiveness of the proposed method, several simulations were carried out [11]. In this study, several experiments using our robot shown in Fig.1 are carried out.

2 Simultaneous Translational and Rotational Motion Control

2.1 Problem for Solution

There are various non-circle robots. These are vertically long robots, or wide robots. These robots have two arms mounted on a torso with wheels so these robots can be used for mobility, manipulation, whole-body activities, and human-robot interaction [12]; [13]. For these wide robots, conventional obstacle avoidance methods are incompatible because they regard the robot as a point or a circle. We are developing a wide robot with a torso, two arms and a head shown in Fig.1. It not only moves indoors but also communicates and interacts with humans through gestures or speech. When the robot opens one or both of its arms slightly, as shown in Fig.2(b), it becomes increasingly difficult to apply conventional obstacle avoidance methods. If these wide robots are regarded as circles in accordance with conventional approaches, it may not be possible for them to go between two obstacles due to the largest radius of the robot, even if they ought to go through by using their shortest radius. In this study, a capsule-shaped case is introduced to make wide robots move smoothly and safely in an environment with obstacles.



(a) Situation A (b) Situation B

Fig. 2. Two robots which are included in respective circles

2.2 Design of Capsule-Shaped Case

The capsule-shaped case is modeled by two circles and two lines tangent to the circles as shown in Fig.3. This closed contour is defined as $l(\phi)$ with the origin at the point P_O.

$$l(\phi)=\begin{cases} C_a/\cos\phi & if \begin{array}{c} 0\le\phi<\phi_1 \\ \phi_4\le\phi<2\pi \end{array} \\ -C_a/\cos\phi & if\ \phi_2\le\phi<\phi_3 \\ \sqrt{X(\phi)^2+Y(\phi)^2} & if \begin{array}{c} \phi_1\le\phi<\phi_2 \\ \phi_3\le\phi<\phi_4 \end{array} \end{cases} \tag{1}$$

where ϕ_i is clockwise from the back direction of the robot.

$$\phi_1=\arctan(C_L/C_a),\ \phi_2=\pi-\arctan(C_L/C_a),$$
$$\phi_3=\pi+\arctan(C_R/C_a),\ \phi_4=2\pi-\arctan(C_R/C_a)$$

$X(\phi)$ and $Y(\phi)$ are calculated as follows.

$$X(\phi)=\begin{cases} \dfrac{-C_L-\sqrt{C_L{}^2-\left(C_L{}^2-C_a{}^2\right)\left\{1+\tan^2\left(\pi/2-\phi\right)\right\}}}{1+\tan^2\left(\pi/2-\phi\right)} \\ if \quad \phi_1\le\phi<\phi_2 \\ \dfrac{C_R+\sqrt{C_R{}^2-\left(C_R{}^2-C_a{}^2\right)\left\{1+\tan^2\left(\pi/2-\phi\right)\right\}}}{1+\tan^2\left(\pi/2-\phi\right)} \\ if \quad \phi_3\le\phi<\phi_4 \end{cases} \tag{2}$$

$$Y(\phi)=X(\phi)\tan\left(\pi/2-\phi\right) \tag{3}$$

In the proposed method, C_L, C_R ,and C_a are decided in a way that makes wide robot shape fall within the capsule-shaped case.

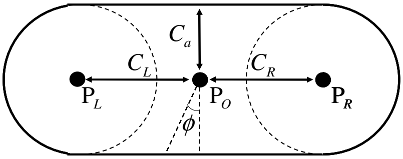


Fig. 3. Capsule-shaped case

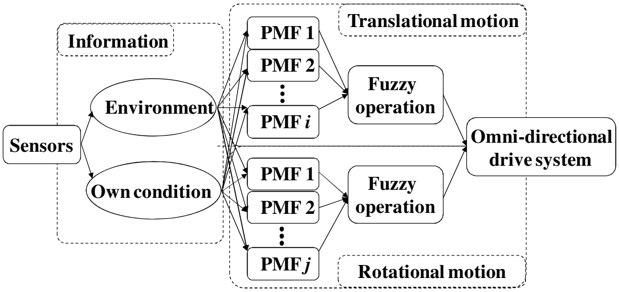


Fig. 4. Concept of fuzzy potential method using both translational and rotational motion with an omni-direction platform

2.3 Controller Design

Figure 4 shows a concept of the fuzzy potential method (FPM) that takes into consideration both translational and rotational motion. In the conventional FPM [14], a command velocity vector that takes into consideration element actions is decided. Element actions are represented as potential membership functions (PMFs), and then they are integrated by means of fuzzy inference. The horizontal axis of PMF is directions which are from $-\pi$ to π radians measured clockwise from the front direction of the robot. The vertical axis of PMF is the grade for the direction. The grade, direction, and configured maximum and minimum speeds, are used to calculate the command velocity vector.

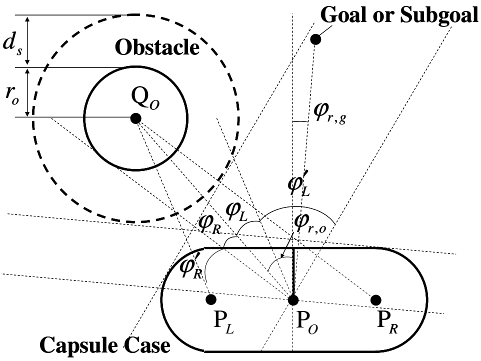


Fig. 5. Wide robot and obstacle

In this research, in addition to conventional approach the PMFs for translational and rotational motion are designed respectively based not only on environmental information but also the robot's condition. Environmental information and the robot's condition are treated separately and divided into a translation problem and a rotational problem. Then the PMFs of each problem are independently integrated using fuzzy inference. Finally, translational and rotational velocity commands, which are calculated by defuzzification of mixed PMFs, are realized by an omni-directional drive system.

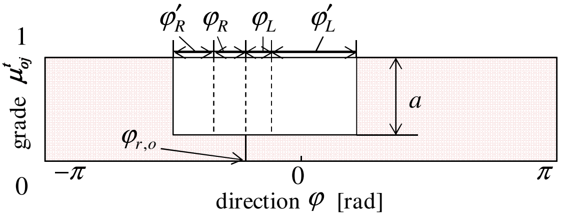


Fig. 6. Example of PMF for an obstacle

2.4 PMF for Translational Motion

2.4.1 PMF for Obstacles

To enable a wide robot to avoid obstacles safely and efficiently in real time, a concave shaped PMF μ^t_{oj} $(j = 1, 2, \cdots, n)$ shown in Fig.6, which takes into consideration the capsule case, is generated. This PMF is specified by depth and width, which are calculated based on the geometrical relation between an obstacle and the robot as shown in Fig.5. By generating a PMF based on the variables φ_L, φ_R, φ'_L, φ'_R, a and $\varphi_{r,o}$ in Fig.6, it can choose a safe direction.

$$\varphi_L = \arccos\left(\frac{\left\|\overrightarrow{\mathrm{P}_O\mathrm{Q}_O}\right\|^2 + \left\|\overrightarrow{\mathrm{P}_L\mathrm{Q}_O}\right\|^2 - \left\|\overrightarrow{\mathrm{P}_O\mathrm{P}_L}\right\|^2}{2\left\|\overrightarrow{\mathrm{P}_O\mathrm{Q}_O}\right\| \cdot \left\|\overrightarrow{\mathrm{P}_L\mathrm{Q}_O}\right\|} \right). \tag{4}$$

$$\varphi_R = \arccos\left(\frac{\left\|\overrightarrow{\mathrm{P}_O\mathrm{Q}_O}\right\|^2 + \left\|\overrightarrow{\mathrm{P}_R\mathrm{Q}_O}\right\|^2 - \left\|\overrightarrow{\mathrm{P}_O\mathrm{P}_R}\right\|^2}{2\left\|\overrightarrow{\mathrm{P}_O\mathrm{Q}_O}\right\| \cdot \left\|\overrightarrow{\mathrm{P}_R\mathrm{Q}_O}\right\|} \right). \tag{5}$$

$$\varphi'_L = \begin{cases} \arcsin\left(D \big/ \left\|\overrightarrow{\mathrm{P}_L\mathrm{Q}_O}\right\| \right) & \mathit{if}\ D < \left\|\overrightarrow{\mathrm{P}_L\mathrm{Q}_O}\right\|. \\ \pi - \arcsin\left\{ \left(\left\|\overrightarrow{\mathrm{P}_L\mathrm{Q}_O}\right\| - d_s \right) \big/ \left(D - d_s \right) \right\} & \mathit{if}\ D \geq \left\|\overrightarrow{\mathrm{P}_L\mathrm{Q}_O}\right\|. \end{cases} \tag{6}$$

$$\varphi'_R = \begin{cases} \arcsin\left(D \big/ \left\|\overrightarrow{\mathrm{P}_R\mathrm{Q}_O}\right\| \right) & \mathit{if}\ D < \left\|\overrightarrow{\mathrm{P}_R\mathrm{Q}_O}\right\|. \\ \pi - \arcsin\left\{ \left(\left\|\overrightarrow{\mathrm{P}_R\mathrm{Q}_O}\right\| - d_s \right) \big/ \left(D - d_s \right) \right\} & \mathit{if}\ D \geq \left\|\overrightarrow{\mathrm{P}_R\mathrm{Q}_O}\right\|. \end{cases} \tag{7}$$

As a measure to decide how far the robot should depart from the obstacle, a is defined as the depth of the concave PMF.

$$a = \frac{\alpha - \left\| \mathbf{r}_{r,o} \right\|}{\alpha - D} \qquad if \ \left\| \mathbf{r}_{r,o} \right\| < \alpha \ . \tag{8}$$

where $\mathbf{r}_{r,o} = (\mathrm{r}_x, \mathrm{r}_y)$ is the current position vector of the obstacle relative to the robot. If the current obstacle position is inside a circle with radius α from the robot position, a PMF for obstacle avoidance is generated. D is decided to ensure a safe distance.

$$D = C_a + r_o + d_s \ . \tag{9}$$

C_a is the minimum size of the capsule case, r_o and d_s denote respectively the radius of the obstacle and the safe distance. $\varphi_{r,o}$ is the angle of the direction to the obstacle relative to the robot.

$$\varphi_{r,o} = \arctan(\mathrm{r}_y / \mathrm{r}_x) \ . \tag{10}$$

For safe avoidance, the PMF μ_{oj}^t is generated for all the obstacles that the robot has detected. Then, they are all integrated by calculating the logical product μ_o^t.

$$\mu_o^t = \mu_{o1}^t \wedge \mu_{o2}^t \wedge \ \cdots \ \wedge \mu_{oj}^t \ . \tag{11}$$

By deciding the depth and the base width of the concave PMF μ_o^t is generated.

2.4.2 PMF for a Goal

To head to the goal, a triangular PMF μ_g^t is generated, as shown in Fig.7. μ_g^t is specified by g_a, g_b ,and $\varphi_{r,g}$. As a measure to decide how close to the goal the robot should go, g_a is defined as the height of the triangular PMF. As a measure to decide how much the robot can back away from obstacles, g_b is defined. μ_g^t

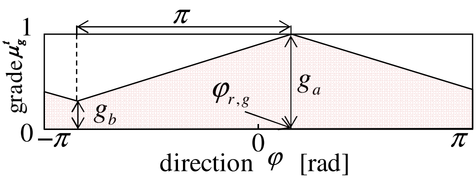


Fig. 7. Example of PMF for a goal

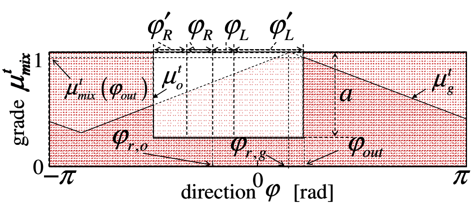


Fig. 8. Example of mixed PMF for translational motion

reaches the maximum value as g_a at an angle of the goal direction relative to the front direction of the robot $\varphi_{r,g}$.

$$g_a = \begin{cases} \dfrac{\left\|\mathbf{r}_{r,g}\right\|}{\varepsilon} & if \quad \left\|\mathbf{r}_{r,g}\right\| \le \varepsilon \\ 1.0 & if \quad \left\|\mathbf{r}_{r,g}\right\| > \varepsilon \end{cases} . \tag{12}$$

$$g_b = \eta g_a \quad (0 \le \eta < 1) . \tag{13}$$

where $\left\|\mathbf{r}_{r,d}\right\|$ is an absolute value of the position vector of the goal relative to the robot. ε and η are constants. If $\left\|\mathbf{r}_{r,d}\right\|$ is below ε, g_a is defined. The robot can decelerate and stop stably.

2.4.3 Calculation of a Translational Command Velocity Vector

The proposed method uses fuzzy inference to calculate the command velocity vector. The PMFs μ_o^t and μ_g^t are integrated by fuzzy operation into a mixed PMF μ_{mix}^t as shown in Fig.8. μ_{mix}^t is an algebraic product of μ_o^t and μ_g^t.

$$\mu_{mix}^t = \mu_g^t \wedge \mu_o^t . \tag{14}$$

By defuzzifier, a velocity command vector is calculated as a traveling direction φ_{out} and an absolute value of the reference speed of the robot based on the mixed PMF μ_{mix}^t. φ_{out} is decided as the direction that makes the PMF $\mu_{mix}^t(\varphi)$ maximum. Based on φ_{out}, v_{out} is calculated as follows.

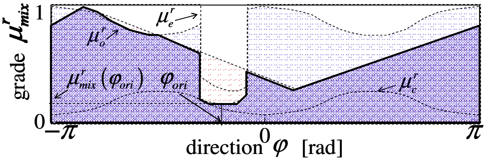


Fig. 9. Example of mixed PMF for rotational motion

$$v_{out} = \mu_{mix}^t(\varphi_{out})(v_{max} - v_{min}) + v_{min} . \tag{15}$$

where $\mu_{mix}^t(\varphi_{out})$ is the mixed PMF for translational motion corresponding to the φ_{out}. v_{max} and v_{min} are respectively the upper and lower limits of the robot speed.

$$\mu_o^r = \mu_e^r - \mu_c^r . \tag{16}$$

2.5 PMF for Rotational Motion

2.5.1 PMF for Obstacles

To enable a wide robot to decide the appropriate angle of the direction for obstacle avoidance in real time, PMF μ_o^r is generated. μ_e^r is generated based on the distance from the center of the robot to obstacles corresponding to all directions, as shown in Fig. 9. The relative distances are obtained with range sensors such as laser range finder, ultra sonic sensors or infrared sensors. μ_c^r is generated based on the capsule case.

$$\mu_c^r(\varphi) = \frac{l(\varphi+\pi)}{\alpha} . \tag{17}$$

The aim of the PMF μ_o^r is to search for an orientation angle of the robot that would maximize the distance between a point on capsule case and each obstacle by turning the front or back side of the robot. By using the capsule case, a PMF design can deal with the width of the robot for rotational motion.

2.5.2 PMF for a Goal

In order to turn the front of the robot toward the goal direction or the travelling direction if there is no obstacle to avoid, PMF for a goal is generated as μ_g^r. This shape is decided in same way as μ_g^t.

2.5.3 Calculation of a Rotational Command Velocity

For the rotational motion, like the translational motion, the rotational command velocity is derived. The PMFs μ_e^r and μ_g^r are integrated by fuzzy operation into a mixed PMF μ_{mix}^r, as shown in Fig.9.

$$\mu_{mix}^r = \mu_g^r \wedge \mu_o^r . \tag{18}$$

By defuzzifier, the command velocity is calculated as a rotational direction φ_{ori} and an absolute value of the reference speed of the robot. φ_{ori} is decided as the direction φ_i that makes the following function $h(\varphi)$ minimum.

$$h(\varphi) = \int_{\varphi-\zeta}^{\varphi+\zeta} \mu_{mix}^r(\psi)\, d\psi \tag{19}$$

where ζ is the parameter to avoid choosing an uncertainty φ_i caused by, for example, noise on the sensor data. On the basis of φ_{ori}, ω is calculated.

$$\omega = \omega_a \operatorname{sgn}(\varphi_{ori}) . \tag{20}$$

where ω_a is design variable.

2.6 Omni-Directional Platform

An omni-directional platform was used for the autonomous mobile robot's motion. The command velocity vector was realized by four DC motors and omni wheels.

$$v_r^x = v_{out} \cos \varphi_{out} \quad . \tag{21}$$

$$v_r^y = v_{out} \sin \varphi_{out} \tag{22}$$

where v_{out} and ω are respectively command translational velocity vector and rotational velocity.

$$\begin{bmatrix} v_1^w \\ v_2^w \\ v_3^w \\ v_4^w \end{bmatrix} = \begin{bmatrix} \cos\delta & \sin\delta & R \\ -\cos\delta & \sin\delta & R \\ -\cos\delta & -\sin\delta & R \\ \cos\delta & -\sin\delta & R \end{bmatrix} \begin{bmatrix} v_r^x \\ v_r^y \\ \omega \end{bmatrix} \tag{23}$$

δ is an angle of gradient for each wheel. R is half the distance between two diagonal wheels. v_i^w is a command velocity of each i-th wheel.

Table 1. Parameters in experiments

L	0.4 m	ε	1.0 m	D	0.9 m
C_a	0.3 m	ω_{max}	1.0 rad/s	η	0.2
C_R	0.3 m	W	1.0 m	a_r	1.0 m/s^2
d_s	0.3 m	C_L	0.3 m	ω_{min}	0.0 rad/s
α	4.0 m	r_a	0.3 m		

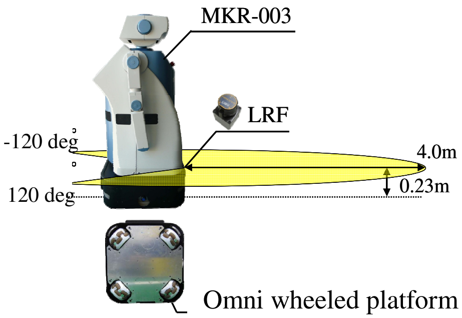
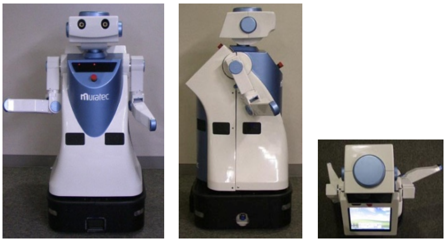


Fig. 10. Laser sensor and an omni-directional platform on an autonomous mobile robot



(a) Front (b) Side (c) Top

Fig. 11. Experimental situation

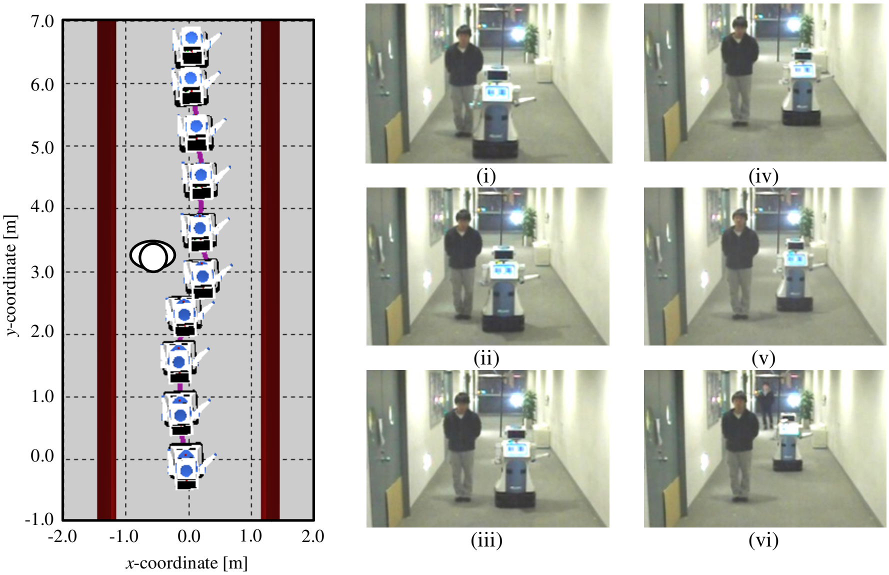
3 Experimental Results

3.1 Experimental Condition

To verify the performance of the proposed collision avoidance method to static obstacles, an experiment using the real robot were carried out. In order to recognize the environment, as shown in Fig.10, the robot has external sensors, such as a stereo camera, laser range finder and ultrasonic sensors. However, in this research the robot recognizes the environment using only laser range finder. The upper limit of the velocity of the robot was 0.50 m/s. The upper limit of the acceleration of the robot was 1.0 m/s^2. The arm position was set as shown in Fig.11. Each parameter was shown in Table 1.

3.2 Experimental Results to Static Obstacle

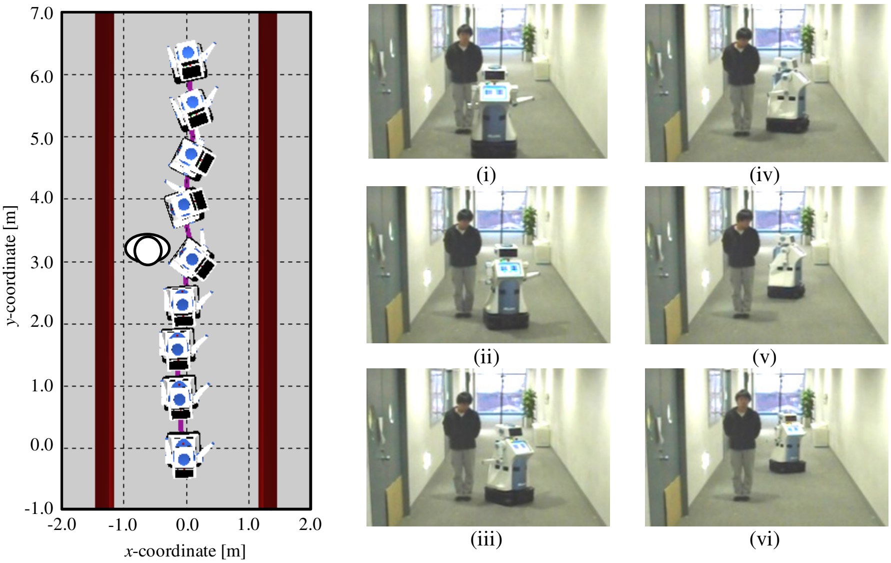
Figure 12 showed that the robot with method I can reach the goal without colliding with the obstacle. However, the position of the right arm comes close to the right-side wall. On the other hand, it was confirmed in Figures 13 and 14 that the robot with the proposed method (method II) changes the orientation angle of the robot to keep the safe distance with the right-side wall and can reach the goal point without colliding with the obstacle.



(a) Robot trajectory

(b) Scenes of the experiment

Fig. 12. Experimental result to static human using fuzzy potential method without PMF for rotational motion (method I)



(a) Robot trajectory

(b) Scenes of the experiment

Fig. 13. Experimental result to static human using fuzzy potential method with PMF for rotational motion (method II)

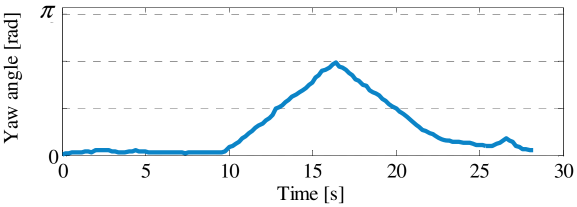


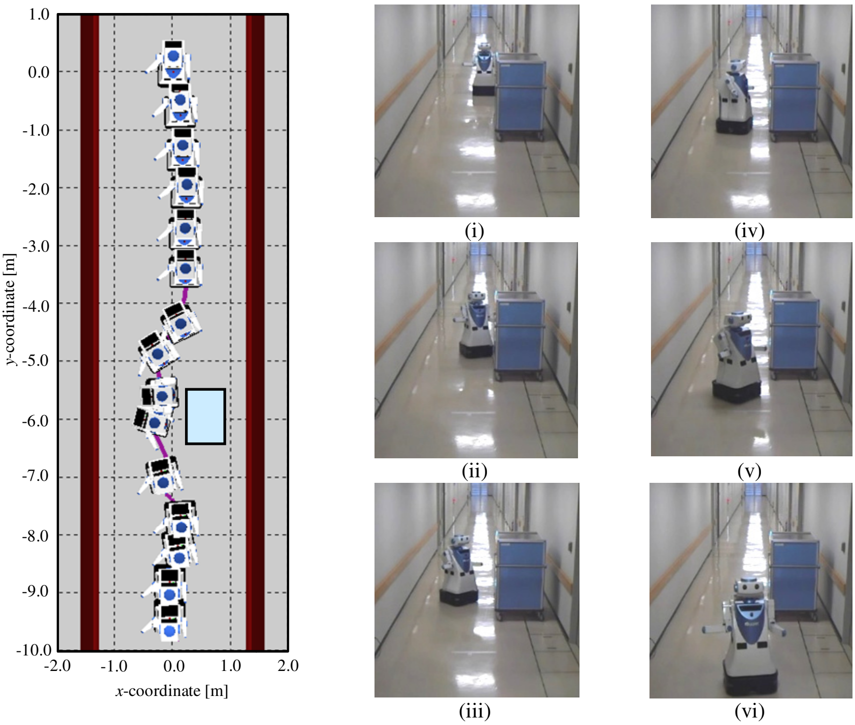
Fig. 14. Time history of yaw angle of the robot

3.3 Experimental Results to Static and Moving Obstacles in Hospital

We carried out the experiments in the hospital. The dish cart was selected as an obstacle. The width of the passage is 2.53 m. The width of the cart is 1.04 m and the depth of the cart is 0.92 m. The safety distance is 0.30 m.

Figure 15 showed the robot trajectory to the static cart using the proposed method. The robot changes the orientation angle of the robot to keep the safe distance with the right-side wall and left-side cart. The minimum distance to the right-side wall was 0.44 m and the minimum distance to the left-side cart was 0.44 m.

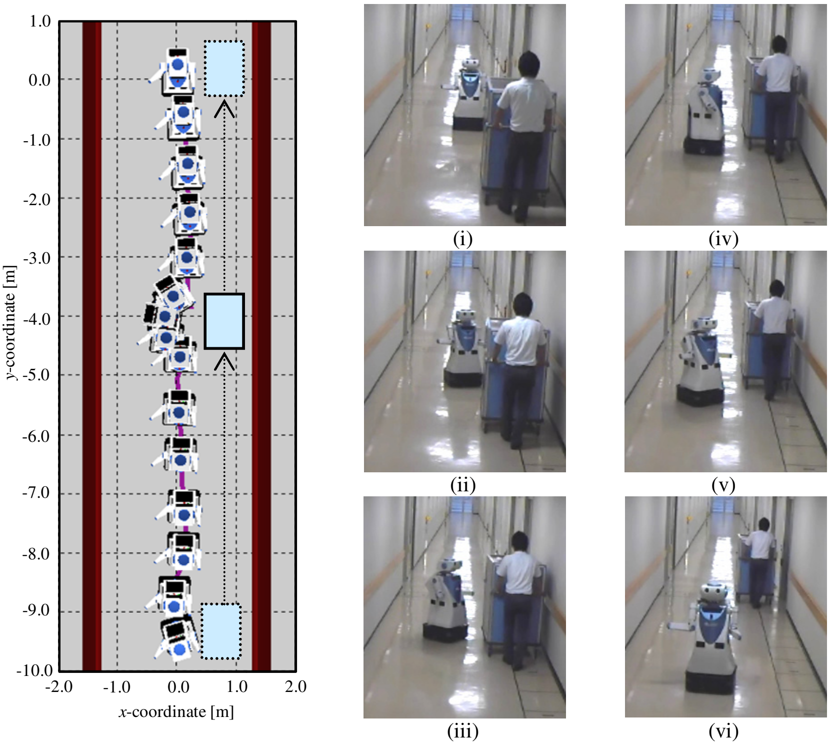
Figure 16 showed the robot trajectory to the moving cart using the proposed method. The robot keeps the safe distance with both sides by changing the orientation angle of the robot and can reach the goal point without colliding with the moving cart. However, the robot takes waste motion when the robot closes to the moving obstacle. Therefore, our proposed method should consider the velocity of the obstacle.



(a) Robot trajectory

(b) Scenes of the experiment

Fig. 15. Experimental result to static cart using fuzzy potential method with PMF for rotational motion (method II)



(a) Robot trajectory

(b) Scenes of the experiment

Fig. 16. Experimental result to moving cart using fuzzy potential method with PMF for rotational motion (method II)

4 Conclusions

In this paper, the real-time collision avoidance method with simultaneous control of both translational and rotational motion with consideration of a robot width for an autonomous mobile robot, which is horizontally long, has been proposed. This method used an omni-directional platform for the drive system and was based on the fuzzy potential method. The novel design method of potential membership function, which takes the robot's size into consideration using the capsule case, was introduced. With the proposed method, the wide robot can decide the direction of translational motion to avoid obstacles safely. In addition, by controlling rotational motion in real time, the wide robot moves while keeping a safe distance with surroundings in narrow space. The effectiveness has been verified by numerical simulations and experiments. It has been shown that the proposed method performs translational and rotational motion simultaneously according to the situation.

Acknowledgements. This work was supported in part by Grant in Aid for the Global Center of Excellence Program for "Center for Education and Research of Symbiotic, Safe and Secure System Design" from the Ministry of Education, Culture, Sport, and Technology in Japan.

References

1. Du, Z., Qu, D., Yu, F., Xu, D.: A Hybrid Approach for Mobile Robot Path Planning in Dynamic Environments. In: Proc. IEEE Int. Conf. on Robotics and Biomimetics, pp. 1058–1063 (2007)
2. Khatib, O.: Real-time Obstacle Avoidance for Manipulators and Mobile Robots. Int. J. of Robotics Research 5(1), 90–98 (1986)
3. Koren, Y., Borenstein, J.: Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1398–1404 (1991)
4. Borenstein, J., Koren, Y.: Real-Time Obstacle Avoidance for Fast Mobile Robots. IEEE Trans. on Systems, Man and Cybernetics 19(5), 1179–1187 (1989)
5. Borenstein, J., Koren, Y.: The Vector Field Histogram Fast Obstacle Avoidance for Mobile Robots. IEEE Trans. on Robotics and Automation 7(3), 278–288 (1991)
6. Lumelsky, V.J., Cheung, E.: Real Time Obstacle Collistion Avoidance in Teleoperated Whole Sensitive Robot Arm Manipulators. IEEE Trans. Systems, Man and Cybernetics 23(1), 194–203 (1993)
7. Borenstein, J., Koren, Y.: The Vector Field Histogram Fast Obstacle Avoidance for Mobile Robots. IEEE Trans. on Robotics and Automation 7(3), 278–288 (1991)
8. Dieter, F., Wolfram, B., Sebastian, T.: The Dynamic Window Approach to Collision Avoidance. IEEE Robotics and Automation 4(1), 1–23 (1997)
9. Kavraki, L.: Computation of Configuration Space Obstacles Using the Fast Fourier Transform. IEEE Trans. on Robotics and Automation 11(3), 408–413 (1995)
10. Wang, Y., Chirikjian, G.S.: A New Potential Field Method for Robot Path Planning. In: Proc. IEEE Int. Conf. on Robotics and Automation, San Francisco, CA, pp. 977–982 (2000)
11. Takahashi, M., Suzuki, T., Matsumura, T., Yorozu, A.: Obstacle Avoidance with Simultaneous Translational and Rotational Motion Control for Autonomous Mobile Robot. In: Proc. of the 8th Int. Conf. on Informatics in Control, Automation and Robotics (2011)
12. Ambrose, R.O., Savely, R.T., Goza, S.M., Strawser, P., Diftler, M.A., Spain, I., Radford, N.: Mobile manipulation using NASA's robonaut. In: Proc. IEEE ICRA, pp. 2104–2109 (2004)
13. Takahashi, M., Suzuki, T.: Multi Scale Moving Control Method for Autonomous Omnidirectional Mobile Robot. In: Proc. of the 6th Int. Conf. on Informatics in Control, Automation and Robotics (2009)
14. Tsuzaki, R., Yoshida, K.: Motion Control Based on Fuzzy Potential Method for Autonomous Mobile Robot with Omnidirectional Vision. Journal of the Robotics Society of Japan 21(6), 656–662 (2003)

An Economical Testbed for Cooperative Control and Sensing Strategies of Robotic Micro-vehicles

David S. Hermina Martinez[1], Maximillian Gonzalez[2], Xinheng Huang[1],
Benjamin Irvine[1], Chung H. Hsieh[3], Yuan R. Huang[3],
Martin B. Short[1], and Andrea L. Bertozzi[1]

[1] University of California, Los Angeles, CA 90059, U.S.A.
[2] Harvey Mudd College, Claremont, CA 91711, U.S.A.
[3] Anteros Labs, Inc, Torrance, CA 90505, U.S.A.

Abstract. Laboratory testbeds are an important part of the design of cooperative control algorithms. Sensor noise, communication delays, dropped packets, and network connectivity issues can all affect algorithm performance in different ways, and although these sources of error can be included in simulations, the degree of their effects is often not known until one attempts to implement the algorithm on real platforms. Moreover, by quantitatively assessing these constraints, one can propose new model problems of relevance to the design of these algorithms. The UCLA Applied Mathematics Laboratory is at the forefront of the development and testing of cost-efficient micro-vehicle testbeds. This work describes the third generation UCLA economical cooperative control testbed, and is an updated and expanded version of [1]. Here, we describe the setup of the testbed in detail, and demonstrate practical use of the testbed for algorithm validation by implementing cooperative steering and barrier avoidance algorithms.

Keywords: Robotics testbed, Mobile robots, Cooperative motion, Barrier avoidance, Target detection.

1 Introduction

The motivations behind the development of algorithms for multi-agent cooperative behavior have roots in a variety of disciplines. For example, unmanned aircraft reduce the risks put on human lives in hazardous environments and combat zones, and greater development of autonomous motor vehicle behavior could greatly reduce the number of traffic accidents, of which the vast majority are caused by human error. In addition, increases in mechanical autonomy have already reduced the need for many human operators in industry and commerce, and further work in this field can only accelerate their efficiency. The need to understand these and similar problems [2, 3] has therefore resulted in the construction of many laboratory testbeds [4–10].

Laboratory testing of cooperative control algorithms is important to the development of the field because it brings real-world sensor issues, communication issues, and movement issues to the forefront of the research. One of the biggest challenges for laboratory testbeds, however, is lack of adequate space. Rarely do users have access to a dedicated area large enough to allow for the testing of multi-vehicle path planning algorithms with vehicles that possess the capacity for on-board computing. Typically, such a

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 65–75.
springerlink.com

vehicle footprint would be at least the size of a laptop computer, necessitating a testbed arena on the order of 10 m across or more for meaningful experiments.

To avoid such space constraints, we have focused on the development of a micro-sized testbed that allows for testing of complex algorithms with vehicles that have a footprint smaller than a typical mobile phone. This requires the design and implementation of custom vehicles as well as careful thought into how algorithms are developed and managed. In prior work [11, 12], we developed a robust testbed based on modified toy cars that was successful in testing cooperative control algorithms, but also suffered from some major disadvantages. For example, the first generation testbed [11] required a centralized computer to perform all algorithm processing and send commands to the micro-cars, which lacked any form of on-board computing or sensors. The second generation [12] added IR range sensors and very modest on-board computing to the microcars, but they were still reliant on a processing computer for any advanced algorithms. This paper describes a third generation vehicle that is completely customized with vastly increased computational power, allowing not only for on-board algorithm processing but also for realtime user interaction via a remote terminal, as well as an array of sensors and enhanced communication capabilities. All this is done while still maintaining the compact footprint of prior work, avoiding space constraint issues.

The remainder of the paper is organized as follows: Section 2 describes the setup of the testbed and the new generation micro-car hardware and software, Section 3 presents some of the experiments that have been implemented on the testbed, and Section 4 outlines future testbed goals and conclusions.

2 Testbed Setup and Hardware

The UCLA Applied Mathematics Laboratory Testbed (AMLT) is divided into three subsystems: an overhead-camera and PC tracking system, a remote terminal/programming PC, and the micro-car robotic vehicles (see Fig. 1).

Physically, the testbed is a 1.5 m x 2.0 m rectangular area in which the micro-cars operate. The area itself is made of black asphalt felt paper with a white boundary, providing a uniform, non-reflective background for imaging purposes. The cars' positions are tracked by 2 overhead Imaging Source DMK 21F04 1/4 Monochrome CCD cameras with a resolution of 640 x 480 pixels. They have a frame rate of 30 fps and are connected to an image processing PC via firewire cable. The tracking software uses Intels OpenCV library and Microsofts DirectShow Software Development Kit for data transmission from the cameras to the tracker PC, frame processing, and data packaging to the Wi.232 transceiver. The data required for the tracker to calculate a vehicles position, heading angle, and ID is obtained by processing each vehicles ID tag (see Fig. 2). When a frame is captured, an OpenCV contour function is used to find the vehicle ID tags. This function detects the region of highest contrast at each frame, a Car Object , and creates a boundary around it. The same task is repeated within the bounded Car Object to find the Heading and ID symbols, to calculate all the information needed. These data are then packaged in sequential vehicle ID order and broadcasted to the vehicles through the Wi.232DTS transceiver, for use in control algorithms. This mimics the functionality of a GPS unit that may be present in more advanced vehicles in the field.

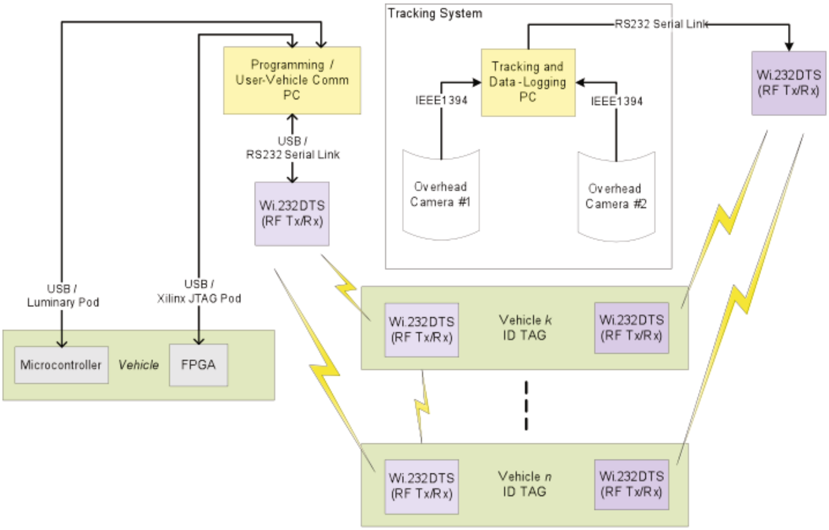


Fig. 1. An overview of the AMLT testbed's three subsystems, and the communications between them

A real-time tracking interface, written in ANSI C, displays all the spatial information and shows live camera feed with the contrast boundaries of each vehicle. Additionally, all transmitted data can be logged by ID in a tab delimited text file for post-processing. The tracking system is capable of managing up to 13 vehicles [11, 12].

The cars are given commands and can relay status information by communicating via a separate serial radio with an interface PC that serves primarily as a remote terminal for the cars. The ability to broadcast messages to the interface PC for display proves to be a very useful debugging tool.

2.1 Vehicle Hardware

The third generation micro-cars (model ALMC-100, see Fig. 3) are purpose built from the ground up, in contrast to previous generation vehicles that were modified off-the-shelf toy cars. The ALMC-100 is designed to mimic many of the features one would expect to find in a full sized autonomous vehicle, in a compact package. The vehicle measures approximately 8 cm (wheelbase) x 5 cm (width); the height varies from 5.8 cm to 8 cm depending on configuration. The ALMC-100 is a rear wheel drive, front steering car with a maximum speed of 20 cm/s and maximum turning angle of ±18°. Power comes from four AAA batteries with approximately 3.8 W, yielding a run time of greater than 30 minutes.

The ALMC-100 features two processing units on individual printed circuit boards, which are stacked atop each other. The lower "chassis" board is the base of the vehicle where the drive train is bolted in addition to the electronics. The chassis board contains a 50MHz ARM Cortex-M3 processor with 8KB SRAM and 64KB flash memory. A 1KB EEPROM is also included to store unique, non-volatile information such as vehicle identification number and motor control gains and offsets. The chassis board

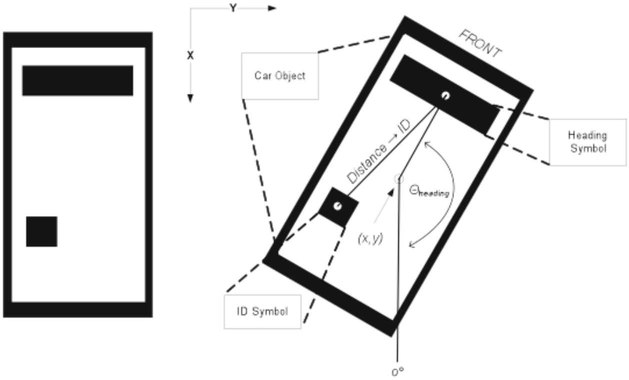


Fig. 2. The tracking system relies on vehicle-specific barcodes to determine position and heading of the vehicles

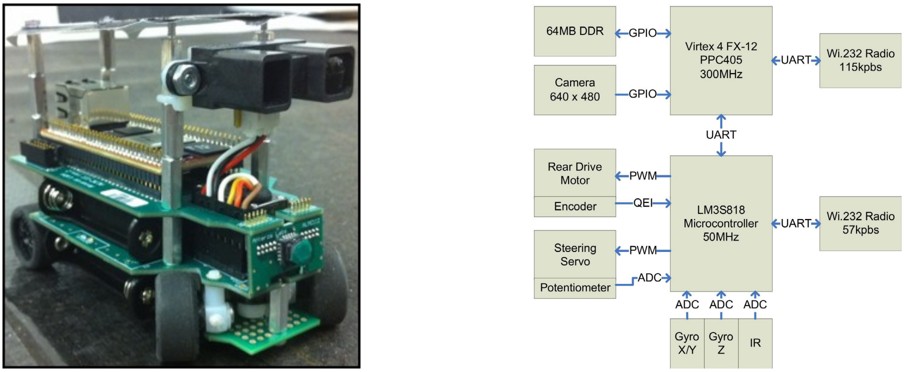


Fig. 3. (Left) A photograph of the ALMC-100 micro-car. (Right) A block diagram of the ALMC-100 micro-car hardware

also houses two gyroscopes for 3-axis measurements, a $0.45°$ optical encoder used for velocity estimation, and is attached to either a long-range or short-range IR module; the long range sensors can detect objects in the 10 cm - 140 cm range, while the short range sensors can only detect from 10 cm - 80 cm.

The upper "processing" board contains an off-the-shelf Xilinx Virtex-4 FX12 FPGA Mini-Module. Currently, the FPGA is configured to embed a 300MHz PowerPC405 processor plus a number of peripheral interfaces. The interfaces allow the PPC405 to access 64MB of DDR SDRAM and 4MB of flash memory among other peripherals.

The wireless communication system consists of two Wi.232 radio modules, one on each board, capable of transmitting and receiving at 11520 bps. The wireless module on the chassis board is configured to 57600 bps and receives only information from the overhead tracking system, mimicking GPS. The wireless module on the processing board is configured to 115200 bps and is intended for inter-vehicle communication and for access of the vehicle via the remote terminal. The two radios operate on different frequencies to avoid interference.

The driving factor behind the use of two processing units is to segregate motion control and path planning. The motion control is accomplished on the chassis board, which maintains its control loop at 1000 Hz while sampling the various sensors at 500 Hz. The chassis processor extracts the vehicle's own position from the overhead tracking system's broadcast sent at 30 Hz. The vehicle's position and other vehicle and sensor states are relayed to the processing board also at 30 Hz over the universal asynchronous receiver/transmitter (UART) connecting the two boards. Thanks to the powerful processing available to the upper board, the cars can perform all required path planning themselves; in previous versions of the AMLT, vehicles relied on a desktop computer to perform all such calculations and relay instructions to the cars.

2.2 Vehicle Software

Each vehicle runs two separate controlling programs. The chassis board controller is based on FreeRTOS [13], a realtime operating system that can run multiple tasks at up to 1000 Hz, and is designed to provide two main functionalities: control the vehicle's motion by providing PWM signals for the drive motor and the servo, and supply sensory data and system information for the processing board. At the center of the chassis board controller is a path-generation task that calculates target velocities at 100 Hz and generates PWM signals at 50 Hz that feed the servo. Based on the target velocities, a velocity-control task generates a 1000Hz PWM signal to control the drive motor. The controller is also designed to recognize a set of serial commands, thus allowing the processing board to control high-level motion and access data.

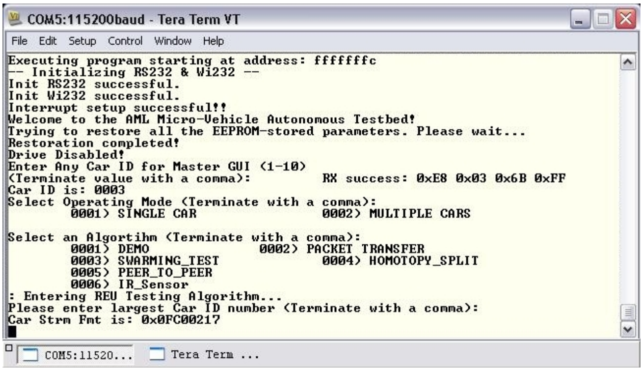
The processing board boasts much deeper memory space and faster processing speed, and can thus support more complex programs that can feature user-friendly interfaces and large data-sets. Currently, this controller serves as a remote terminal interface [14], an example of which is shown in Fig. 4. In order to interface with the user, the micro-car transmits messages to the interface PC, via its upper serial radio. If many cars are in use at once, only one car, chosen by the user, broadcasts the terminal messages. The user interacts with the micro-cars through the interface PC's keyboard. These inputs are transmitted to the micro-cars, which then execute the chosen algorithms. The interface is composed of two modes: a single car mode for demonstrating basic operation of the vehicles, and a multiple car mode to implement cooperative algorithms.

3 Testbed Experiments

The motion of the cars is modeled based on a Frenet-Serret framework as in the second-generation testbed [15]. Each car k has its own coordinate frame relative to its heading, with $\mathbf{x}_k$ being the unit vector oriented in the direction the car's motion and $\mathbf{y}_k$ being the unit vector oriented perpendicular to $\mathbf{x}_k$ (Fig. 5). The car's motion can then be described with the following equations:

$$\dot{\mathbf{z}}_k = \mathbf{x}_k, \qquad \dot{\mathbf{x}}_k = v u_k \mathbf{y}_k, \qquad \dot{\mathbf{y}}_k = -v u_k \mathbf{x}_k \ , \tag{1}$$

where $\mathbf{z}_k(s)$ is the arclength parametrized path of the car with respect to a fixed coordinate frame, the scalar u_k is the curvature of the path at a specific point, and v is



```
Executing program starting at address: ffffffc
-- Initializing RS232 & Wi232 --
Init RS232 successful.
Init Wi232 successful.
Interrupt setup successful!!
Welcome to the AML Micro-Vehicle Autonomous Testbed!
Trying to restore all the EEPROM-stored parameters. Please wait...
Restoration completed!
Drive Disabled!
Enter Any Car ID for Master GUI (1-10)
(Terminate value with a comma):         RX success: 0xE8 0x03 0x6B 0xFF
Car ID is: 0003
Select Operating Mode (Terminate with a comma):
        0001) SINGLE CAR                0002) MULTIPLE CARS

Select an Algortihm (Terminate with a comma):
        0001) DEMO              0002) PACKET TRANSFER
        0003) SWARMING_TEST             0004) HOMOTOPY_SPLIT
        0005) PEER_TO_PEER
        0006) IR_Sensor
: Entering REU Testing Algorithm...
Please enter largest Car ID number (Terminate with a comma):
Car Strm Fmt is: 0x0FC00217
```

Fig. 4. A typical screenshot of the remote terminal on the interface PC, showing several menus in its structure. Near the bottom, the multiple car mode menu is shown with several of the algorithm selections visible.

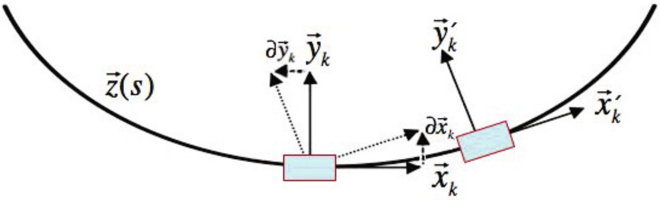


Fig. 5. Diagram of the coordinate frame of a micro-car moving along a parametrized path $\mathbf{z}(s)$.

the (typically constant) speed of the vehicle. Thus, the path of each car can be determined simply by specifying its curvature over time. To convert from curvature u_k to the desired turning angle ϕ_k of the vehicle's wheels, we use the equation

$$\phi_k = \tan^{-1}(u_k L_{\text{car}}) \ , \tag{2}$$

where L_{car} is the car length of 8 cm. If $|\phi_k| > 18°$, the maximum turning angle that the servos can turn, $|\phi_k|$ is rounded down to the maximum. This limits the minimum turning diameter to approximately 50 cm, a notable constraint for the testbed, which is 1.5 m x 2.0 m.

3.1 Cooperative Motion Algorithms

From Morgan & Schwartz's model for swarming [15], the curvature for each car k is calculated as

$$u_k = \sum_{j \neq k} u_{jk}, \tag{3}$$

where j cycles through the indices of all the other cars on the testbed. For simple swarming, the following equation for u_{jk} is used:

$$u_{jk} = -\eta \left(\frac{\mathbf{r}_{jk}}{\|\mathbf{r}_{jk}\|} \cdot \mathbf{x}_k \right) \left(\frac{\mathbf{r}_{jk}}{\|\mathbf{r}_{jk}\|} \cdot \mathbf{y}_k \right) - \alpha \left[1 - \left(\frac{r_0}{\|\mathbf{r}_{jk}\|} \right)^2 \right] \left(\frac{\mathbf{r}_{jk}}{\|\mathbf{r}_{jk}\|} \cdot \mathbf{y}_k \right) + \mu \mathbf{x}_j \cdot \mathbf{y}_k, \quad (4)$$

where $\mathbf{r}_{jk}$ is the vector from car j to car k, r_0 is the desired distance between cars for the swarm, and α, η, and μ are weighting parameters for three separate aspects of the desired motion. The term with coefficient η works to turn each car perpendicular to $\mathbf{r}_{jk}$; the term with coefficient α turns the cars toward each other if they are further than r_0 apart, and turns them away from each other if they are closer than r_0; and the term with coefficient μ orients the cars toward a common heading. By varying the three weights of these terms and introducing slight modifications, different cooperative motion can be achieved, such as the circle-tracking, leader following, and homotopy control laws described in [12].

The summation in (3) is typically over all cars, leading to global coupling. However, such coupling should rarely be expected in real-world scenarios, where each vehicle may only know its own position and perhaps the positions of a few other nearby vehicles. Therefore, in addition to global coupling, we have also tested a form of daisy-chain coupling whereby each car k is only coupled to the two cars with indices $j = k \pm 1$, as illustrated in Fig. 6. The implementation of this daisy-chain algorithm on larger swarms has yielded promising results on the testbed. Although each car is only aware of two others, the swarm operates as a whole because of the iterative coupling utilized by the daisy-chain system. The performance of the swarm is somewhat dependent on initial placement of the micro-cars, however, a behavior noted in similar systems [16]. If the cars are ordered by ID, the daisy chain swarm performs excellently; the cars rarely collide and find a common heading quickly. However, if they are placed in a random order, collisions may occur as the cars are not necessarily aware of their closest neighbors, which may no longer be the cars they are coupled to. This especially occurs if the cars are initially placed in close proximity to each other, typically less than approximately two car lengths. However, if groups of cars are initially separated by distances larger than this, they are usually able to regroup and find a common heading with very few collisions, as shown in Fig. 7.

In addition to the fixed daisy-chain based on car index numbers described above, we have also implemented an algorithm that creates a closed daisy-chain upon algorithm startup, based on initial vehicle locations. For each car k, the algorithm determines the two cars “connected” to k in the following way:

- Partition all cars into two groups – those to the left of k (subset $\mathbb{L}_k$) and those to the right of k (subset $\mathbb{R}_k$), based upon the position and orientation of k and the positions of all other cars.
- If $\mathbb{R}_k$ is not empty, partner one is the physically closest member of $\mathbb{R}_k$. Otherwise, partner one is the physically furthest member of $\mathbb{L}_k$.
- If $\mathbb{L}_k$ is not empty, partner two is the physically closest member of $\mathbb{L}_k$. Otherwise, partner two is the physically furthest member of $\mathbb{R}_k$.

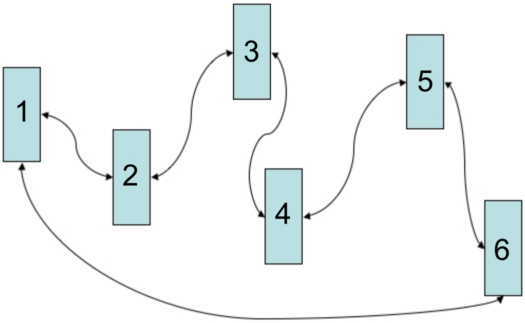


Fig. 6. Diagram of daisy-chain coupling with the cars iteratively connected. End conditions were used to create a closed loop.

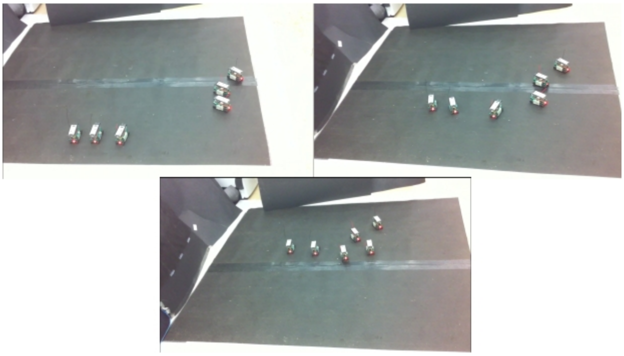


Fig. 7. Three frames of an experimental run using daisy-chain coupling. The cars are originally separated in two groups (top left). During the run, the cars regroup (top right) and find a common orientation before exiting the testbed (bottom).

This algorithm allows the cars to create a daisy-chain that operates well under a broader range of initial conditions, since partners are chosen at startup in such a way as to generally minimize the initial distances between partners while maintaining the closed loop that allows the chain to function as a whole.

3.2 Barrier Avoidance and Target Seeking

A control law similar to (4) can be used to avoid barriers and seek a target. This is accomplished with the equation

$$u_{jk} = \gamma \left[1 - \left(\frac{r_0}{\|\mathbf{r}_{tk}\|}\right)^2\right] \left(\frac{\mathbf{r}_{tk}}{\|\mathbf{r}_{tk}\|} \cdot \mathbf{y}_k\right) - \beta \,\text{sign}\left[\sum_b C\left(\mathbf{r}_{bk}, 0, \omega\right) \left(\frac{\mathbf{r}_{bk}}{\|\mathbf{r}_{bk}\|} \cdot \mathbf{y}_k\right)\right] \times \left[\sum_b C\left(\mathbf{r}_{bk}, 0, \omega\right) \left(\frac{\mathbf{r}_{bk}}{\|\mathbf{r}_{bk}\|} \cdot \mathbf{x}_k\right)\right], \tag{5}$$

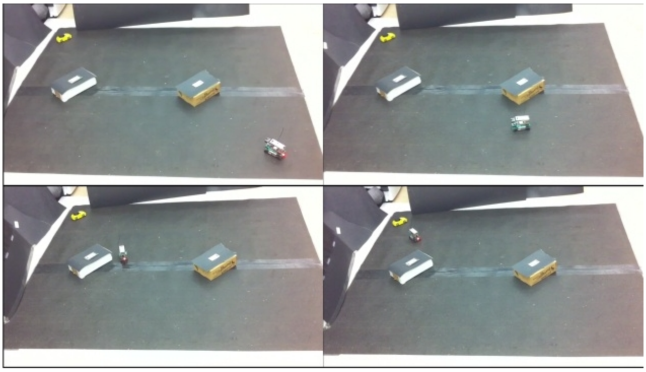


Fig. 8. Four frames of an experimental run of tracking camera-assisted barrier avoidance, with the yellow object in the top left of each frame as the target. The frames are ordered from left to right, top to bottom. The car detects the first barrier in the second frame and begins to avoid it. It continues toward the target and avoids the second barrier in frame three and reaches the target in frame four.

where

$$C(\mathbf{r}, q, \omega) = \begin{cases} 1 \; \|\mathbf{r}\| < \omega \\ q \text{ otherwise} \end{cases}, \tag{6}$$

$\mathbf{r}_{tk}$ is the vector from car k to the target, $\mathbf{r}_{bk}$ is the vector from car k to the barrier b, and γ and β are the weights assigned to the target seeking and barrier avoidance behaviors, respectively.

Equation (5) is dependent on the car's knowledge of the position of the target and of all barriers within a certain distance ω of itself. This has been implemented on our testbed in two ways. The first uses a hard-coded target position and makes use of hats sitting on the barriers. The tracking cameras pick up the barrier locations and relay the information to the target-seeking car, which then weaves its way through the obstacles to its goal. Several photos of such an experiment are shown in Fig. 8.

Ideally, though, the cars would not rely on the tracking computer to provide them with information on the location of barriers, but would instead use their IR sensors to detect barriers on the fly and adjust their movement accordingly; our second implementation of barrier avoidance does just this. It again uses a hard-coded target position, but now uses the IR sensor readings of the cars to estimate values for $\mathbf{r}_{bk}$. Since the sensors only work over a limited range, this naturally incorporates a term such as $C(\mathbf{r}_{bk}, 0, \omega)$ into the behavior. One disadvantage of this method is that only barriers nearly directly ahead of the car can be sensed, so that cars may occasionally turn into nearby barriers that lie at their sides. We find, though, that the relatively slow motion and large turning radius of the cars minimizes these issues, and that swarms of cars employing this algorithm, in conjunction with (4), typically navigate the barriers successfully, as show in Fig. 9.

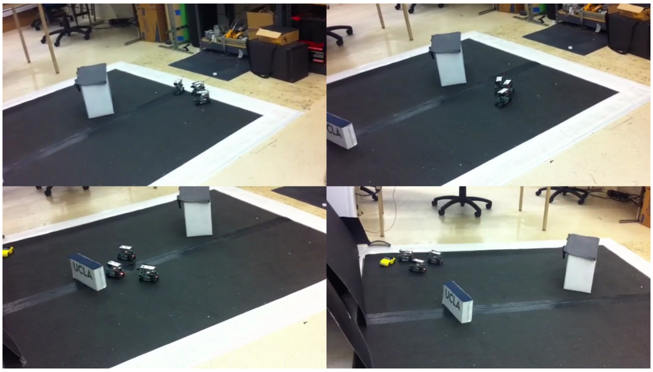


Fig. 9. Four frames of an experimental run of IR sensor barrier avoidance, with the yellow object near the top left of frames three and four as the target. The frames are ordered from left to right, top to bottom. The cars detect the first barrier in the second frame and avoid it. They continue toward the target and avoid the second barrier in frame three and reach the target in frame four.

4 Future Work and Conclusions

Though the third generation of the AMLT has already surpassed previous versions in terms of successfully performed experiments, there are still many aspects of the new hardware that have not yet been utilized. Foremost amongst these is the potential for a true peer-to-peer communication network for the vehicles. Creating such a network would allow for the implementation of many advanced algorithms. For example, it would allow for the vehicles in our daisy-chain setup to determine their nearest neighbors in realtime, decreasing the chances of collisions occurring.

Another avenue of exploration is greater usage of the many on-board sensors. The optical encoder and gyroscopes could allow each car to estimate its own position over time with less reliance on the tracking computer, assuming the car's initial coordinates are known. This would mimic the intermittent GPS outages expected in field vehicles, and give methods for dealing with them. Also, the on-board camera could perhaps be used in conjunction with the IR sensor for enhanced barrier detection, or possibly target detection when the location of the target is unknown.

Acknowledgements. This paper is supported by ARO MURI grant 50363-MA-MUR and NSF grants DMS-0914856, DMS-0907931, DMS-0601395, and EFRI-1024765.

References

1. Gonzalez, M., Huang, X., Martinez, D.S.H., Hsieh, C.H., Huang, Y.R., Irvine, B., Short, M.B., Bertozzi, A.L.: A third generation micro-vehicle testbed for cooperative control and sensing strategies. In: Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics (2011)

2. Michael, N., Fink, J., Kumar, V.: Cooperative manipulation and transportation with aerial robots. Autonomous Robots 30, 73–86 (2011)
3. Bhattacharya, S., Michael, N., Kumar, V.: Distributed coverage and exploration in unknown non-convex environments. In: 10th International Symposium on Distributed Autonomous Robots. Springer (2010)
4. Jin, Z., Waydo, S., Wildanger, E.B., Lammers, M., Scholze, H., Foley, P., Held, D., Murray, R.M.: MVWT-II: the second generation Caltech Multi-Vehicle Wireless Testbed. In: Proceedings of the American Control Conference, pp. 5321–5326 (2004)
5. Turgut, A.E., Gokçe, F., Celikkanat, H., Bayındır, L., Sahin, E.: Kobot: A mobile robot designed specifically for swarm robotics research. Technical report, Department of Computer Engineering, Middle East Technical University, Ankara, Turkey (2007)
6. Cruz, D., McClintock, J., Perteet, B., Orqueda, O.A.A., Cao, Y., Fierro, R.: Decentralized cooperative control - a multivehicle platform for research in networked embedded systems. IEEE Control Systems Magazine 27, 58–78 (2007)
7. Punpaisarn, S., Sujitjorn, S.: SUT-CARG car-like robots: their electronics and control architecture. WSEAS Transactions on Circuits and Systems 7, 579–589 (2008)
8. McLain, T., Beard, R.W.: Unmanned air vehicle testbed for cooperative control experiments. In: Proceedings of the American Control Conference, pp. 5327–5331 (2004)
9. Azarnasab, E., Hu, X.: An integrated multi-robot test bed to support incremental simulation-based design. In: Proceedings of the IEEE International Conference on System of Systems Engineering (2007)
10. Sibley, G., Rahimi, M., Sukhatme, G.: Robomote: A tiny mobile robot platform for large-scale sensor networks. In: Proceedings of the 2002 IEEE International Conference on Robotics and Automation (2002)
11. Hsieh, C.H., Chuang, Y.L., Huang, Y., Leung, K.K., Bertozzi, A.L., Frazzoli, E.: An economical micro-car testbed for validation of cooperative control strategies. In: Proceedings of the American Control Conference, pp. 1446–1451 (2006)
12. Leung, K.K., Hsieh, C.H., Huang, Y.R., Joshi, A., Voroninski, V., Bertozzi, A.L.: A second generation micro-vehicle testbed for cooperative control and sensing strategies. In: Proceedings of the American Control Conference, pp. 1900–1907 (2007)
13. Real Time Engineers Ltd.: FreeRTOS-A Free RTOS for ARM7, ARM9, Cortex-M3, MSP430, MicroBlaze, AVR, x86, PIC32, PIC24, dsPIC, H8S, HCS12 and 8051, http://www.freertos.org
14. Hermina Martinez, D.S.: Integration of 3rd generation vehicles to the applied mathematics laboratory autonomous micro-vehicle testbed. Master's thesis, Department of Electrical Engineering, University of California, Los Angeles (2010)
15. Morgan, D.S., Schwartz, I.B.: Dynamic coordinated control laws in multiple agent models. Physics Letters A 340, 121–131 (2005)
16. Marshall, J.A., Fung, T., Broucke, M.E., D'Eleuterio, G.M.T., Francis, B.A.: Experimental validation of multi-vehicle coordination strategies. In: Proceedings of the American Control Conference, pp. 1091–1095 (2005)

Mobile Manipulators Motion Planning Based on Trajectory Tracking Control

Razvan Solea and Daniela Cristina Cernega

Department of Automation and Electrical Engineering,
“Dunarea de Jos“ University of Galati
Domneasca Street, No.47, 800008-Galati, Romania
{razvan.solea,daniela.cernega}@ugal.ro

Abstract. In order for collaborative manipulators systems to perform their tasks, they have to move an object together. The control purpose for such coordinated systems is to ensure the movement of the mobile platforms and manipulators from an initial position to a desired position. The approach presented in this paper focuses on solving the motion planning problem for only one mobile platform equipped with a manipulator. In order to ensure the smooth movement of the considered system the nonlinear sliding mode control was used to solve the motion planning problem. The paper presents the controller design for the trajectory-tracking problem using the sliding mode control for a mobile platform equipped with a manipulator.

Keywords: Mobile manipulators, Nonlinear control, Kinematics, Trajectory tracking.

1 Introduction

Mobile Manipulator systems are typically composed of a mobile base platform with one (or more) mounted manipulators. Taking advantage of the increased mobility and workspace provided by the mobile base, such systems have found applications in industry and in research principally due to their engineering simplicity (easy to build and to control than legged robots) and their low specific resistance (high energy efficiency). Moving mobile manipulators systems, present many unique problems that are due to the coupling of holonomic manipulators with nonholonomic bases.

Any system combining a wheeled mobile platform and one or several manipulators (classically arms) is named *wheeled mobile manipulators* (WMM) - like in Fig. 1. From the set of constraints and characteristics, different approaches have been developed to control WMM. A first class of approach is inherited from the control schemes that have been developed for manipulators. Those control schemes have been extended to WMM in order to account for their specificities. Among those approaches, the pioneering work of H. Seraji [1] can be distinguished. He proposed an extension of kinematic based control laws to the case of a mobile manipulator equipped with a wheeled platform (unicycle) and a manipulator.

A variety of theoretical and applied control problems have been studied for various classes of nonholonomic control systems. Motion planning problems are concerned

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 77–88.
springerlink.com

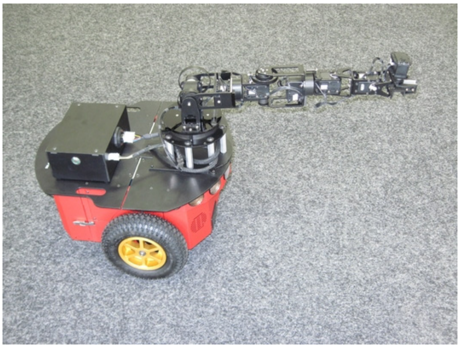


Fig. 1. Pioneer 3DX mobile robot with manipulator

with obtaining open loop controls that steer the system from an initial state to a final state without violating the nonholonomic constraints.

The determination of the actuator rates for a given end effector motion of a redundant manipulator is typically an under-constrained problem. A number of schemes have been proposed in the literature for the resolution of the redundancy. The principal underlying theme is one of optimizing some measure of performance based on kinematics of the system and in some cases extended to include the dynamics. However, in this paper, we focus our attention purely on kinematic redundancy resolution schemes.

The control algorithms and strategies have been categorized into three groups, namely continuous time-variant, discontinuous and hybrid control strategies [2], [3], [4], [5], [6], [7], [8]. Output tracking laws are easier to design and implement, and can be embedded in a sensorbased control architecture when the task is not fully known in advance. For this reason, with the exception of [9] that takes a somehow intermediate approach, most works on WMMs focus on kinematic control, e.g., [10], [11], [12].

The rest of the paper is organized as follows: Section 2 develops the notation and the kinematic model for the WMM under consideration. Section 3 focuses on creation of a kinematic control law based on sliding-mode strategy. Section 4 presents simulation results to show the effectiveness of the trajectory-traking control scheme. Section 5 concludes the paper with a brief discussion and summarizes the avenues for future work.

2 Kinematic Model

In this section, we present the notation and the kinematic model of the system under consideration. Referring to Figure 2, the WMM under consideration consists of a differentially driven WMR base with a mounted planar two-link manipulator (is considered for simplity). The wheels are located at a distance of b from the center of the wheel axle. The wheel has a radius of r. The base of the manipulator is located at a distance of a D from the center of the wheel axle. The length of the first and second links are L_1 and L_2 respectively.

Motion planning has been treated mostly as a kinematic problem where the dynamics of the system have been generally neglected. However, with nonholonomic systems,

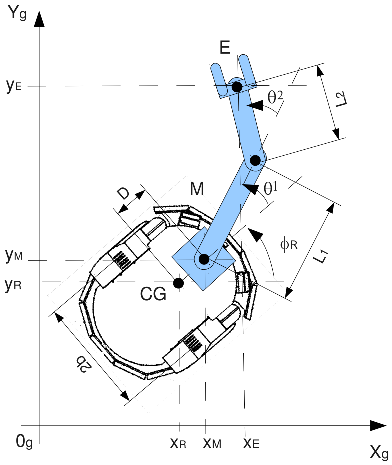


Fig. 2. Schematic of the WMM

ignoring the dynamics reduces the significance of the results to low speeds although it is well documented that avoidance of obstacles, parking maneuverability, and more motion control is feasible at higher speeds as well.

The configuration of a WMM can be completely described by the following generalized coordinates:

$$q^T = [x_R, y_R, \phi_R, \theta_1, \theta_2] \tag{1}$$

where $[x_R, y_R, \phi_R]$ describes the configuration of the WMR and $[\theta_1, \theta_2]$ describes the configuration of the planar manipulator. (x_R, y_R) is the Cartesian position of the center of the axle of the WMR, ϕ_R is the orientation of the WMR, and θ_2, θ_2 are the relative angles that parameterize the first and second link of the mounted manipulator. The kinematics of the differentially-driven WMR can be represented by its equivalent unicycle model, and described as:

$$\begin{aligned} \dot{x}_R &= v_R \cdot cos(\phi_R) \\ \dot{y}_R &= v_R \cdot sin(\phi_R) \\ \dot{\phi}_R &= \omega_R \end{aligned} \tag{2}$$

where v_R and ω_R are the forward and angular velocities inputs.

The position and orientation of the end-effector in the world frame can be derived from homogeneous transform according to the position and orientation of the mobile robot in the world frame, that of the end-effector in the manipulators base frame, and the transform between the mobile robot frame and the manipulators base frame. The kinematics of the mobile manipulator can be described like:

$$\begin{aligned} x_E &= x_M + L_1 \cdot cos(\phi_R + \theta_1) + L_2 \cdot cos(\phi_R + \theta_1 + \theta_2) \\ y_E &= y_M + L_1 \cdot sin(\phi_R + \theta_1) + L_2 \cdot sin(\phi_R + \theta_1 + \theta_2) \end{aligned} \tag{3}$$

where (x_M, y_M) is the position of mounting point M of the mobile platform and ϕ_R is the platform orientation. Eqs.3 show that the position of the end-effector E depends on the position and the orientation of the mobile platform. This illustrates the fact that mobile manipulators, in contrast to fixed ones, can have an infinite workspace.

$$\begin{aligned} x_M &= x_R + D \cdot cos(\phi_R) \\ y_M &= y_R + D \cdot sin(\phi_R) \end{aligned} \tag{4}$$

By differentiating eqs. (3) and (4) will get:

$$\begin{aligned} &\dot{x}_E = v_R \cdot cos(\phi_R) - D \cdot \omega_R \cdot sin(\phi_R) - L_1 \cdot (\omega_R + \omega_1) \cdot sin(\phi_R + \theta_1) - \\ &-L_2 \cdot (\omega_R + \omega_1 + \omega_2) \cdot sin(\phi_R + \theta_1 + \theta_2) \\ &\dot{y}_E = v_R \cdot sin(\phi_R) + D \cdot \omega_R \cdot cos(\phi_R) + L_1 \cdot (\omega_R + \omega_1) \cdot cos(\phi_R + \theta_1) - \\ &-L_2 \cdot (\omega_R + \omega_1 + \omega_2) \cdot cos(\phi_R + \theta_1 + \theta_2) \end{aligned} \tag{5}$$

If the next inequality is not satisfied, then the target is outside the manipulator reach and thus the mobile platform must move in order to bring the target into the manipulator's workspace.

$$|cos(\phi_R)| \leqslant 1 \Rightarrow (x_E - x_M)^2 + (y_E - y_M)^2 \leqslant (L_1 + L_2)^2 \tag{6}$$

The first constraint accounts for the non-holonomic behavior of the wheels. It constrains the velocity of the WMR to be along the rolling direction of the wheels only. Velocity perpendicular to the rolling direction must be zero as follows :

$$\dot{x}_R \cdot sin(\phi_R) - \dot{y}_R \cdot cos(\phi_R) = 0 \tag{7}$$

This constraint, written for the manipulator attachment point M, becomes:

$$\dot{x}_R \cdot sin(\phi_R) - \dot{y}_R \cdot cos(\phi_R) + \dot{\phi}_R \cdot D = 0 \tag{8}$$

3 Sliding-Mode Controller Design

A WMM system is especially useful when the manipulator task is outside the manipulator reach. Therefore, in this section we assume that this is always the case, in other words that inequality (6) is not satisfied for a given target.

In this chapter is developed a control routine for the mobile robot that allows for independent control of both the task-space (end-effector) and the configuration-space (mobile base). As mentioned before the primary task is controlling the position of the end-effector and attached payload. The trajectory of the mobile base consists of a time varying function of the position of the end-effector. Once the end-effector final position is known, there exists extra degrees-of-freedom that need to be controlled. These consist of the posture of the mobile robot base and arms. This is depicted in Figure 3, where a mobile robot is shown moving from initial position to the final position.

Assumption: The prescribed final posture (position, $[x_E, y_E]$ and orientation $[\phi_R, \theta_1, \theta_2]$) is required.

Although the final position is reachable, it is virtually impossible to harvest exact orientations via continuous feedback controllers at the equilibrium point of nonholonomic

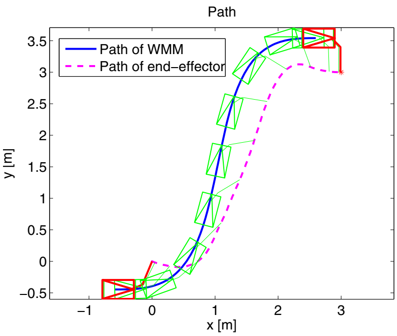


Fig. 3. Maneuver example using WMM

systems, a direct result of Brocketts Theorem [13]. Notwithstanding the limitation, we adopt the sliding-mode technique from [14] to maneuver the WMM into a final position such that the prescribed final orientation could also be accomplished.

A trajectory planner for wheeled mobile manipulators must generate smooth velocity profiles (linear and angular) with low associated accelerations. The trajectory planning process can be divided into two separate parts. First, a continuous collision-free path is generated. In a second step, called trajectory generation, a velocity profile along the path is determined. A method to generate a velocity profile, respecting human body comfort, for any two-dimensional path in static environments was proposed in [15].

Uncertainties which exist in real mobile robot applications degrade the control performance significantly, and accordingly, need to be compensated. In this section, is proposed a sliding-mode trajectory-tracking controller, in Cartesian space, where trajectory-tracking is achieved even in the presence of large initial pose errors and disturbances. The application of SMC strategies in nonlinear systems has received considerable attention in recent years. A well-studied example of a non-holonomic system is a WMM that is subject to the rolling without slipping constraint. In trajectory tracking is an objective to control the nonholonomic WMM to follow a desired trajectory, with a given orientation relatively to the path tangent, even when disturbances exist.

Let us define the sliding surface $S = [s_1, s_2, s_3, s_4]^T$ as:

$$\begin{aligned} s_1 &= \dot{x}_e + \gamma_x \cdot x_e \\ s_2 &= \dot{y}_e + \gamma_y \cdot y_e + \gamma_0 \cdot sgn(y_e) \cdot \phi_e \\ s_3 &= \dot{\theta}_{e1} + \gamma_{\theta_1} \cdot \theta_{e1} \\ s_4 &= \dot{\theta}_{e2} + \gamma_{\theta_2} \cdot \theta_{e2} \end{aligned} \tag{9}$$

where γ_0, γ_x, γ_y, γ_{θ_1} and γ_{θ_2} are positive constant parameters, x_e, y_e ϕ_e, θ_{e1} and θ_{e2} are the trajectory-tracking errors defined in Fig. 4:

$$\begin{aligned} x_e &= (x_R - x_d) \cdot cos(\phi_d) + (y_R - y_d) \cdot sin(\phi_d) \\ y_e &= -(x_R - x_d) \cdot sin(\phi_d) + (y_R - y_d) \cdot cos(\phi_d) \\ \phi_e &= \phi_R - \phi_d \\ \theta_{e1} &= \theta_1 - \theta_{d1} \\ \theta_{e2} &= \theta_2 - \theta_{d2} \end{aligned} \tag{10}$$

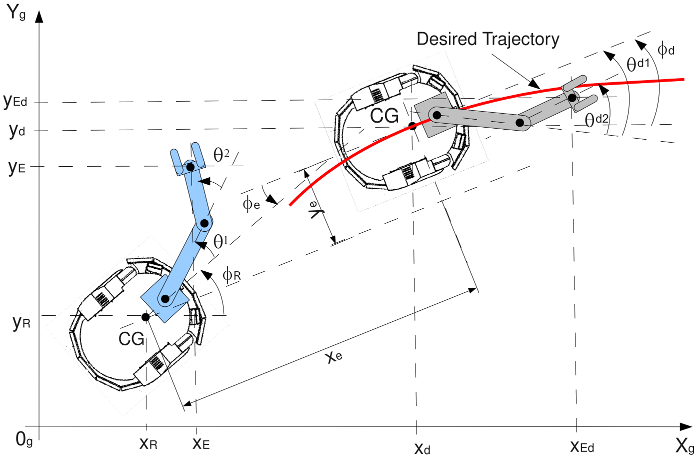


Fig. 4. Lateral, longitudinal and orientation errors for WMM

If s_1 converges to zero, trivially x_e converges to zero. If s_2 converges to zero, in steady-state it becomes $\dot{y}_e = -\gamma_y \cdot y_e - \gamma_0 \cdot sgn(y_e) \cdot \phi_e$. For $y_e < 0 \Rightarrow \dot{y}_e > 0$ if only if $\gamma_0 < \gamma_y \cdot |y_e|/|\phi_e|$. For $y_e > 0 \Rightarrow \dot{y}_e < 0$ if only if $\gamma_0 < \gamma_y \cdot |y_e|/|\phi_e|$. Finally, it can be known from s_2 that convergence of y_e and $\dot{y}_e$ leads to convergence of ϕ_e to zero. If s_3, s_4 converges to zero, trivially θ_{e1}, θ_{e2} converges to zero.

The reaching law is a differential equation which specifies the dynamics of a switching function S. Gao and Hung [16] proposed a reaching law which directly specifies the dynamics of the switching surface by the differential equation

$$\dot{s}_i = -p_i \cdot s_i - q_i \cdot sgn(s_i) \tag{11}$$

where $p_i > 0$, $q_i > 0$, $i = 1, 2, 3, 4$.

By adding the proportional rate term $p_i \cdot s_i$, the state is forced to approach the switching manifolds faster when s is large. It can be shown that the reaching time is finite, and is given by:

$$T_i = \frac{1}{p_i} \cdot ln \frac{p_i \cdot |s_i| + q_i}{q_i} \tag{12}$$

From the time derivative of (9) and using the reaching laws defined in (11) yields:

$$\begin{array}{l} \ddot{x}_e + \gamma_x \cdot \dot{x}_e = -p_1 \cdot s_1 - q_1 \cdot sgn(s_1) \\ \ddot{y}_e + \gamma_y \cdot \dot{y}_e + \gamma_0 \cdot sgn(y_e) \cdot \dot{\phi}_e = -p_2 \cdot s_2 - q_2 \cdot sgn(s_2) \\ \ddot{\theta}_{e1} + \gamma_{\theta_1} \cdot \dot{\theta}_{e1} = -p_3 \cdot s_3 - q_3 \cdot sgn(s_3) \\ \ddot{\theta}_{e2} + \gamma_{\theta_2} \cdot \dot{\theta}_{e2} = -p_4 \cdot s_4 - q_4 \cdot sgn(s_4) \end{array} \tag{13}$$

From (2), (5) (10) and (13), and after some mathematical manipulation, we get the output commands of the sliding-mode trajectory-tracking controller:

$$\begin{aligned}
&\dot{v}_{cR} = \frac{1}{cos(\phi_e)} \cdot (-p_1 \cdot s_1 - q_1 \cdot sgn(s_1) - \gamma_x \cdot \dot{x}_e - y_e \cdot \dot{\omega}_d - \dot{y}_e \cdot \omega_d + \\
&+ v_R \cdot \dot{\phi}_e \cdot sin(\phi_e) + \dot{v}_d) \\
&\omega_{cR} = \frac{1}{v_R \cdot cos(\phi_e) + \gamma_0 \cdot sgn(y_e)} \cdot (-p_2 \cdot s_2 - q_2 \cdot sgn(s_2) - \gamma_y \cdot \dot{y}_e - \\
&- \dot{v}_r \cdot sin(\phi_e) + x_e \cdot \dot{\omega}_d + \dot{x}_e \cdot \omega_d) + \omega_d \\
&\dot{\omega}_{c\theta_1} = -p_3 \cdot s_3 - q_3 \cdot sgn(s_3) - \gamma_{\theta_1} \cdot \dot{\theta}_{e1} \\
&\dot{\omega}_{c\theta_2} = -p_4 \cdot s_4 - q_4 \cdot sgn(s_4) - \gamma_{\theta_2} \cdot \dot{\theta}_{e2}
\end{aligned} \tag{14}$$

The signum functions in the control laws were replaced by saturation functions, to reduce the chattering phenomenon [17].

4 Simulations and Results

In this section, simulation results for the proposed sliding-mode controllers are presented. The simulation are performed in Matlab/Simulink environment to verify behavior of the controlled system. The parameters of the WMM model were chosen to correspond as closely as possible to the real experimental robot presented in Fig. 1 in the following manner: D = 0.25 [m], L_1 = 0.20 [m], L_2 = 0.40 [m], b = 0.04 [m], r = 0.04 [m]. Wheel velocity commands, are sent to the power modules of the follower mobile robot, and encoder measures N_R and N_L are received in the robots pose estimator for odometric computations.

$$\omega R_{right} = \frac{v_{cR} + b \cdot \omega_{cR}}{r}; \ \omega R_{left} = \frac{v_{cR} - b \cdot \omega_{cR}}{r} \tag{15}$$

Figure 5 shows a block diagram of the proposed sliding-mode controller.

4.1 Scenario 1

The first scenario captures a situation where have to maneuver the WMM from an initial to a final state without pose error. The corresponding states and workspace of the simulation are tabulated below (see Table 1).

Table 1. Initial and final pose - Scenario 1

Name	Value
Initial position of end-effector	$x_E = 0, y_E = 0$
Initial angular position	$\phi_R = 0, \theta_1 = \pi/4, \theta_2 = \pi/4$
Final position of end-effector	$x_E = 4, y_E = 2$
Final angular position	$\phi_R = -\pi/4, \theta_1 = -\pi/4, \theta_2 = -\pi/8$

The results of the Scenario 1 are given in Figures 6, 7(a) and 7(b). Figure 6 shows the desired and real trajectory of the mobile platform and the real trajectory of the end-effector. In figures 7(a) presents the desired and real velocities (linear and angular) of mobile platform and the relative angles for first and second link of the manipulator. In Fig. 7(b) one can observe the performances of sliding-mode controllers.

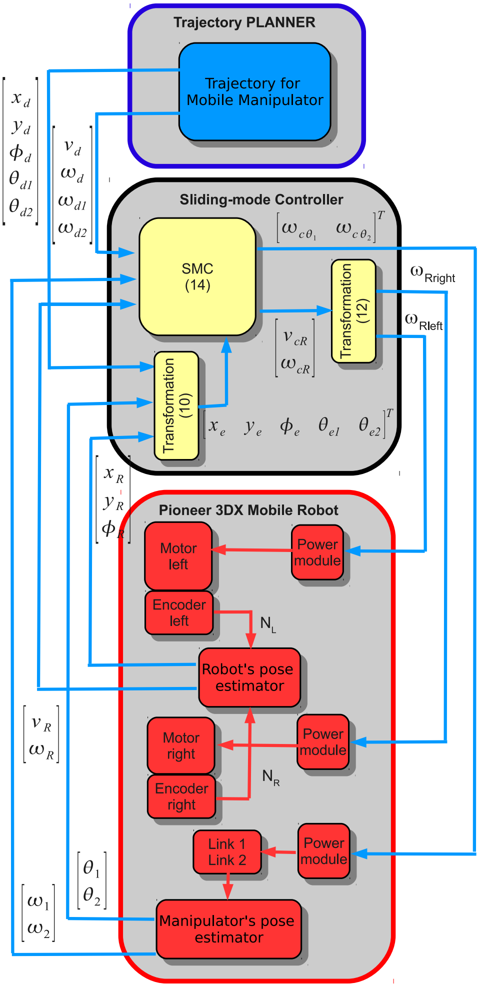


Fig. 5. Sliding-mode trajectory tracking control architecture for WMM

4.2 Scenario 2

The second scenario captures a situation where have to maneuver the WMM from an initial to a final state with initial pose error. The corresponding states and workspace of the simulation are tabulated below (see Table 2).

The results of the Scenario 2 are given in Figures 8, 9(a) and 9(b). Figure 8 shows the desired and real trajectory of the mobile platform and the real trajectory of the end-effector. In figures 9(a) presents the desired and real velocities (linear and angular) of mobile platform. In Fig. 9(b) one can observe the performances of sliding-mode controllers. All the initial errors asymptotically converge to zero.

Table 2. Initial and final pose - Scenario 2

Name	Value
Initial pos. of end-effector	$x_E = 0, y_E = 0$
Initial angular position	$\phi_R = 0, \theta_1 = \pi/8, \theta_2 = -\pi/4$
Final pos. of end-effector	$x_E = 3, y_E = 3$
Final angular position	$\phi_R = 0, \theta_1 = \pi/4, \theta_2 = -\pi/8$
Initial pose errors of WMM	$x_e = -0.20, y_e = -0.6$

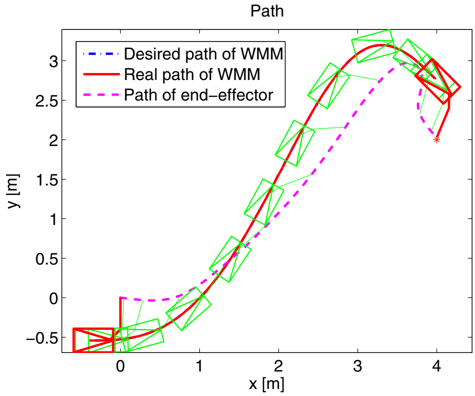
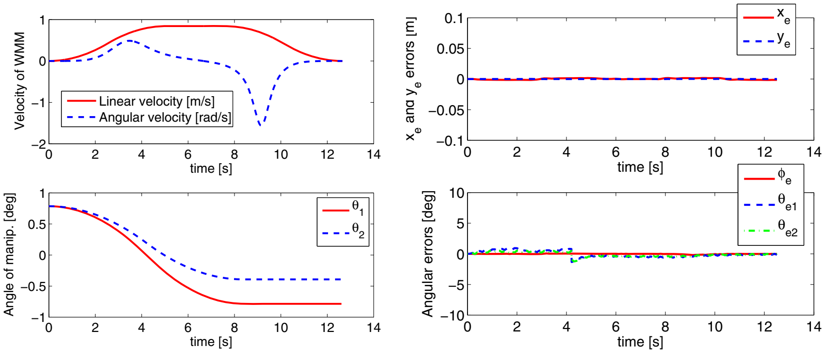


Fig. 6. Scenario 1 - Path of mobile manipulator



(a) (b)

Fig. 7. Scenario 1 - a) Desired velocities for WMM and the relative angles for first and second link of the manipulator. b) Evolution of the errors.

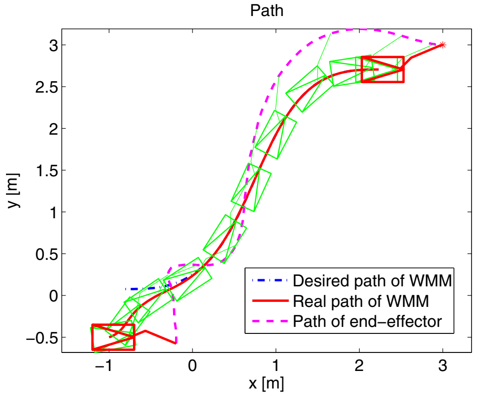
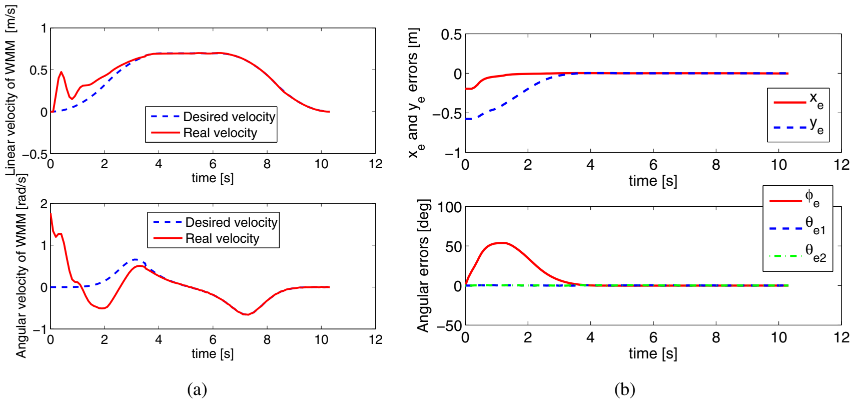


Fig. 8. Scenario 2 - Path of mobile manipulator



(a)

(b)

Fig. 9. Scenario 2 - a) Desired and real velocities for WMM. b) Evolution of the errors.

5 Conclusions

This paper focuses on the motion planning problem of mobile manipulator systems, i.e. manipulators attached on mobile platforms. The paper presents a methodology for generating trajectories for both the mobile plat- form and the manipulator that will take a system from an initial configuration to a pre-specified final one, without violating the nonholonomic constraint. Sliding mode control as a sub-class of variable structure control (VSC) is the most acceptable approach to controller design in these cases. This control scheme provides a simple but effective means of harnessing control laws of nonlinear systems.

The framework developed here lends itself well to implementations on larger systems with further addition of mobile manipulator modules.

Future research lines include the experimental validation of our control scheme and the extension of our results to wheeled mobile manipulators. Future investigations will cover the more general case of coordinated system, consisting of multiple mobile manipulators carrying a common object. The purpose of controlling such coordinated system is to control the object in the desired motion.

Acknowledgements. This work was supported by CNCSIS-UEFISCSU, project PNII-IDEI 506/2008.

References

1. Seraji, H.: An on-line approach to coordinated mobility and manipulation. In: Proceedings of the 1993 IEEE International Conference on Robotics and Automation, Atlanta, USA, vol. 1, pp. 28–35 (1993)
2. Kolmanovsky, I., McClamroch, N.: Developments in nonholonomic control problems. IEEE Control Systems Magazine 15(6), 20–36 (1995)
3. Tanner, H.G., Loizou, S., Kyriakopoulos, K.J.: Nonholonomic navigation and control of cooperating mobile manipulators. IEEE Transactions on Robotics and Automation 19(1), 53–64 (2003)
4. Sharma, B.N., Vanualailai, J., Prasad, A.: Trajectory planning and posture control of multiple mobile manipulators. International Journal of Applied Mathematics and Computation 2(1), 11–31 (2010)
5. Murray, R.M.: Recent research in cooperative control of multi-vehicle systems. Journal of Dynamic Systems, Measurement and Control 129(5), 571–583 (2007)
6. Klancar, G., Matko, D., Blazic, S.: Wheeled mobile robots control in a linear platoon. Journal of Intelligent and Robotic Systems 54(5), 709–731 (2009)
7. Mazo, M., Speranzon, A., Johansson, K., Hu, X.: Multi-robot tracking of a moving object using directional sensors. In: IEEE International Conference on Robotics and Automation - ICRA 2004, vol. 2, pp. 1103–1108 (2004)
8. Zavlanos, M.M., Pappas, G.J.: Dynamic assignment in distributed motion planning with local coordination. IEEE Transaction on Robotics 24(1), 232–242 (2008)
9. Fruchard, M., Morin, P., Samson, C.: A framework for the control of nonholonomic mobile manipulators. In: Rapport De Recherche INRIA, vol. 5556, pp. 1–52 (2005)
10. Bayle, B., Fourquet, J.-Y., Lamiraux, F., Renaud, M.: Kinematic control of wheeled mobile manipulators. In: Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems, Switzerland, vol. 1, pp. 1572–1577 (2002)
11. De Luca, A., Oriolo, G., Giordano, P.R.: Kinematic Control of Nonholonomic Mobile Manipulators in the Presence of Steering Wheels. In: IEEE International Conference on Robotics and Automation, Alaska, USA, pp. 1792–1798 (2010)
12. Tang, C.P., Miller, P.T., Krov, V.N.: Ryu, Ji-C., Agrawal, S.K.: Kinematic control of a nonholonomic wheeled mobile manipulator - a differential flatness aproach. In: Proceedings of ASME Dynamic Systems and Control Conference, Ann Arbor, Michigan, USA, pp. 1–8 (2008)
13. Brockett, R.W.: Asymptotic Stability and Feedback Stabilization. In: Brockett, R.W., Millman, R.S., Sussmann, H.J. (eds.) Differential Geometric Control Theory, pp. 181–191. Birkhauser, Boston (1983)

14. Solea, R., Cernega, D.C.: Sliding Mode Control for Trajectory Tracking Problem - Performance Evaluation. In: Alippi, C., Polycarpou, M., Panayiotou, C., Ellinas, G. (eds.) ICANN 2009. LNCS, vol. 5769, pp. 865–874. Springer, Heidelberg (2009)
15. Solea, R., Nunes, U.: Trajectory planning and sliding-mode control based trajectory-tracking for cybercars. In: Integrated Computer-Aided Engineering, vol. 14(1), pp. 33–47. IOS Press, Amsterdam (2007)
16. Gao, W., Hung, J.: Variable structure control of nonlinear systems: A new approach. IEEE Transactions on Industrial Electronics 40(1), 45–55 (1993)
17. Slotine, J., Li, W.: Applied Nonliner Control. Prentice Hall, New Jersey (1991)

Study and Development of the Rescue Robot to Accommodate Victims under Earthquake Disasters

Taku Sahashi[1,2], Akira Sahshi[2], Hisashi Uchiyama[1], and Ichiro Fukumoto[1]

[1] Nagaoka University of Technology, 1603-1 Kamitomiokamachi, Nagaoka, Niigata, Japan
[2] Project-I Co. Ltd., 154-403 Utasato-cho, Nishi-ku, Nagoya, Aichi, Japan
t.sahashi.project-i@nifty.com, project-i@s7.dion.ne.jp,
{utiyama,ichiro}@vos.nagaokaut.ac.jp

Abstract. At the Great Hanshin earthquake in Japan, many victims were crushed by rubble and died by "crush syndrome". Even the simple rescue method that removes the rubble, it could be a cause of death. As the only countermeasure, DMAT (=Disaster Medical Assistance Team) creeps into fallen houses, and treat dialysis or drip infusion. But these methods are very dangerous for victims and DMAT to treat amid the aftershocks. We think that Rescue Robot technique is just suitable solution for the dilemma under various and dangerous situations. The Rescue Robot has the dual tiered crawler. The Rescue Robot goes through under rubble open, and the victims are held in the container inside of the Robot body. We already built a prototype model, so we present an outline of our rescue robot. And we will show you the reformed design of the Rescue Robot, and the new mechanism to prevent Crush Syndrome.

Keywords: Disaster, Earthquake, Rescue Robot, Crush Syndrome.

1 Introduction

Japan has 0.25 percent of global land in the world. Compared to it, the frequency of earthquakes above magnitude 6 is 20.8 percent (see Fig. 1), and the number of active volcanoes is 7.0 percent compared with the world total.

Japan is a small country, but there has the most frequent earthquake.

This year, our country was attacked by "Great East Japan Earthquake". In this earthquake, about 20,000 peoples were dead or missing. And this earthquake has brought the worst damage in Japan after world war 2nd. This earthquake proved that disaster preparation of Japan was completely poor.

1.1 Problem with Current Disaster Medical Engineering (ME) in Japan

Professor Fukumoto of Nagaoka University of Technology, who was performed disaster medical treatment at 2004 Chu-etsu earthquake and 2007 Chu-etsu offshore earthquake in Japan. He got a shock because disaster ME was not progressed since Great Hanshin earthquake 1995. In shelter, medicines, water and electricity were not enough supplied for several days cause of lifeline damage. Most medical equipments

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 89–100.
springerlink.com

were broken by falling on the floor or/and power failure, and the run out of manpower caused by many patients rushing to an Alternative Medical Treatment Site at the same time. The doctor was not able to take even a short break for 4 days.

If this situation was ignored, the likelihood of medical negligence of doctors increases, and decrease doctors who voluntarily treat in a disaster time.

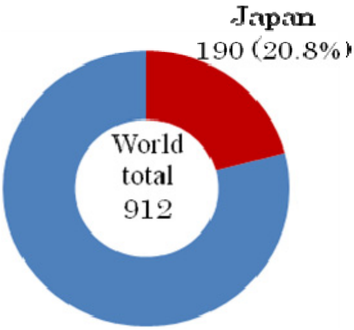


Fig. 1. Number of earthquakes above magnitude 6. (Copied from Homepage of Japanese Cabinet Office, http://www.bousai.go.jp/1info/pdf/saigaipanf.pdf)

1.2 The Disaster Medical Assistance System

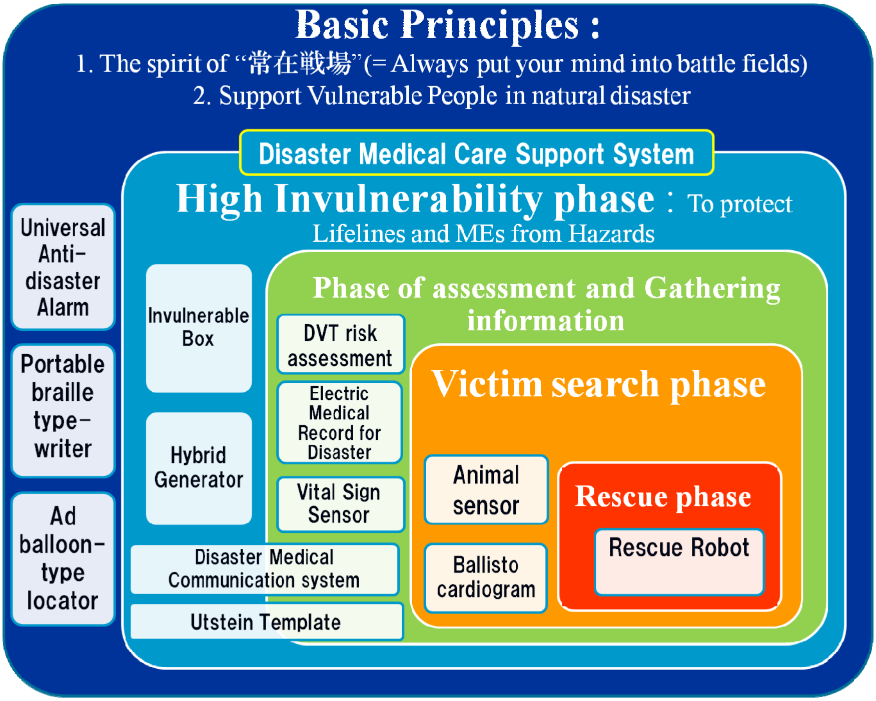

Basic Principles :
1. The spirit of "常在戦場"(= Always put your mind into battle fields)
2. Support Vulnerable People in natural disaster
Disaster Medical Care Support System
High Invulnerability phase : To protect Lifelines and MEs from Hazards
Universal Anti-disaster Alarm
Portable braille type-writer
Ad balloon-type locator
Invulnerable Box
Hybrid Generator
Disaster Medical Communication system
Utstein Template
Phase of assessment and Gathering information
DVT risk assessment
Electric Medical Record for Disaster
Vital Sign Sensor
Victim search phase
Animal sensor
Ballisto cardiogram
Rescue phase
Rescue Robot

Fig. 2. Outline of the Disaster Medical Assistance System

We, “The Disaster ME laboratory” are aiming to construct “the Disaster Medical Assistance System” (see Fig. 2) to solve these problems. We have been designed to support disaster medical care through medical engineering that based on bitter experiences of these earthquakes.

All devices of “the Disaster Medical Assistance System” have correlations with each other in terms of by functioning systematically and dealing with various disaster situations.

The Rescue Robot is one of the devices of the Disaster Medical Assistance System. It is belonging to the center of this system that was defined as “Rescue phase”.

In other words, this Rescue Robot is not intended to operate alone. This Robot will be able to work only after embedding this system.

2 The Rescue Robot

We will introduce “The dual crawler-type Rescue Robot” to prevent crush syndrome.

Fig. 3 is the picture which prototype model of the robot was demonstrated in “The safe and secure festival in Yokohama 2009”.

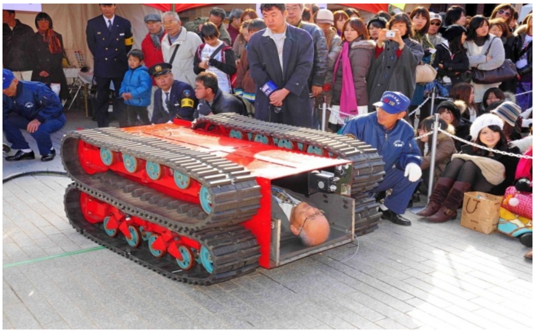


Fig. 3. Appearance of the Rescue Robot

2.1 Necessity of the Rescue Robot

In the Hanshin and Chu-etsu earthquake, many victims were crushed by rubble. But many victims died by symptom of “crush syndrome”, too. Even the simple rescue method that remove the rubble, it could be a cause of death.

As the only countermeasure, Disaster medical assistance team (DMAT) creeps into fallen houses, and performs confined space medicine (CMS) that treats dialysis or administration of intravenous fluids at present.

These methods are too dangerous to treat amid the aftershocks for both victims and DMAT.

Nonetheless, it is problem that cannot begin the rescue operation until safety is secured.

We think that the Rescue Robot technique is just suitable solution for the dilemma under various and dangerous situations.

2.2 Crush Syndrome

Crush syndrome is a form of traumatic rhabdomyolysis (= a condition in which damaged skeletal muscle tissue breaks down rapidly) that occurs after prolonged continuous pressure and characterized by systemic involvement. (see Fig. 4)

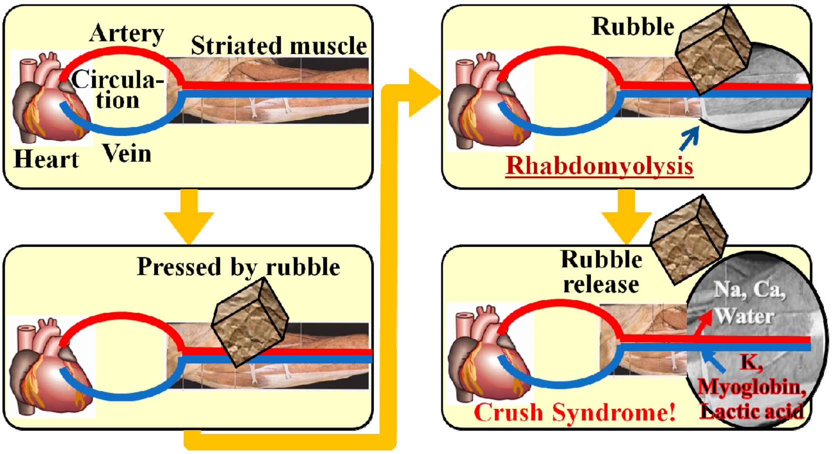


Fig. 4. Mechanism of Crush Syndrome

Extensive muscle crush injury culminating in the crush syndrome is often lethal unless treated promptly and vigorously.

The damages are seen after a prolonged period of pressure on a muscle group. The pressure causes necrosis of the muscle, and during revascularisation, diffusion of calcium, sodium, and water into the damaged muscle cells is seen, together with loss of potassium, phosphate, lactic acid, myoglobin, and creatinine kinase. These changes can lead to hyperkalemia (= medical emergency due to the risk of potentially fatal abnormal heart rhythms), acidosis (=an increased acidity in the blood), acute renal failure, and hypovolemic shock (= cause of lack of blood).

Crush syndrome is commonly encountered after earthquakes, and in various other disasters.

When rubble removed from the victim, myoglobin into the circulatory system leads to myoglobinuria (= the presence of myoglobin in the urine), which causes renal failure if untreated. And its action may result in “reperfusion syndrome” (= acute hypovolemia and metabolic abnormalities). This condition may cause lethal cardiac arrhythmias.

In other word, the simple rescue method that removes the rubble, it could be a cause of death.

2.3 Advantage of the Rescue Robot

Strong Point of Our Crawler System. We wanted to make the Rescue Robot that can work under rubble. But, we could not adopt existing crawler mechanism.

Existing crawler robots are too hard to proceed under rubble. Because the crawler's rotational direction leads that it cannot do through narrow space vertically. (see Fig. 5).

Therefore, we thought out the dual tiered crawler that coupled with crawler runs adverse directions. This crawler system could grip obstructions above, and it was also possible to even use as foothold (see Fig. 6).

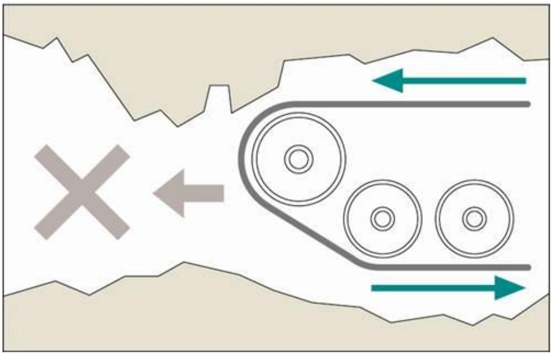


Fig. 5. Problem of existing crawlers

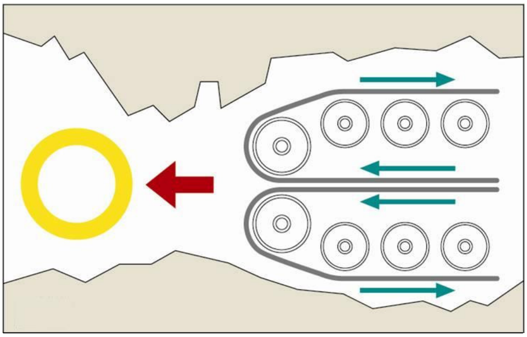


Fig. 6. Advantage of our crawler system

Mechanism of the Crush Syndrome Prevention. As "2.1 Necessary of Rescue Robot", even if the Rescue Robot could reach at the victim under rubble, "remove-rubble" rescue method may cause the victim's Crush syndrome.

Therefore, we designed "the mechanism of the Crush syndrome prevention".

The Rescue Robot is equipped with the cuff into that body. The cuff is inflatable band used in sphygmomanometer (= blood pressure meter).

The cuff is inflated rapidly by using carbon dioxide gas from small-sized cylinder. It is able to stop blood flow around affected part (mostly leg) by inflate cuff pressure. In other words, this cuff works to prevent re-circulating of blood that causes contamination, which prevents the Crash syndrome.

This method is only a first–aid treatment, it is necessary to treatment minutely by doctor after the rescue.

In addition, there is no practical example of this approach. It will require further research and experimentation.

Vital Sign Sensor Probe. Procedure of Crush syndrome prevention is a medical practice. Therefore, measurement of vital signs is essential.

The Vital Sign Sensor Probe is composed of Near Infrared Camera, Infrared LED Light and Microphone/Speaker. These Sensors are equipped with tip of the flexible arm.

The Near Infrared Camera and the Light checks the victim status and blood flow measurement visually. The Microphone is used to determine responsiveness of victims and check their heartbeat. The Speaker is used to call the victim and give their encouragement. (see Fig. 7)

The doctor and operator utilize them for determining the severity of the victims.

Because hemoglobin absorbs near infrared light (wavelength about 780nm), we can confirm superficial veins and the flow.

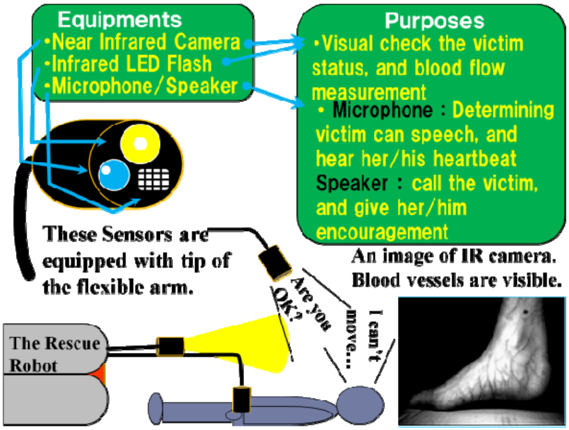

Equipments
•Near Infrared Camera
•Infrared LED Flash
•Microphone/Speaker
Purposes
•Visual check the victim status, and blood flow measurement
• Microphone : Determining victim can speech, and hear her/his heartbeat
Speaker : call the victim, and give her/him encouragement
These Sensors are equipped with tip of the flexible arm.
An image of IR camera. Blood vessels are visible.
Are you OK?
I can't move...
The Rescue Robot

Fig. 7. Outline of the Vital Sign Sensor Probe

2.4 Expected Procedures of the Rescue Operation

First, the Rescue Robot is operated by three peoples and more. As the detail of the member's professions and rolls, Emergency Medical Technician (EMT) as operator, Firefighters (or any can operate) as cable management assistant, and doctor who performed the operational instructions (see Fig. 8). This selection was designed based on the laws of Japan.

This Robot can be controlled remotely by wired control panel. It goes through under rubble with squeezing rubble and reached to victim (see Fig. 9). After removal of rubble, affected parts of victims are tighten smoothly by cuff to prevent crush syndrome. And the victim is picked up in the container inside of the Robot body (see Fig. 10).

After picking up the victim in the container, the Rescue Robot goes back from under rubble to ground. In the safe place to both victims and rescuers, after that the victims are took enough treatment by doctors.

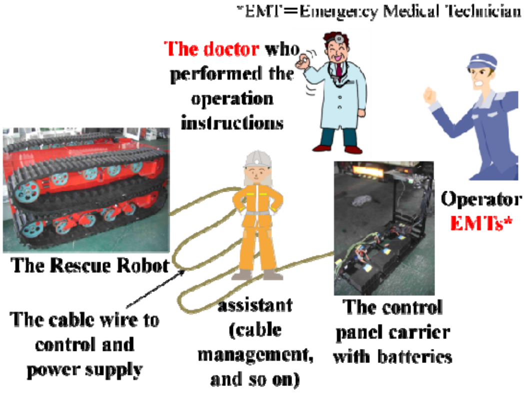

*EMT=Emergency Medical Technician
The doctor who performed the operation instructions
The Rescue Robot
Operator EMTs*
The cable wire to control and power supply
assistant (cable management, and so on)
The control panel carrier with batteries

Fig. 8. Expected operators of the Rescue Robot

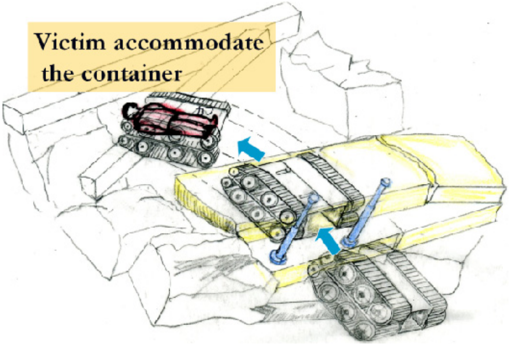


Fig. 9. Assumed figure of the Rescue Robot operation

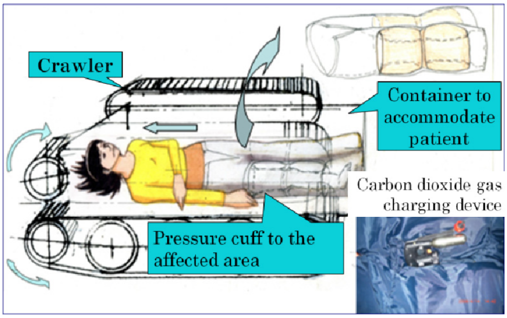

Crawler
Container to accommodate patient
Carbon dioxide gas charging device
Pressure cuff to the affected area

Fig. 10. Assumed figure of putted on the cuff

2.5 Specification of the Rescue Robot

The following are the specs of 1/1-sized prototype model of the Rescue Robot that currently under development. Incidentally, all of figures do not include the figure of “the mechanism to open rubble” and “the mechanism of the Crush syndrome prevention” that cannot be installed to the Rescue Robot currently.

Length: 1.92m

Width: 1.33m

Height: 0.76m

Weight: 329kg (Only the vehicle weight. Not include control panel’s and cable’s)

Engines: 2 motors of CVVF (Constant-Voltage Variable-Frequency) control drive

Batteries: DC12V × 6 = 72V (see Fig. 11)

Controls: Using the remote control panel connected by 10 meters cable wire (Batteries are mounted on control panel carrier)

Sensors: Front and Rear Cameras (and considering mount a roof top camera), the Vital Sign Sensor (made by Near Infrared Camera, Microphone/Speaker and LED Light)

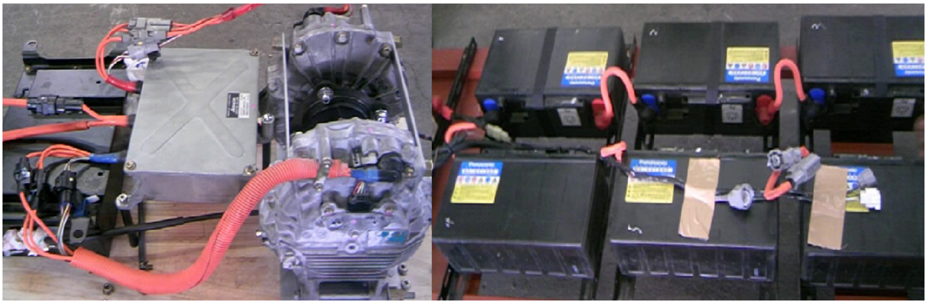


Fig. 11. Motors and Batteries of the Rescue Robot

3 Issues of the Rescue Robot

3.1 Current Issues

Various problems arise in disaster time, but this prototype may be able to withstand these situations. Introduce of the following improvements to the Robot enable the handling of various situations from now on.

1) Improvement of the anti-crush durability of whole parts of this Robot.
2) Waterproof and fireproof.
3) Study and development of the mechanism to open rubble.
4) Improve the performance of mechanism to creep into rubble.
5) Ensure the route to way back.
6) Improvement of the victim accommodation mechanism.
7) Improvement of the mechanism of prevents Crush Syndrome.
8) Improve maneuverability.
9) Improve Safety.

Importantly, 3) to 7) and these quick and smooth cooperation are very important to rescue victims.

3.2 Present Status of the Blade Mechanism to Open Rubble

Fig. 12 shows the latest design of the Rescue Robot.

Separate each functions of "Creep into rubble" and "Accommodate the victim" to each Robots. Therefore, Carrier Robot can specialize to creep into rubble and ensure the escape route, Accommodation Robot can specialize to sensitive work that treatment victim's condition.

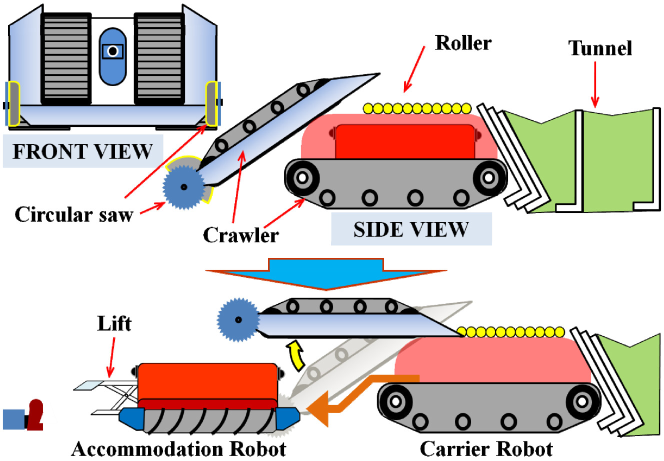


Fig. 12. Expected design of "The Robot in Robot style Rescue Robot"

Carrier Robot. Carrier Robot has sturdy body to withstand load of rubble and Blade mechanism mount with crawler to open rubble. Circular saw is cutting rubbles to lift it easily. By making a simple tunnel with sturdy cloth and flame to secure the escape route. (see Fig. 13).

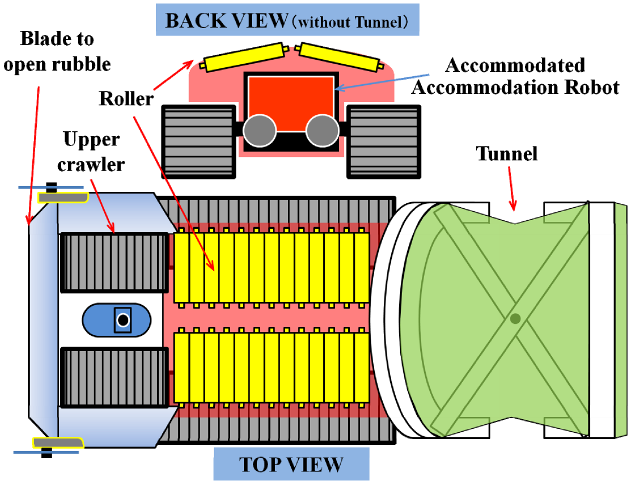


Fig. 13. Expected design of the Carrier Robot

If this Carrier Robot cannot escape from under rubble by itself, we will leave this Robot after made the route and Accommodation Robot is launched. Because we can dig it by heavy equipments after all rescue processes were finished.

Accommodation Robot. Fig. 14 shows expected design of Accommodation Robot. Before this robot starts the work, robot is already creeping into rubbles by Carrier Robot, so accommodation robot doesn't need to equip crawler belt. The lift power of accommodation robot is not so strong as like Carrier Robot's, on the other hand, it can move the victim more and more sensitive.

We adopt an Archimedes' screw drive system to underbody of this robot. Because…

1) It has high performance against bad surface.

2) It can easy to change direction and horizontal movement.

3) Structure is simple, whereby it can take high reliability.

4) This drive system has poor energy efficiency, but it is not so big problem because operational distance of the Accommodation Robot is short. (Maximum within 100m).

Pressure Cuff, we plan to use "Two-step inflate method" type cuff that described below.

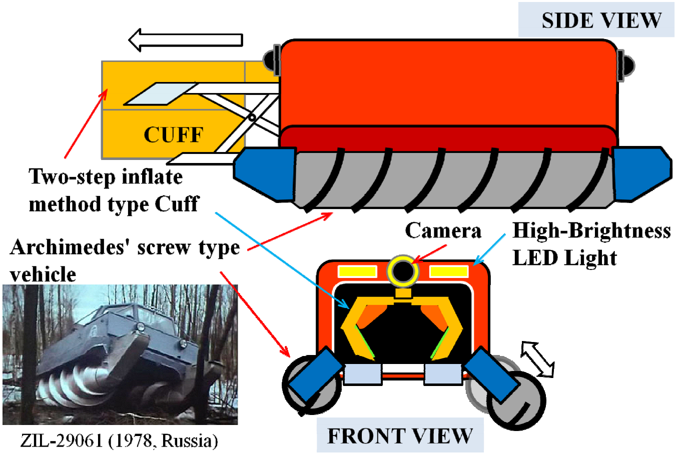

SIDE VIEW
CUFF
Two-step inflate method type Cuff
Archimedes' screw type vehicle
Camera
High-Brightness LED Light
ZIL-29061 (1978, Russia)
FRONT VIEW

Fig. 14. Expected design of the Accommodation Robot

3.3 Present Status of the Mechanism to Prevent Crush Syndrome

Fig. 15 is the latest design of the mechanism to prevent Crush syndrome.

This mechanism needs to touch the living human body directly, it is requires a very delicate design. For example, the disaster scene severity, victim's body position, place of the affected area, and so on.

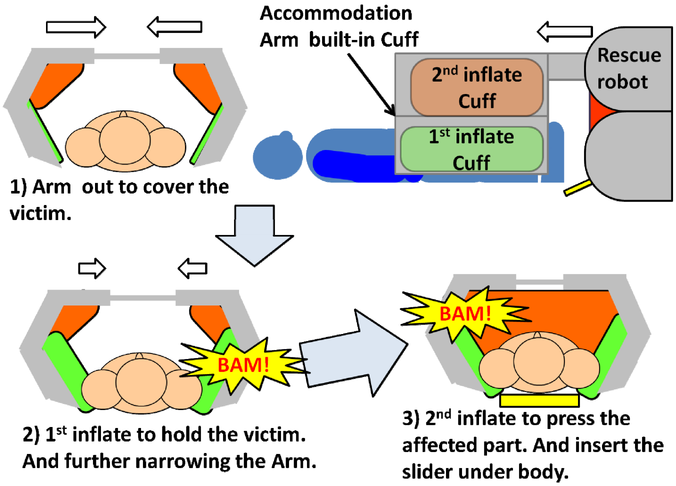

Accommodation
Arm built-in Cuff
2nd inflate Cuff
1st inflate Cuff
Rescue robot
1) Arm out to cover the victim.
BAM!
BAM!
2) 1st inflate to hold the victim. And further narrowing the Arm.
3) 2nd inflate to press the affected part. And insert the slider under body.

Fig. 15. Proposed design and its procedures of "Two-step inflates method"

The advantage of this method will not have to touch the victim directly by the mechanical arm. And it can make the mechanism simply, and thus it can be expected to improve reliability. This cuff system can deal with victim's various positions. (Head side or foot side, supine or prone).

However, we will have to conduct a detailed examination, including the shape of Cuff that can prevent Crush syndrome effectively or not.

4 Conclusions

In this study, we produced the first 1/1 sized prototype model. And we have designed new Rescue Robot to improve their problems that found by its prototype development.

Now we making 1/5 sized model of the new designed Rescue Robot to verify its performance and more improvement. And we will start to make medical tests (top priority is Crush syndrome) using 1/1 sized prototype in the near future. But many issues still remain to perform medical tests before.

This robot was born from the needs of experience in an actual disaster medicine. Our development could not in time for the Great East Japan Earthquake. It made us deeply regret.

In order to prepare for next big earthquake, we will finish the research and development as soon as possible.

Acknowledgements. I received generous support from member of Disaster ME Lab. and Fukumoto Lab. Especially, Ms. Wen-Hsin Lin and Mr. Tomoyuki Fujiki of Disaster ME Lab., They gave me a grateful help to my poor English language skill.

In addition, a part of this work was supported by International Risk Management Program 2006 by The Japanese Ministry of Health Labour and Welfare.

References

1. Sahashi, T., Sahshi, A., Uchiyama, H., Fukumoto, I.: Study and Development of the Rescue Robot preventing Crush Syndrome of earthquake victims. In: 8th International Conference on Informatics in Control, Automation and Robotics, Final Program and Book of Abstract of ICINCO 2011 and SIMULTECH 2011, p. 55 (2011)
2. Tadokoro, S.: Technical Challenge of Rescue Robotics. Journal of the Robotic Society of Japan. The Robotic Society of Japan 28(2), 134–137 (2010)
3. Sahashi, T., Sahshi, A., Uchiyama, H., Fukumoto, I.: Development and management researches of medical rescue robot under natural disasters. In: The 10th Asia Pacific Conference on Disaster Medicine, Program Book of 10th APCDM, p. 212 (2010)
4. Sahashi, A., Sahashi, T., Uchiyama, H., Fukumoto, I.: A developmental research of the disaster medical assistant equipment systems which does not isolate a stricken area under natural disasters. In: Proceedings of the 29th JSMBE Kou-Shin-Etsu Branch Conference 2010, Japanese Society for Medical Biological Engineering, pp. 3–4 (2010)
5. Sahashi, T., Sahshi, A., Uchiyama, H., Fukumoto, I.: Development and operational liability of Rescue robot under natural disaster. In: Proceeding of the Robotic Society of Japan (DVD), pp. 3G1–7 (2010)
6. Sahashi, T., Sahshi, A., Uchiyama, H., Fukumoto, I.: A basic study of automated medical rescue robot under natural disaster. In: The 9th Asia Pacific Conference on Disaster Medicine, Program Book of 9th APCDM (2009)
7. Fukumoto, I.: A study of volunteers and personal burden for the public under natural disasters. Japanese Journal of Disaster Medicine. Japanese Association for Disaster Medicine 13(2), 218–225 (2008)
8. Higuchi, N.: Good Samariyan Act and Phisicians' Duty to Rescue (Series-Law and Medicine(8)). Japanese Journal of Radiological Technology. Japanese Society of Radiological Technology 63(3), 382–384 (2008)
9. Mahoney, E.L., Whiteside, F.D., Belue, E., Mortisugu, P.K., Esch, H.V.: Disaster medical assistance teams. Annals of Emergency Medicine. Journal of the American College of Emergency Physicians 16(3), 354–358 (1987)
10. Ensari, C., Tüfekçioglu, O., Ayli, D., Gümüs, T., Izdes, S., Turanli, S.: Response to Delayed Fluid Therapy in Crush Syndrome. Nephron Journals. Karger AG 92(4), 941–943 (2002)
11. Greaves, I., Porter, K.M.: Consensus statement on crush injury and crush syndrome. Accident and Emergency Nursing 12(1), 47–52 (2004)
12. Demirkiran, O., Dikmen, Y., Utku, T., Urkmez, S.: Crush syndrome patients after the Marmara earthquake. Emergency Medicine Journal 20, 247–250 (2003)

Digital Traveler Assistant

Andreea Radu[1], Leon Rothkrantz[1,2], and Mirko Novak[3]

[1] Department Man Machine Interaction, Delft University of Technology
Mekelweg 4, Delft, The Netherlands
[2] Department of SEWACO, The Netherlands Defence Academy
Nieuwe Diep 8, Den Helder, The Netherlands
[3] Faculty of Transportation Sciences, Czech Technical University
Konvitska 20, Prague, Czech Republic
L.J.M.Rothkrantz@tudelft.nl

Abstract. Currently most car drivers use static routing algorithms based on the shortest distance between start and end position. But the shortest route is different from the fastest route in time. Because existing routing algorithms lack the ability to react to dynamic changes in the road network, drivers are not optimally routed. The current traffic situation can be assessed by tracking car drivers provided with a smart GPS device. The real challenge is to predict future delays in travelling time. In this paper we present a multi-agent approach for routing vehicle drivers using historically-based traffic information. we successfully implemented a working prototype that uses various technologies such as Java, the Open Street Map API for rendering the map or J2ME for the mobile phone client.

Keywords: Dynamic Routing, Predicting Travelling Time, Personal Assistant, Hand-held Devices.

1 Introduction

In densely populated regions the capacity of roads is not sufficient to optimally route all drivers from a source point to a destination point. During the last years the burden caused by growth of traffic intensity and frequency of traffic jams on major roads, highways and in urban areas has been stigmatized. However, regrettably enough, the expansion of infrastructure cannot always keep up with these increased demands so other solutions have to be taken into account to solve traffic problems. Congestion causes loss of time and money, it damages the environment, encourages energy waste and it affects our health. In fact we have to think about travelling in a completely different way. We need to make use of the already existing road networks and we need to make smart choices about the way in which we travel.

In many countries there are research programs, therefore, which deal with the causes and effects of congestion. Mechanisms for congestion prediction have been considered and countermeasures affecting road infrastructure (e.g. reversible lanes, improving network junctions, separate lanes for specific user groups), supply and demand traffic management (e.g. adding more capacity at bottlenecks, road pricing, road space rationing) as well as intelligent transportation systems facilities (e.g. traffic reporting, variable message signs, navigation systems, automated highway systems

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 101–114.
springerlink.com

etc.) are being applied. These are of limited help as they offer no alternative route when traffic jams. In The Netherlands, during every workday there are more than two million Dutch drivers caught in traffic-jam (see Fig 1).

Fig. 1. Traffic congestion in The Netherlands

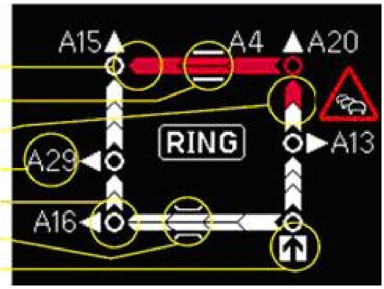


Fig. 2. Dynamic routing information panel (DRIP) on the ring of Rotterdam

At some time every road user is mainly on highways about the current movement conditions using DRIP (Dynamic Route Information Panel). DRIPs are programmable graphic and textual panels, where symbols and/or text can be displayed. These panels allow the user to have better access at traffic information in a graphical form (see Figure 2). Route information such as travel times (in minutes) or traffic-jam lengths (in kilometers) for a specific route can also be provided. Besides, there is the alternative of warning users about incidents, about upcoming congestion and about certain weather condition like fog or ice. Through these panels information traffic managers attempt to improve road conditions.

Surveys have indicated that motorists’ main concern is the uncertainty of their journey times. Because of that, several automotive navigation manufacturers have developed and advertised navigation systems for congestion avoidance. These devices have an integrated real-time traffic and road incident receiver to display information based on Traffic Message Chanel (TMC), RDC or GPRS/3G. For example, the navigation manufacturer Tom Tom provides real-time traffic information based on cellular floating phone data system which exploits data from anonymous cell phone users and is enhanced by GPS-based probe information and other third party messages. A similar application is the Mobile Millennium Project in Berkeley California, which uses anonymous speed and position information gathered by GPS-equipped mobile phones, fuses it with data from static traffic sensors and broadcasts the traffic information back.

This project was our main source of inspiration to develop our routing device.

The complexity of traffic management is due to the interaction of three main processes [1]:

- the traveller’s decision behavior, as the decisions of drivers influence the outcome of the traffic network;
- the dynamic traffic assignment in a traffic network;
- the traffic flow behavior, in particular when incidents and accidents occur in the network.

One approach to solve the congestion issues and though to reduce the travel time for individuals is to develop a route planner that incorporates current and future traffic information when searching for the best route. When congestions or incidents occur on this route the planner has to compute the best alternative solution which may lead the driver on different roads or to a train station. The main purpose is to minimize the travelling time by taking into account the changes and the future situation in the traffic network. In this area the personal advanced travelling assistants play a crucial role. Such an assistant has multiple functions including that it computes the shortest travelling time-routes based on current information received from traffic.

The outline of the paper is as follows. In the next section we will describe related work on dynamic routing. Next we will describe the used databases of historical travel times. Next to open sources we used to build our prototype. Then we describe the main part the dynamic routing algorithm. We report some experimental results and end up with a conclusion.

Fig. 3. Car drivers use a Digital Traveler Assistant to plan a trip

2 Related Work

Traffic assignment is defined as the problem of finding traffic flows given an origin-destination trip matrix and a set of costs associated to the links. One solution for this problem is either that the driver drives on the optimum path according to his preferences, known as the User Equilibrium (UE) assignment or alternatively the path that minimizes the overall network's travelling time, known as the System Optimum (SO) assignment.

Wardrop was the first one to differentiate the two methods [2]. A spectacular example that actually shows that the UE assignment is in general different from the SO solution is the Braess network. The mathematician Dietrich Braess obtained the paradoxical result that the addition of an arc to the network can result in increased origin to destination and overall travel cost. Fisk studied the Braess paradox more in detail [3]. She presented the sensitivity of travel costs to changes in the input flows while they are in Wardropian equilibrium. Examples which state the fact that an increased capacity of the input flow can decrease the travelling time are presented.

Non-equilibrium methods assign traffic to a single minimum path between two zones. The minimum path infers the minimum travel time. Minimum path algorithms include for example the models developed by Dantzig [4] and Dijkstra [5]. Other non-equilibrium methods include diversion models, multipath assignments and eventually combined methods.

Equilibrium methods are algorithmic approaches which assume equal travel times. They are optimal assignments since they are formulated on the basis of linear or nonlinear mathematical programming [6]. The user optimum equilibrium can be found by solving a nonlinear programming problem.

When a time dimension is added at the models previously described then the DTA is obtained. Thus, by including temporal dimensions we can represent the real life traffic situation and compute the traveling time. Literature surveys in this field generally mention two main approaches for DTA: the analytical-based models and the simulations.

The first approach which is the analytical-based approach model considers two time indices: the time at which the path flow leaves its origin and the time at which it is observed on a link. In other words, the approach assumes that the whole time is divided in intervals. Then, static mathematical analytical control models are applied to each interval, on the assumption that one interval is long enough so that drivers can complete the trip within that certain time interval.

Literature within this area of research is extensive. DTA has evolved a lot since the work of Merchant and Nemhauser [7] who considered a discrete time model for dynamic traffic assignment with a single destination. The model they assumed was nonlinear and non-convex.

Meantime, researchers became aware that DTA theory was still undeveloped and necessitated new approaches to account for the challenges from the application domain. DTA comes across a large set of problems depending on various decision variables, possessing varying data requirements and capabilities of control.

The second approach is the simulation-based model. This approach simulates the behaviour of the drivers in different traffic settings. Due to their capability of better representing the real world they increased their popularity. Simulations usually try to replicate the complex dynamics of the traffic. Although that this is considered a different approach, the mathematical abstraction of the problem is a typical analytical formulation.

Next we consider some analytical-based approaches and mathematical programming models for DTA from literature. Ziliaskopoulus split the analytical models from literature in four broad methodological groups where the first ones are the mathematical programming formulations [8]. Within this approach flow equations are deducted and a nonlinear mathematical programming problem has to be solved. Merchant and Nemhauser [7] and Ho [9] studied such models. Due to the complexity of a nonlinear problem, a linear version of the model with additional constraints can be created and solved for a global optimum using a simplex algorithm. The linear program has a staircase structure and can be solved by decomposition techniques.

In optimal control theory the routes are assumed to be known functions of time and the link flows are considered continuous functions of time. The constraints are similar to the ones at the mathematical programming formulation, but defined in continuous-time setting. This results in a continuous control formulation and not in a discrete-time mathematical program. Friesz et al. [10] discuss two continuous link-based time formulations of the DTA for both the SO and UE objectives considering the single destination case. The model assumes that the adjustments of the system from one state to another may occur while the network conditions are changing. The routing is done based on the current condition of the network but it is continuously modelled as conditions change. The SO model is a temporal extension of the static SO model and proves that at the optimal solution the costs for the O-D used paths are identical to the ones on the unused paths. They established as well a dynamic generalization of the well-known Beckmann's equivalent optimization problem.

Simulation environments address key issues of the traffic assignment, such as the flow's propagation in time and the spatio-temporal interactions. Contemporary DTA models were developed using different traffic simulators (such as CONTRAM (CONtinous TRaffic Assignment Model), DYNASMART or SATURN etc.). SATURN, [11] is an early DTA simulation tool that uses an equilibrium technique.

The CONTRAM, [12] simulation environment is more dynamic than the previous ones as it allows the re-routing of cars if traffic conditions worsen. However, it does not consider a maximum storage capacity for roads and it assigns cars only based on the Wardropian principle. DYNASMART is a contemporary DTA model which uses the basic CONTRAM concept. Abdelfatah and Mahmasanni show an example of a DTA model developed by the DYNASMART approach [13].

Lum at al. showed that the average speed depends on the road's geometry, on the traffic flow characteristics and on the traffic signal coordination [14]. A new travel time-density model was formulated by incorporating the minimum-delay per intersection and the frequency of intersections as parameters. The travelling time and the traffic volume are two main field items that have to be considered for the speed flow study along arterial roads.

Most influencing factors that have been cited in literature are the special incidents and holidays, signal delays, weather conditions and the level of congestion. The prediction error might be also directly proportional with the length of the forecasting period [15].

Most of the short-term forecasting methods that were used in literature can be divided in two categories, namely regression methods and time series estimation methods. A third category can be described as combining these two. Relevant forecasting techniques examples which belong to previous research studies are presented in the following paragraphs. The type of traffic data that was used along with possible inconveniences that we detected is included.

Hobeika and Kim constructed three models for short-term traffic prediction by combining the current traffic, the average historical data and the upstream traffic [16]. Li and McDonald use GPS equipped probe vehicles and determine mean speed values in order to develop a fuzzy mathematical travel time estimation model [17]. Time series analysis is as well a popular method to infer the travel time prediction due to their strong potential for on-line implementation. Ishak et al. describe a short-term prediction model for speed that follows a nonlinear time series approach and uses a single variable [18].

In review of literature, researchers have used parametric models in order to forecast the travel time, such as regression models or time series and nonparametric models that include ANN models [19],[20]. Studies have shown that Ann's (including modular neural network model and state space neural network model) is a powerful tool to predict travel time on freeways [20]. Yu et. al. proposed a travel time prediction model which comprised two parts: a base travel time and a travel time variation [19]. The first term is computed using a fuzzy membership value average of the clustered historical data that reflects the traffic pattern. The variation is predicted through a cluster based ANN in order to capture the traffic fluctuation.

3 Data

Historical data may consist of single vehicles trajectories or it may be in the form of databases of traffic variables measurements recorded at spots on the roadways. The broadcasters along the roadways identify and report the travelling speed of vehicles at fixed time intervals, the number of vehicles or the congestion level. But most of the traffic measurements infer the travelling speed, which is most important for detecting the travelling time.

The raw data is processed in order to obtain traffic indicators, such as the average speed or the congestion level on the roads at fixed intervals. Missing data is usually computed by interpolation from the surrounding data (if it does not exceed a certain interval). Other problems might come up at the analysis stage because if the recordings are not done for each lane there are differences between trucks (which have a different speed limit) and cars. An example of a travel time plot obtained from historical data on highway A9 on the 25th of March 2003 is given in Figure 4.

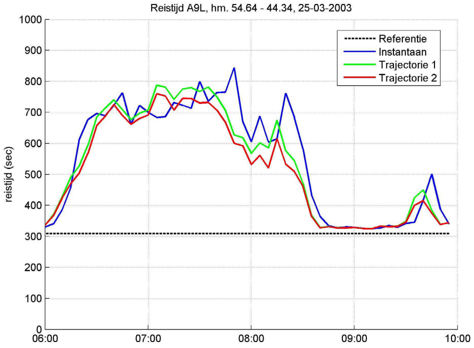


Fig. 4. Dynamic route information

Traffic data can be collected by a variety of data sensors, such as inductive loop detectors (ILD) (see Figure 5), videos, floating cars, remote traffic microware sensors etc. The latter represent a relatively new technology for collecting traffic data. But since it is still in the testing stage, only a limited number of such sensors have been installed in the United States. Therefore, it cannot be used for wide-area data collection.

ANWB is one of the services which attempts to offer a live traffic update for the highways network in The Netherlands. The application is using data from the monitoring system. It shows real life graphical information about the bottlenecks on the highway network by giving an estimate of the current average speeds (see Figure 6). This traffic information is available 24 hours a day on their website and is free of charge. The file that we used to fill in our database was built by collecting data from the ANWB website. The traffic data was collected each 10 minutes for a couple of weeks for a roadway network that comprised the highways and a few national roads from the country. For each road the traveling time was extracted from the text files that are offered by ANWB. All data was organized in an Excel file.

For each road the following information was stored:

- name of the road (such as A1, A2, etc.);
- names of the intersections bounding the road;
- length of the road in kilometres;
- maximum speed allowed on the road;
- associated travelling time computed based on the maximum speed;

starting from 0:00 to 23:50 for each 10 minutes interval the added travelling time in case of congestion.

The missing data in some cases was computed by interpolation from the surrounding data. After the processing and analysis of the collected data, 4 Thursdays were chosen in order to be further used. The file in Excel needed further processing in order to be integrated in our application.

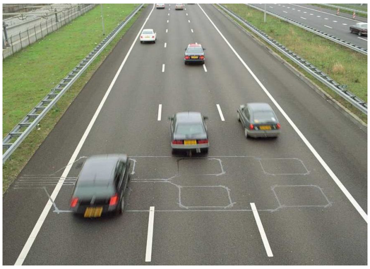
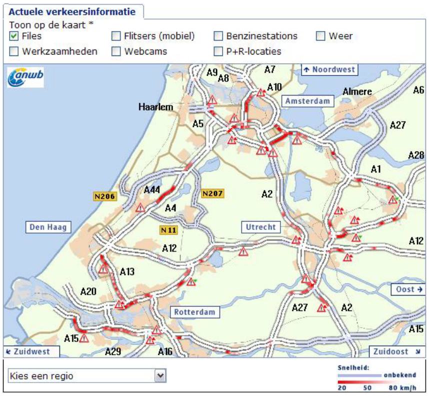


Fig. 5. Wires in the surface of the road

4 Open Source Tools

OpenStreetMap (OSM) is a collaborative project to create a free editable map of the world. OSM follows a similar concept as Wikipedia does, but for maps and other geographical facts. An important fact is that the OSM data does not resume to streets and roads. Anybody can gather location data across the globe from a variety of sources such as recordings from GPS devices, from free satellite imagery or simply from knowing an area very well, for example because they live there. This information then gets uploaded to OSM's central database from where it can be further modified, corrected and enriched by anyone who notices missing facts or errors about the area. OSM creates and provides free geographic data such as street maps to anyone who wants to use them. The OSM project was started because most maps that people think of as free actually have legal or technical restrictions on their use, holding back people from using them in creative, productive, or unexpected ways. Libraries to access the resources provided by the project are available for multiple languages and purposes. As an example, several rendering libraries exist (in Javascript, Python, C and Java) and also several editing clients that allow to interact with the data. In order to implement the graphical user interface of the system and to construct an initial database out of intersections and highways from The Netherlands, we embedded an OpenStreetMap map viewer in the application. This was a Java panel which allowed several listeners and functions to be redefined.



Actuele verkeersinformatie

Toon op de kaart *

☑ Files □ Flitsers (mobiel) □ Benzinestations □ Weer

□ Werkzaamheden □ Webcams □ P+R-locaties

Kies een regio

Fig. 6. Dynamic route information

Due to the modular design of Swing component library, the integration was an easy task.

The API of the OSM viewer provides a number of utilities for manipulating maps, allowing us to construct a robust user interface for our system. We were able to:

- create the intersections for the traffic network graph by clicking on the map.
- create directed links between two nodes - and we assume that all the roads are straight.
- calculate distances in kilometres between nodes by using their latitude and longitude.
- relate geographical coordinates to plane coordinates on the map.
- design (in different colours) paths on the map, parallel to the main roads in order to display a requested route.

An example of the OSM integrated in our application is presented in Figure 7. The intersections and the roads from the database are displayed on the map.

5 Dynamic Dijkstra Traffic Assignment

The starting point of the implementation for a dynamic traffic assignment is to build the traffic network as a time expanded graph. Given the traffic model that we presented in the previous section we need to implement the algorithm on a graph that is extended in time. This is determined by the time varying speed graphs, which we also presented in the previous chapter. Using this representation we can then apply known mathematical algorithms to solve our problem. As for general routing problems, the Dijkstra's shortest path algorithm could be applied. The main difference to a classic traffic assignment, in representing the network graph is that the cost varies in time.

The input to the algorithm is represented by a route request. A route request consists of an O-D (origin-destination) trip demand at a specific time. The algorithm will be applied on the network graph and it uses the nodes, the roads and the estimation of the travelling time based on the varying average speed associated to each road.

The network is represented by a graph $G = \{N,A\}$, where A is the set of directed links and N a set of nodes. G represents the spatial network, meaning the network of nodes and roads. In order to represent the dynamic travel time we will use a time extended network. The time expanded network can be constructed in the following way: the planning horizon is divided into variable time periods $\{t = 1,....T\}$ and each node is copied for each period t so that for each node k there are now T time-space nodes denoted kt. For each link j in the spatial network consider time-spaced links, $jt\tau$ joining the entry node of link j at each time t to the exit nodes of link j at latter times, $\tau = t + 1; t + 2,....$ Thus for each spatial link we have time-expanded links $(jt\tau)$, $\tau = 1..T$, $t = 1....T$. This approach brings one constraint: the travel time has to be discretized to intervals. If we use a very high sample rate then an enormous graph is required whereas a lower sample rate results in loss of information.

An example of a space time extended graph constructed in the modality that we just described is presented in Figure 8. The space graph represented by the nodes (A, B, C, D, E) is repeated for three time intervals (t1 at 09:00, t2 at 09:05 and t3 at 09:10). The edges that connect the nodes from A to E, coloured in red represent the initial connections in the graph. For clarity these edges were kept similar also for the other layers. But the edges in dotted lines are the real connections of the time expanded graph. They show the evolution in time of the speed flow along with the travelling time in the network. Their length, between the layers, represents the travelling time associated to the corresponding edge when starting at each layer. It should be noticed, however, that not all edges were represented in the Figure in order to keep it readable. For example, the travelling time from B to D is 5 minutes at 09:00 and 20 minutes at 09:05. In order to show more clearly which are the differences between the DDTA and STA averages of the travelling time we use Table 1. In this table the first part shows the average time that DDTA gains compared to STA in minutes. Some values here may appear to be lower than expected but these are the averages.

It is worth mentioning that in the worst case DDTA gives a result with the same travelling time as the other algorithms, but never worse. This is the case of the 0

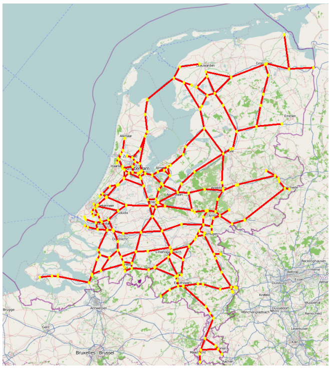


Fig. 7. Dynamic route information

values in the second part of the table. In this part we show at each hour the route with greatest gain in time for DDTA. We notice for example that from Rotterdam to Amsterdam we have a gain of 16.78 minutes at 8 o'clock.

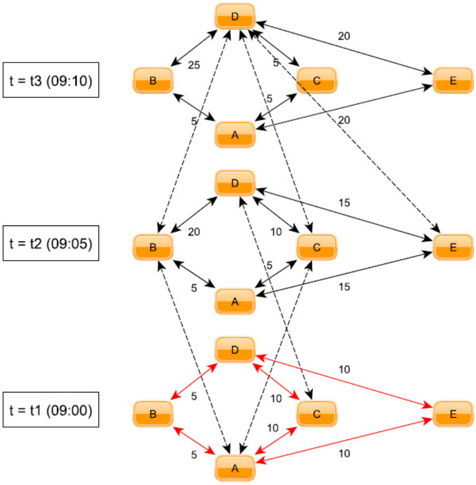


Fig. 8. Dynamic route information

6 Model

Digital Traveler Assistant (DTA) gives the traveller routing advices during his trip starting from departure to destination. The traveller will benefit of the best available solution according to his preferences at the departure time. If unexpected events occur, it will result in modifications during the trip. DTA should distribute the traffic in the network so that it satisfies the preferences of the users by taking into account the availability in the network. The system will use continuously updated traffic flow information. This information would be available from the GPS-equipped mobile phones of the users. Given that the system knows which are the route requests and the routes assigned already to drivers it can give a prediction of the travelling times on the roads in the future. This can be done by training a neural network on the relation between various traffic parameters such as the traffic stream and the travelling time.

If we return to the individual routes assignment, in case of an incident/road work the system informs the traveller on the delays and best alternative solutions. The driver will be also informed on the travelling time associated to the recommended alternative, the types of roads and eventually the advantages/disadvantages. As we already mentioned, DTA connects to the users by a hand held device. This can be a smart phone, a routing device or a PDA.

DTA is a distributed system that links the users to the central server. All components are connected through Internet. Users are connected to the server but they can also communicate among themselves by using ad hoc wireless networks. A possibility to do this is by using the wireless network between light poles on the highways. The advantage of DTA is that if the system becomes a centralized one (because of network problems) it still manages by connecting to the other cars in order to get the information it needs. Figure 9 depicts the architecture of DTA. The main server has to be connected to the roads and nodes database, to the historical database and to an incidents database. As we already mentioned, it is important for DTA to benefit of live traffic information. The system is designed in such a way that it uses the information from the vehicles that already exist in the network. The GPS-equipped mobile phones report their positions at fixed time intervals. Moreover, the routes of the vehicles are supervised, meaning that the system knows the origin, the destination and the departure time of each route. In this way the travelling time of the traffic flow in the future can be estimated.

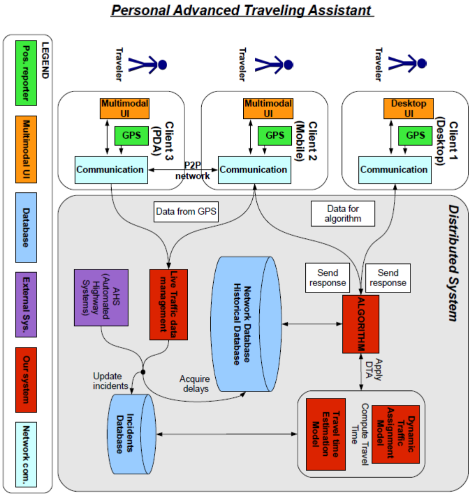


Fig. 9. Dynamic route information

7 Experiments

In order to be able to perform our experiment, testing the possible advantages using a routing algorithm based on historical data we need to define some test routes. Our expectation is that during the rush hours with high traffic streams between big cities, dynamic routing using historical traffic data should provide a better solution than static routing. To ensure reliability of our samples, we have selected 21 cities in The Netherlands with number of populations of more than 100.000. Next we Calculate the travel time of every combination of departure/arrival cities and departure time from 15.20 until 18.40 with 20 minutes interval (T1,….T10. We compute the shortest path by using Dijkstra's algorithm and calculate the total travelling time using historical travelling data. We plot the difference using colour matrices. The range is 0 (blue area) up to more than 30 minutes (red area)We observe that in the rush hours between some cities there are significant differences.

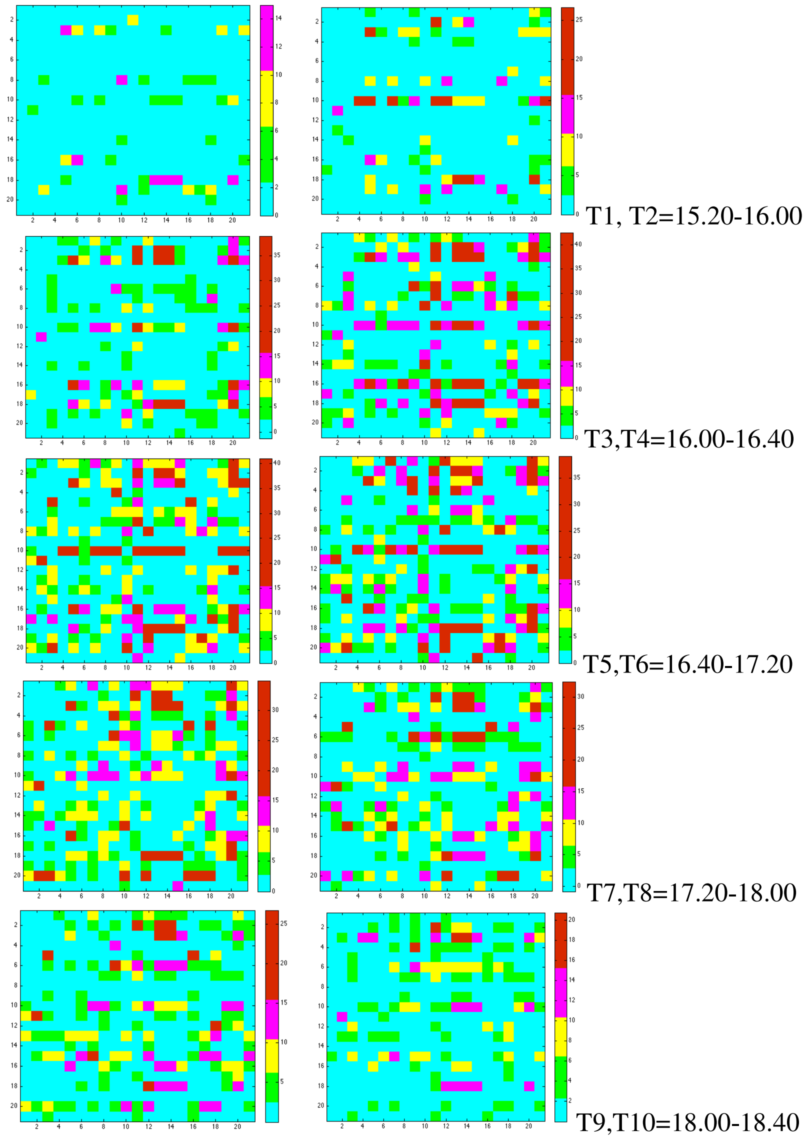


Fig. 10. Visual display of the differences in travel time between computed by the Dijkstra and adapted Dijkstra algorithm

8 Conclusions

As we mentioned in the introduction of this thesis, our main purpose was to build a dynamic traffic assignment. An important component of this is the prediction of the travelling time. In this paper we proposed a prediction method that would update the historical data based on supervising the routes in the network together with the real time traffic information.

Our prototype implements an algorithm that is a time dimensional extended version of Dijkstra shortest path algorithm. The main difference is that our algorithm takes into account the traffic variations in time. The cost function in the algorithm is associated with the travelling time. The algorithm uses the prediction model that we described previously. Because of the fact that the algorithm gives the route with the shortest time to each user we may categorize it as a user equilibrium assignment. However, we assume that just a part of the drivers in the network are connected to the system. If we deal with the whole network the situation would change.

The results of the algorithm were compared to the results of two variants of the static Dijkstra algorithm, one that computes the shortest path and one that uses the fastest roads given their maximum speed limit.

We developed a complex design for an advanced traveller information system that relies on the concept of distributed systems. The system that we designed integrates the use of live traffic information that derives from tracking the individuals and use of highway sensors. Travellers are routed through hand held devices which can be their mobile phones. Another important feature of the system that we designed is that it is usable by everybody, without any special training or knowledge needed. In order to get more insight into users' preferences with regard to such a system we did a user survey that mostly confirmed our expectations but also brought new ideas.

The system is also seen as an intelligent assistant as it has the capability to detect, learn the user's profile and associate it with his schedule. It combines this information with the traffic data and advises him about the best route to take.

Building the design of such a system was a challenging experience as there are numerous aspects to be taken into account. For each feature that we included in the design we also presented a possible manner to achieve it.

Given the huge complexity of the system we chose to implement the most important components with their basic characteristics.

References

1. Kerner, B.S.: The Physics of Traffic. Springer (2004)
2. Wardrop, J.: Some theoretical aspects of road traffic research. In: Proceedings of the Institution of Civil Engineers, Part II, vol. 1(36), pp. 352–362 (1952)
3. Fisk, C.: More paradoxes in the equilibrium assignment problem. Transportation Research Part B: Methodological 13(4), 305–309 (1979)
4. Dantzig, G.B.: Discrete-variable extremum problems. Operations Res. 5, 266–277 (1957)
5. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numer. Math. 1, 269–271 (1959)
6. Matsoukis, E.C.: Road traffic assignment, a review, part 1: non-equilibrium methods. Transportation Planning and Technology 11(1), 69–79 (1986)
7. Merchant, D.K., Nemhauser, G.L.: A model and an algorithm for the dynamic traffic assignment problems. Transportation Science 12(3), 183–199 (1978)
8. Ziliaskopoulos, A.K.: A linear programming model for the single destination system optimum dynamic traffic assignment problem. Transportation Science 34(1), 37–49 (2000)
9. Ho, J.K.: A successive linear optimization approach to the dynamic traffic assignment problem. Transportation Science 14(4), 295–305 (1980)

10. Friesz, T.L., Luque, J., Tobin, R.L., Wie, B.W.: Dynamic network traffic assignment considered as a continuous time optimal control problem. Operations Research 37(6), 893–901 (1989)
11. Vliet, Hall, M.V., Willumsen, D.: Saturn - a simulation assignment model for the evaluation of traffic management schemes. Traffic Engineering and Control 21(4), 168–176 (1980)
12. Taylor, N.B.: Contram dynamic traffic assignment model. Networks and Spatial Economics, 297–322 (1980)
13. Abdelfatah, A., Mahmassani, H.: A simulation-based signal optimization algorithm within a dynamic traffic assignment framework. In: Proceedings of 2001 IEEE Intelligent Transportation Systems, pp. 428–433 (2001)
14. Lum, K.M., Fan, H.S.L., Lam, S.H., Olszewski, P.: Speed-flow modeling of arterial roads in Singapore. Journal of Transportation Engineering 124(3), 213–222 (1998)
15. Kisgy, L., Rilett, L.R.: Travel time prediction by advanced neural network. Periodica Polytechnica Ser. Civ. Eng. 46 (2002)
16. Hobeika, A., Kim, C.K.: Traffic flow-prediction systems based on upstream traffic, pp. 345–350 (1994)
17. Li, Y., McDonald, M.: Link travel time estimation using single GPS-equipped probe vehicle, pp. 932–937 (2002)
18. Ishak, S., Al-Deek, H.: Performance evaluation of short-term time-series traffic prediction model. Journal of Transportation Engineering 128(6), 490–498 (2002)
19. Yu, J., Chang, G.-L., Ho, H., Liu, Y.: Variation based online travel time prediction using clustered neural networks (2008)
20. Lint, J., van Hoogendoorn, S., van Zuylen, H.: Accurate freeway travel time prediction with state-space neural networks under missing data. Transportation Research Part C: Emerging Technologies 13(5-6), 347–369 (2005)

Part III
Signal Processing, Sensors, Systems Modeling and Control

Analysis of a Class of Infinite Dimensional Frames

Cishen Zhang[1], Jingxin Zhang[2], and Xiaofang Chen[3]

[1] Faculty of Engineering and Industrial Sciences
Swinburne University of Technology, Melbourne, VIC 3122, Australia
[2] Department of Electrical and Computer Systems Engineering
Monash University, Clayton, VIC 3800, Australia
[3] School of Electrical Electronic Engineering
Nanyang Technological University, 639798 Singapore

Abstract. Frames are a mathematical tool which can represent redundancies in many application problems. In this article, a class of infinite dimensional and bi-directional frames are studied. It is shown that the infinite dimensional and bi-directional frames can be represented by milti-input, multi-output state space equations. Such a state space representation can enable the application of powerful linear system methods and numerical tools to the performance analysis and evaluation of frames.

Keywords: Frames, Linear systems, Mixed causal-anticausal realizations, State space equations.

1 Introduction

In the study of vector spaces, frame is a more flexible tool compared with basis. The frame elements are linearly dependent, so to provide redundancies, and all elements in the vector space can be written as a linear combination of the frame elements. Frames, as a mathematical theory, were introduced by Duffin and Schaeffer [1] in 1950s. Since 1980s, frames have played an important role in signal processing, see [2] written by Daubechies, Grossman, and Meyer. Some particular classes of frames have been extensively studied, for example, Gabor frames, which are also called Weyl-Heisenberg frames described in [3] and [4], and wavelet frames, which were introduced in [2], [5], and [6]. Frames have also found numerous practical applications including pyramid coding [7], source coding [8], denoising [9], robust transmission [10], CDMA systems, multiantenna code design, image segmentation, classification, restoration and enhancement, etc. More comprehensive reviews of the frames and their applications can be found in [11] and [12].

The theory of frames is a powerful means for the analysis and design of the oversampled uniform filter banks (FBs). [13] and [14] studied properties of oversampled FBs. Necessary and sufficient conditions on a FB to implement a frame or a tight frame in $l^2(\mathbb{Z})$ were given in terms of the properties of the corresponding polyphase analysis matrix. Further frame analysis works were based on the fact that the polyphase matrix provides matrix representation of a frame operator, which can be found in [15], [16],

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 117–128.
springerlink.com

[17], etc. Later, [18] presented a direct computational method for the frame based analysis and design of oversampled FBs, which employed the state space representation of the polyphase matrix.

In the studies of frames, the frame bounds and frame bound ratio are very important indices characterizing the robustness and numerical performance of frame systems. The quantification and computation of frame bounds have been actively investigated in past years. The classic approach to obtain frame bounds of multirate FBs is in the frequency domain, for example, [15], [16], [17], [19]. [20] stated the frame bounds of iterated FBs making use of the wavelet frame bounds computed in the frequency domain. The frequency domain approach to computing the frame bounds is an approximation method which samples the polyphase matrix of the frame operator over the frequency range $\omega \in [0, 2\pi)$ and then performs eigenanalysis on the sampled matrices. Such a sampling approach can be very tedious when the frequency grid is dense and the polyphase matrix is nondiagonal and of infinite impulse response. Moreover, errors due to the frequency domain sampling of the polyphase matrix cannot be precisely quantified and predicted by the density of the frequency grid for generic oversampled FBs.

Frame bounds can also be evaluated in the time domain using the linear matrix inequality (LMI) technique, see [23], which is an application of the KYP lemma stated in [24]. This method avoids the frequency domain sampling and approximation, but is only applicable to causal FB frames in the forward direction rather than to bi-directional frames.

In this article, a time domain direct state space representation and analysis of frames is presented. It is shown that a general class of frames including multidimensional and bi-directional frames with mixed causal-anticausal components can be formulated in the form of multi-input multi-output state space equations. This state space representation can enable the powerful linear system methods in the time domain to be applied to the performance analysis and evaluation of the frames, dual frames and frame bounds. Based on the state space representation, the well known LMI solutions can be applied to provide accurate and efficient numerical computation of the frame performance.

The rest of this article is organized as follows. Section 2 presents notations followed by the fundamentals of frames. Different representations of mixed causal-anticausal LTI systems are introduced, including the state space representation and the transfer function representation in Section 3. Section 4 presents the state space representation and the analysis of frames and frame bound properties. The article is concluded by Section 5.

2 Preliminaries

Let $\ell_2(\mathbb{Z})$ denote the square summable vector space. A sequence of vectors $\{\hat{h}_i;\ \hat{h}_i \in \ell_2,\ i \in \mathbb{Z}\}$, with each $\hat{h}_i$ written as $\hat{h}_i = \{\cdots,\ \hat{h}_i(-1),\ \hat{h}_i(0),\ \hat{h}_i(1),\ \cdots\}$, is a frame for ℓ_2 if there exist constants $0 < \alpha \le \beta < \infty$ such that

$$\alpha \left\|\hat{v}\right\|^2 \le \sum_{i\in\mathbb{Z}} \left|\langle \hat{v}, \hat{h}_i \rangle\right|^2 \le \beta \left\|\hat{v}\right\|^2, \ \ \forall \hat{v} \in \ell_2. \tag{1}$$

The frame $\{\hat{h}_i\}$ is called a tight frame if $\alpha = \beta$ is satisfied.

A sequence $\{\hat{f}_i;\ \hat{f}_i \in \ell_2,\ i \in \mathbb{Z}\}$ is a dual frame of $\{\hat{h}_i\}$ if it satisfies

$$\hat{v} = \sum_{i \in \mathbb{Z}} \langle \hat{v}, \hat{h}_i \rangle \hat{f}_i;\ \forall \hat{v} \in \ell_2. \tag{2}$$

A frame $\{\hat{r}_i\}$ is the canonical dual frame of $\{\hat{h}_i\}$ if it satisfies

$$\sum_{i \in \mathbb{Z}} |\langle \hat{v}, \hat{r}_i \rangle|^2 \leq \sum_{i \in \mathbb{Z}} |\langle \hat{v}, \hat{f}_i \rangle|^2, \tag{3}$$

for all frames $\{\hat{f}_i\}$ which are dual frames of $\{\hat{h}_i\}$.

The frame operator $\hat{\mathcal{S}} :\ \ell_2 \to \ell_2$ of a given frame $\{h_i\}$ is defined as

$$\hat{\mathcal{S}}\hat{v} = \sum_{i \in \mathbb{Z}} \langle \hat{v}, \hat{h}_i \rangle \hat{h}_i, \tag{4}$$

and it is a positive and invertible operator. Applying $\hat{\mathcal{S}}^{-1}$ to both sides of (4) yields

$$\hat{v} = \sum_{i \in \mathbb{Z}} \langle \hat{v}, \hat{h}_i \rangle \hat{\mathcal{S}}^{-1} \hat{h}_i.$$

It shows that $\{\hat{\mathcal{S}}^{-1}\hat{h}_i\}$ is a dual frame of $\{\hat{h}_i\}$. Moreover, it can be verified that $\{\hat{\mathcal{S}}^{-1}\hat{h}_i\}$ is the canonical dual frame of $\{\hat{h}_i\}$.

Define the blocking operator $\mathcal{B}_M : \ell_2 \to \ell_2^M$ as follows.

$$\mathcal{B}_M \hat{v} = v = \{\cdots, v(-1), v(0), v(1), \cdots\} \in \ell_2^M,$$

where

$$v(k) = [v_0(k),\ v_1(k),\ \cdots,\ v_{M-1}(k)]^T,$$

with $v_m(k) = \hat{v}(kM + m)$, $m = 0, 1, \cdots, M-1$. The corresponding deblocking operator is $\mathcal{B}_M^{-1} :\ \ell_2^M \to \ell_2$, such that $\mathcal{B}_M^{-1} v = \hat{v}$. It is clear that $\|v\| = \|\hat{v}\|$ and $\mathcal{B}_M$ and $\mathcal{B}_M^{-1}$ satisfy $\|\mathcal{B}_M\| = \|\mathcal{B}_M^{-1}\| = 1$. For $v, h \in \ell_2^M$, define the inner product between v and h as

$$\langle v, h \rangle = \sum_{k \in \mathbb{Z}} \langle v(k), h(k) \rangle.$$

It follows that $\langle v, h \rangle = \langle \hat{v}, \hat{h} \rangle$.

Applying the blocking operator to the frame $\{\hat{h}_i\}$ yields

$$h_i = \mathcal{B}_M \hat{h}_i;\ i \in \mathbb{Z}. \tag{5}$$

If $\{\hat{h}_i\}$ is a frame satisfying (3), the sequence $\{h_i\}$ is a frame for ℓ_2^M in the sense

$$\alpha \|v\|^2 \leq \sum_{i \in \mathbb{Z}} |\langle v, h_i \rangle|^2 \leq \beta \|v\|^2,\ \ \forall v \in \ell_2^M.$$

Following from the equivalence between $\{\hat{h}_i\}$ and $\{h_i\}$, it is straightforward that $\{h_i\}$ is a tight frame for ℓ_2^M if $\{\hat{h}_i\}$ is a tight frame for ℓ_2; $\{f_i\}$ is a dual frame of $\{h_i\}$ such that

$$v = \sum_{i \in \mathbb{Z}} \langle v, h_i \rangle f_i;\ \forall v \in \ell_2^M, \tag{6}$$

if $\{\hat{f}_i\}$ is a dual frame of $\{\hat{h}_i\}$; and $\{f_i\}$, with $f_i = \mathcal{S}^{-1}h_i$, $i \in \mathbb{Z}$, is a canonical dual frame of $\{h_i\}$ if $\{\hat{f}_i\}$ is a canonical dual frame of $\{\hat{h}_i\}$, where $\mathcal{S} : \ \ell_2^M \to \ell_2^M$ is the frame operator of $\{h_i\}$ given by

$$\mathcal{S}v = \sum_{i\in\mathbb{Z}} \langle v, h_i\rangle h_i. \tag{7}$$

3 State Space Realization of Systems

Consider a multi-input, multi-output dynamical system in terms of the following state equation.

$$x_c(k+1) = A_c x_c(k) + B_c u(k), \tag{8}$$

$$x_a(k-1) = A_a x_a(k) + B_a u(k), \tag{9}$$

$$y(k) = C_c x_c(k) + C_a x_a(k) + Du(k), \tag{10}$$

where (8) and (9) are, respectively, the causal and anticausal state equations, (10) is the output equation, $x_c(k) \in \mathbb{R}^{n_c}$ and $x_a(k) \in \mathbb{R}^{n_a}$ are, respectively, the causal and anticausal state vectors, $u(k) \in \mathbb{R}^M$ is the input vector, $y(k) \in \mathbb{R}^N$ is the output vector, $A_c \in \mathbb{R}^{n_c\times n_c}$, $B_c \in \mathbb{R}^{n_c\times M}$, $C_c \in \mathbb{R}^{N\times n_c}$ and $D_c \in \mathbb{R}^{N\times M}$ are matrices of the causal state equation and $A_a \in \mathbb{R}^{n_a\times n_a}$, $B_a \in \mathbb{R}^{n_a\times M}$ and $C_a \in \mathbb{R}^{N\times n_a}$ are matrices of the anticausal state equation of the system.

The system (8)-(10) is a minimal realization if the matrix dimension of A_a and A_c are minimal among all realizations of the system. The causal system (8) is stable if all the eigenvalues of A_c are within the unit circle and stabilizable if there exists an $F_c \in \mathbb{R}^{M\times N_c}$ such that all the eigenvalues of $A_c + B_c F_c$ are within the unit circle. It is unstable if A_c have one or more eignvalues on or outside the unit circle and is antistable if all the eigenvalues of A_c are outside the unit circle. Similarly, The anticausal system (9) is stable if all the eigenvalues of A_a are within the unit circle and stabilizable if there exists an $F_a \in \mathbb{R}^{M\times N_c}$ such that all the eigenvalues of $A_a + B_a F_a$ are within the unit circle. It is unstable if A_a have one or more eignvalues on or outside the unit circle and is antistable if all the eigenvalues of A_a are outside the unit circle. The anticausal-causal system (8)-(10) is stable, if all the eignvalues of A_c and A_a are within the unit circle, and is unstable otherwise.

Let z denote the shift operator such that $zx(k) = x(k+1)$ and $z^{-1}x(k) = x(k-1)$. The transfer matrix $G_{ac}(z) \in \mathbb{C}^{N\times M}$ of the system (8)-(10) with anticausal and causal states is given by

$$G_{ac}(z) = C_a(z^{-1}I - A_a)^{-1}B_a + D + C_c(zI - A_c)^{-1}B_c. \tag{11}$$

Introduce, respectively, the following short notation for the anticausal, causal and anticausal-causal system state space realizations.

$$\begin{aligned} G_a(z) &= (A_a, B_a, C_a, D)_a \\ &= C_a(z^{-1}I - A_a)^{-1}B_a, \qquad (12) \\ G_c(z) &= (A_c, B_c, C_c, D)_c \end{aligned}$$

$$= C_c(zI - A_c)^{-1}B_c + D, \tag{13}$$

$$\begin{aligned} G_{ac}(z) &= (A_a, B_a, C_a, D, A_c, B_c, C_c) \\ &= C_a(z^{-1}I - A_a)^{-1}B_a + D + C_c(zI - A_c)^{-1}B_c. \end{aligned} \tag{14}$$

The systems $G_a(z)$ and $G_c(z)$ are, respectively, strictly anticausal and causal if $D = 0$.

If the system $G_{ac}(z)$ is stable, it defines an operator $\mathcal{G}_{ac} : \ \ell_2^M \to \ell_2^N$ and the norm of $\mathcal{G}_{ac}$ is defined as

$$\|\mathcal{G}_{ac}\| = \sup_{\|\hat{u}=1\|} \|\mathcal{G}_{ac}\hat{u}\| \tag{15}$$

Let $S_{ac}(z) = G^*_{ac}(z^{-1})G_{ac}(z)$ and $\bar{\lambda}(S_{ac}(e^{j\omega}))$ and $\underline{\lambda}(S_{ac}(e^{j\omega}))$ be the greatest and lest eigenvalues of $S_{ac}(e^{j\omega})$, respectively. It has been well known that $\|\mathcal{G}_{ac}\|$ is the H_∞ norm of $G_{ac}(z)$, i.e.

$$\|\mathcal{G}_{ac}\| = \|G_{ac}\|_\infty = \sup_{\omega\in[0,2\pi)} \bar{\lambda}(S_{ac}(e^{j\omega})) \tag{16}$$

The above result implies that the operator norm of a stable system is the H_∞ norm of the system transfer matrix. It is noted that the H_∞ norm of a transfer matrix exists as long as the transfer matrix is bounded for all $\omega \in [0, 2\pi)$, so the existence of the system H_∞ norm does not necessarily require that the system is stable. It is also noted that state space matrix based standard computational procedures for computing the H_∞ norm of stable and unstable systems have been well known.

The following lemmas are to facilitate the analysis and evaluation of the anticusal-causal systems.

Lemma 1. *If $G_a(z) = (A_a, B_a, C_a, D)_a$ is a stable anticausal realization and A_a is an invertible matrix, it has a symbolically equivalent antistable causal realization*

$$G_{\bar{a}}(z) = (A_a^{-1}, A_a^{-2}B_a, -C_a, D - C_aA_a^{-1}B_a)_c, \tag{17}$$

such that $G_{\bar{a}}(z) = G_a(z)$.

Proof.

$$\begin{aligned} G_a(z) &= C_a(zI - A_a)^{-1}B_a + D \\ &= -C_a(A_a^{-1} - z^{-1}I)^{-1}z^{-1}A_a^{-1}B_a + D \\ &= -C_a(z^{-1}I - A_a^{-1})^{-1}(z^{-1}I - A_a^{-1} + A_a^{-1})A_a^{-1}B_a + D \\ &= -C_a(z^{-1}I - A_a^{-1})^{-1}A_a^{-2}B_a + D - CA_a^{-1}B_a = G_{\bar{a}}(z). \end{aligned}$$

Since all eigenvalues of A_a are strictly within the unit circle, all eigenvalues of A_a^{-1} are strictly outside the unit circle and, hence, $\bar{G}_a(z)$ represents an antistable causal realization. □

Lemma 1 can be immediately extended to obtain the results as stated in the following Lemmas 2-4.

Lemma 2. *If $G_c(z) = (A_c, B_c, C_c, D)_c$ is a stable causal realization and A_c is an invertible matrix, it has a symbolically equivalent antistable anticausal realization*

$$G_{\bar{c}}(z) = (A_c^{-1}, A_c^{-2}B_c, -C_c, D - C_cA_c^{-1}B_c)_a, \tag{18}$$

such that $G_{\bar{c}}(z) = G_c(z)$. □

Lemma 3. *If $G_a(z) = (A_a, B_a, C_a, D)_a$ is a stable anticausal realization and A_a is an invertible matrix, the anticausal-causal realization $G_{ac}(z)$ has a symbolically equivalent unstable causal realization*

$$\begin{aligned} G_{\bar{a}c}(z) &= \left(\begin{bmatrix} A_a^{-1} & 0 \\ 0 & A_c \end{bmatrix}, \begin{bmatrix} A_a^{-2}B_a \\ B_c \end{bmatrix}, \begin{bmatrix} -C_a & C_c \end{bmatrix}, D - C_a A_a^{-1} B_a \right)_c, \\ &= (A_{\bar{a}c}, B_{\bar{a}c}, C_{\bar{a}c}, D_{\bar{a}c})_c, \end{aligned} \tag{19}$$

such that $G_{\bar{a}c}(z) = G_{ac}(z)$. □

Lemma 4. *If $G_c(z) = (A_c, B_c, C_c, D)_c$ is a stable causal realization and A_c is an invertible matrix, the anticausal-causal realization $G_{ac}(z)$ has a symbolically equivalent unstable causal realization*

$$\begin{aligned} G_{a\bar{c}}(z) &= \left(\begin{bmatrix} A_a & 0 \\ 0 & A_c^{-1} \end{bmatrix}, \begin{bmatrix} B_a \\ A_c^{-2}B_c \end{bmatrix}, \begin{bmatrix} C_a & -C_c \end{bmatrix}, D - C_c A_c^{-1} B_c \right)_a, \\ &= (A_{a\bar{c}}, B_{a\bar{c}}, C_{a\bar{c}}, D_{a\bar{c}})_a, \end{aligned} \tag{20}$$

such that $G_{a\bar{c}}(z) = G_{ac}(z)$. □

The results of Lemmas 1-4 show that as long as one of A_a or A_c is an invertible matrix, there is a symbolically equivalent causal realization $G_{\bar{a}c}(z)$ or $G_{a\bar{c}}(z)$ of the anticausal-causal realization $G_{ac}(z)$. It is noted that $G_{\bar{a}c}(z)$ or $G_{a\bar{c}}(z)$ is an unstable system and, hence, does not represent a bounded operator on ℓ_2^M. However, its H_∞ norm, following from the definition in the form (16) exists if $G_{\bar{a}c}(e^{j\omega})$ or $G_{a\bar{c}}(e^{j\omega})$ is analytic over all $\omega \in [0, 2\pi)$. These lead to the following result on the H_∞ norm of $G_{ac}(z)$.

Theorem 1. *If A_a or A_c is an invertible matrix such that $G_{\bar{a}c}(z)$ in (19) or $G_{a\bar{c}}(z)$ in (20) exists, then*

$$\|G_{ac}\|_\infty = \|G_{\bar{a}c}\|_\infty, \tag{21}$$

or

$$\|G_{ac}\|_\infty = \|G_{a\bar{c}}\|_\infty. \tag{22}$$

Proof. For the stable matrix A_a and the corresponding antistable matrix A_a^{-1}, the matrices $(e^{j\omega}I - A_a)^{-1} \in \mathbb{C}^{n_a \times n_a}$ and $(e^{-j\omega}I - A_a^{-1})^{-1} \in \mathbb{C}^{n_a \times n_a}$ and, hence, $G_a(e^{j\omega}) \in \mathbb{C}^{N \times M}$ in (12) and $G_{\bar{a}}(e^{j\omega}) \in \mathbb{C}^{N \times M}$ in (17) are analytic over all $\omega \in [0, 2\pi)$. It follows from (17) and (19) that

$$G_a(e^{j\omega}) = G_{\bar{a}}(e^{j\omega}), \quad G_{ac}(e^{j\omega}) = G_{\bar{a}c}(e^{j\omega}), \quad \forall \omega \in [0, 2\pi).$$

Introduce $S_{\bar{a}c}(z) = G_{\bar{a}c}^*(z^{-1})G_{\bar{a}c}(z)$. It satisfies

$$S_{\bar{a}c}(e^{j\omega}) = G_{ac}^*(e^{-j\omega})G_{ac}(e^{j\omega}) = S_{ac}(e^{j\omega}), \ \forall \omega \in [0, 2\pi).$$

Hence,

$$\|G_{ac}\| = \sup_{\omega \in [0,2\pi)} \bar{\lambda}(S_{ac}(e^{j\omega})) = \sup_{\omega \in [0,2\pi)} \bar{\lambda}(S_{\bar{a}c}(e^{j\omega})) = \|G_{\bar{a}c}\|. \tag{23}$$

The result of $\|G_{ac}\|_\infty = \|G_{a\bar{c}}\|_\infty$ can be obtained similarly. □

While the existing methods for evaluation of the system H_∞ norm are only available for causal or anticausal but not for combined anticausal-causal systems, Theorem 1 provides a very convenient means for the H_∞ norm evaluation of anticausal-causal systems based on the norm equivalence between the different systems.

We now consider to factorize $G_{\bar{a}c}(z)$ and $G_{a\bar{c}}(z)$ and to derive their inverse systems. A transfer matrix $U(z) \in \mathbb{C}^{N\times M}$, with $N \geq M$, is called inner if it satisfies $U^*(e^{-j\omega})U(e^{j\omega}) = I$, for all $\omega \in [0, 2\pi)$. A well known result on factorizing an inner transfer matrix of $G_{\bar{a}c}(z)$ is

Lemma 5. *If the causal system realization (19) is stabilizable, $\begin{bmatrix} A_{\bar{a}c} - e^{j\omega}I & B_{\bar{a}c} \\ C_{\bar{a}c} & D_{\bar{a}c} \end{bmatrix}$ has full column rank for all $\omega \in [0, 2\pi)$ and $D_{\bar{a}c}$ has full column rank, then $G_{\bar{a}c}(z)$ can be factorized into the following form*

$$G_{\bar{a}c}(z) = U_{\bar{a}c}(z)V_{\bar{a}c}^{-1}(z), \tag{24}$$

where $U_{\bar{a}c}(z)$ is inner, $U_{\bar{a}c}(z)$ and $V_{\bar{a}c}(z)$ are stable causal transfer matrices satisfying

$$U_{\bar{a}c}(z) = (A_{\bar{a}c} + B_{\bar{a}c}F_{\bar{a}c}, B_{\bar{a}c}W_{\bar{a}c}^{1/2}, C_{\bar{a}c} + D_{\bar{a}c}F_{\bar{a}c}, D_{\bar{a}c}W_{\bar{a}c}^{1/2})_c \in \mathbb{C}^{N\times M}, \tag{25}$$

$$V_{\bar{a}c}(z) = (A_{\bar{a}c} + B_{\bar{a}c}F_{\bar{a}c}, B_{\bar{a}c}W_{\bar{a}c}^{1/2}, F_{\bar{a}c}, W_{\bar{a}c}^{1/2})_c \in \mathbb{C}^{M\times M}, \tag{26}$$

$F_{\bar{a}c}$ and $W_{\bar{a}c}$ are given by

$$W_{\bar{a}c} = W_{\bar{a}c}^* = D_{\bar{a}c}^*D_{\bar{a}c} + B_{\bar{a}c}^*X_{\bar{a}c}B_{\bar{a}c},$$

$$F_{\bar{a}c} = -W_{\bar{a}c}^{-1}(B_{\bar{a}c}^*X_{\bar{a}c}A_{\bar{a}c} + D_{\bar{a}c}^*C_{\bar{a}c}),$$

and $X_{\bar{a}c} = X_{\bar{a}c}^* \geq 0$ is the unique solution to the following algebraic Riccati equation

$$A_{\bar{a}c}^*X_{\bar{a}c}A_{\bar{a}c} - X_{\bar{a}c} + C_{\bar{a}c}^*C_{\bar{a}c} - (A_{\bar{a}c}^*X_{\bar{a}c}B_{\bar{a}c} + C_{\bar{a}c}^*D_{\bar{a}c})W_{\bar{a}c}^{-1}(B_{\bar{a}c}^*X_{\bar{a}c}A_{\bar{a}c} + D_{\bar{a}c}^*C_{\bar{a}c}) = 0.$$

□

With the factorization of $G_{\bar{a}c}(z)$ given in (24), it is straightforward to verify that

$$R_{\bar{a}c}(z) = (G_{\bar{a}c}^*(z^{-1})G_{\bar{a}c}(z))^{-1}G_{\bar{a}c}^*(z^{-1}) = V_{\bar{a}c}(z)U_{\bar{a}c}^*(z^{-1}) \tag{27}$$

is an inverse system of $G_{\bar{a}c}^*(z)$ such that

$$R_{\bar{a}c}(z)G_{\bar{a}c}(z) = I.$$

Since $U_{\bar{a}c}(z)$ is a stable causal system in the form (25), $U_{\bar{a}c}^*(z^{-1})$ is a stable anticausal system which can be written as

$$\begin{aligned} U_{\bar{a}c}^*(z^{-1}) &= -(C_{\bar{a}c}^* + F_{\bar{a}c}^*D_{\bar{a}c}^*)(z^{-1}I - A_{\bar{a}c}^* - F_{\bar{a}c}^*B_{\bar{a}c}^*)^{-1}W_{\bar{a}c}^{*1/2}B_{\bar{a}c}^* + W_{\bar{a}c}^{*1/2}D_{\bar{a}c}^* \\ &= (A_{\bar{a}c}^* + F_{\bar{a}c}^*B_{\bar{a}c}^*,\ W_{\bar{a}c}^{*1/2}B_{\bar{a}c}^*,\ -C_{\bar{a}c}^* - F_{\bar{a}c}^*D_{\bar{a}c}^*,\ W_{\bar{a}c}^{*1/2}D_{\bar{a}c}^*)_a. \end{aligned} \tag{28}$$

This shows that $R_{\bar{a}c}(z)$ in (27) as an inverse system of $G_{\bar{a}c}(z)$ is the cascade of the stable anticausal system $U_{ac}(z) = U^*_{\bar{a}c}(z^{-1})$ and the stable causal system $V_{\bar{a}c}(z)$. By using the result of [22], $R_{\bar{a}c}(z)$ can be written into a summation of stable anticausal and causal systems as follows

$$R_{\bar{a}c}(z) = R_a(z) + R_c(z) \in \mathbb{C}^{N\times M}, \tag{29}$$

where

$$R_a(z) = (A_{ra}, B_{ra}, C_{ra}, 0)_a, \tag{30}$$
$$R_c(z) = (A_{rc}, B_{rc}, C_{rc}, D_r)_c, \tag{31}$$

are the stable anticasal and causal realizations, respectively, with the state space matrices

$$\begin{aligned}
A_{ra} &= A^*_{\bar{a}c} + F^*_{\bar{a}c}B^*_{\bar{a}c}\\
B_{ra} &= W^{*1/2}_{\bar{a}c}B^*_{\bar{a}c}\\
C_{ra} &= -W^{1/2}_{\bar{a}c}(C^*_{\bar{a}c} + F^*_{\bar{a}c}D^*_{\bar{a}c}) + F_{\bar{a}c}Y(A^*_{\bar{a}c} + F^*_{\bar{a}c}B^*_{\bar{a}c})\\
A_{rc} &= A_{\bar{a}c} + B_{\bar{a}c}F_{\bar{a}c},\\
B_{rc} &= (A_{\bar{a}c} + B_{\bar{a}c}F_{\bar{a}c})YW^{*1/2}_{\bar{a}c}B^*_{\bar{a}c} + B_{\bar{a}c}W^{1/2}_{\bar{a}c}W^{*1/2}_{\bar{a}c}D^*_{\bar{a}c},\\
C_{rc} &= F_{\bar{a}c},\\
D_r &= W^{1/2}_{\bar{a}c}W^{*1/2}_{\bar{a}c}D^*_{\bar{a}c} + F_{\bar{a}c}YW^{*1/2}_{\bar{a}c}B^*_{\bar{a}c},
\end{aligned}$$

The matrix Y in the above equations is the solution of the following Sylvester equation

$$(A_{\bar{a}c} + B_{\bar{a}c}F_{\bar{a}c})Y(A^*_{\bar{a}c} + F^*_{\bar{a}c}B^*_{\bar{a}c}) - Y - B_{\bar{a}c}W^{1/2}_{\bar{a}c}(C^*_{\bar{a}c} + F^*_{\bar{a}c}D^*_{\bar{a}c}) = 0.$$

By Lemma 3 on the equivalence between $G_{\bar{a}c}(z)$ and $G_{ac}(z)$, $R_{\bar{a}c}(z)$ as an inverse system of $G_{\bar{a}c}(z)$ is also an inverse system of $G_{ac}(z)$. This immediately yields the following result.

Theorem 2. *If the anticausal and causal system realization (11) and its associated state matrices in (19) satisfy that A_a is invertible, (A_a^{-1}, B_a) is stabilizable, $D_{\bar{a}c}$ has full column rank and $\begin{bmatrix} A_{\bar{a}c} - e^{j\omega}I & B_{\bar{a}c} \\ C_{\bar{a}c} & D_{\bar{a}c}\end{bmatrix}$ has full column rank for all $\omega \in [0, 2\pi)$, then it has a stable anticausal-causal inverse system*

$$R_{ac}(z) = R_a(z) + R_c(z), \tag{32}$$

such that $R_{ac}(z)G_{ac}(z) = I$, where $R_{ac}(z) = R_{\bar{a}c}(z)$, $R_a(z)$ and $R_c(z)$ are given in (29)-(31). □

4 State Space Representation and Analysis of Frames

The power series expansion of the anticausal-causal transfer matrix $G_{ac}(z)$ determines a bi-directional sequence of matrices $\{g(k): \ g(k) \in \mathbb{R}^{N\times M}, k \in \mathbb{Z}\}$ with

$$g(k) = \begin{cases} C_a A_a^{-k-1} B_a, \ \forall k < 0, \\ D, \qquad\qquad\quad k = 0, \\ C_c A_c^{k-1} B_c, \quad \forall k > 0. \end{cases} \tag{33}$$

Each $g(k) \in \mathbb{R}^{N\times M}$ can be partitioned as

$$g(k) = \begin{bmatrix} g_0^T(k) \\ g_1^T(k) \\ \vdots \\ g_{N-1}^T(k) \end{bmatrix} = \begin{bmatrix} g_{0,0}(k) & g_{0,1}(k) & \cdots & g_{0,M-1}(k) \\ g_{1,0}(k) & g_{1,1}(k) & \cdots & g_{1,M-1}(k) \\ \vdots & \vdots & \vdots & \vdots \\ g_{N-1,0}(k) & g_{N-1,1}(k) & \cdots & g_{N-1,M-1}(k) \end{bmatrix}, \tag{34}$$

where

$$g_{j,m}(k) = \begin{cases} C_{a,j} A_a^{-k-1} B_{a,m} \ \forall k < 0, \\ D_{j,m}, \qquad\qquad\quad k = 0, \\ C_{c,j} A_c^{k-1} B_{c,m} \quad \forall k > 0, \end{cases} \tag{35}$$

for $j = 0, 1, \cdots, N-1$, $m = 0, 1, \cdots, M-1$, $C_{a,j} \in \mathbb{R}^{1\times n_a}$ and $C_{c,j} \in \mathbb{R}^{1\times n_c}$ are, respectively, the jth row of C_a and C_c, $B_{a,j} \in \mathbb{R}^{n_a\times 1}$ and $B_{c,j} \in \mathbb{R}^{n_c\times 1}$ are, respectively, the mth column of B_a and B_c, and $D_{j,m}$ is the entry of D at jth row and mth column.

In fact, $\{g(k)\}$ represents the natural response of the system (8)-(10). If the system (8)-(10) is stable, then $g_j \in \ell_2^M$, for $j = 0, 1, \cdots, N-1$. For a given input $u \in \ell_2^M$, in the blocked sequence form, to the system (8)-(10), the stable system output $y \in \ell_2^N$, is also in the blocked sequence form and determined by the following convolution.

$$y = g * u = \begin{bmatrix} g_0^T \\ g_1^T \\ \vdots \\ g_{N-1}^T \end{bmatrix} * u. \tag{36}$$

Equation (36) is equivalent to

$$y(l) = \sum_{k\in\mathbb{Z}} \begin{bmatrix} g_0^T(l-k) \\ g_1^T(l-k) \\ \vdots \\ g_{N-l}^T(l-k) \end{bmatrix} u(k) = \sum_{k\in\mathbb{Z}} \begin{bmatrix} \langle g_0(l-k), u(k)\rangle \\ \langle g_1(l-k), u(k)\rangle \\ \vdots \\ \langle g_{N-l}(l-k), u(k)\rangle \end{bmatrix}. \tag{37}$$

Denote $g_j(l-k)$ by $h_{lN+j}(k)$ for $l \in \mathbb{Z}$ and $j = 0, 1, \cdots, N-1$. Equation (37) can be written as

$$y(l) = \sum_{k\in\mathbb{Z}} \begin{bmatrix} \langle u(k), h_{lN}(k)\rangle \\ \langle u(k), h_{lN+1}(k)\rangle \\ \vdots \\ \langle u(k), \ h_{lN+N-1}(k)\rangle \end{bmatrix} = \begin{bmatrix} \langle u, h_{lN}\rangle \\ \langle u, h_{lN+1}\rangle \\ \vdots \\ \langle u, h_{lN+N-1}\rangle \end{bmatrix}. \tag{38}$$

As a result, the system output satisfies

$$\|y\|^2 = \sum_{i\in\mathbb{Z}} |\langle u, h_i\rangle|^2. \tag{39}$$

where i denotes $lN+j$ for $l \in \mathbb{Z}$ and $j = 0, 1, \cdots, N-1$. It follows from (35) that

$$h_i^T(k) = h_{lN+j}^T(k) = g_j^T(l-k) = \begin{cases} C_{a,j}A_a^{k-l-1}B_a \ \forall (k-l) > 0, \\ D_j, \qquad\qquad\qquad k-l=0, \\ C_{c,j}A_c^{l-k-1}B_c \ \forall (k-l) < 0, \end{cases} \tag{40}$$

for $l \in \mathbb{Z}$ and $j = 0, 1, \cdots, N-1$, where $D_j \in \mathbb{R}^{1\times M}$ is the jth row of D. If (8)- (10) is a stable system, then $g_j \in \ell_2^M$, $j = 0, 1, \cdots, N-1$, and consequently $h_i = \{\cdots, h_i(-1), h_i(0), h_i(1), \cdots\} \in \ell_2^M$. Thus the stable system (8)-(10) determines a set of ℓ_2^M sequences $\{h_i\}$ in term of the system matrices which can possibly be a frame for ℓ_2^M. In the following, we study conditions for $\{h_i\}$ being a frame for ℓ_2^M and performance and properties of the frames represented by state space equations.

Theorem 3. *The sequence set $\{h_i\}$, $\forall i \in \mathbb{Z}$, as defined by the state matrices of the anticausal-causal system $G_{ac}(z)$ in (11) represents a frame for ℓ_2^M if and only if it is stable and $G_{ac}(e^{j\omega})$ has full column rank for all $\omega \in [0, 2\pi)$.*

Proof. The stability of $G_{ac}(z)$ implies that, for any input sequence $u \in \ell_2^M$, the system output satisfies $y \in \ell_2^N$ and there exists a constant β such that $\|y\|^2 \leq \beta\|u\|^2$. It then follows from (39) that

$$\sum_{i\in\mathbb{Z}} |\langle u, h_i\rangle|^2 \leq \beta\|u\|^2, \ \ \forall u \in \ell_2^M. \tag{41}$$

On the other hand, the full column rank condition for $G_{ac}(e^{j\omega})$ implies that the null space of $\mathcal{G}_{ac}$ is empty. Thus there exists a constant α such that the output y satisfies

$$\|y\|^2 = \sum_{i\in\mathbb{Z}} |\langle u, h_i\rangle|^2 \geq \alpha\|u\|^2, \ \ \forall u \in \ell_2^M. \tag{42}$$

This together with (41) shows that the sequence set $\{h_i\}$ defined by $G_{ac}(z)$ satisfies the definition of the frame. □

Theorem 4. *If $\{h_i\}$, $\forall i \in \mathbb{Z}$, is a frame for ℓ_2^M defined by the state matrices of the anticausal-causal system $G_{ac}(z)$ in (11) and $G_{ac}(z)$ satisfies the system factorization conditions in Lemma 5, then the tightest upper and lower bounds, β and α, of the frame $\{h_i\}$ are given by*

$$\beta = \|G_{ac}\|_\infty^2 = \|G_{\bar{a}c}\|_\infty^2, \ \ \alpha = \frac{1}{\|R_{ac}\|_\infty^2} = \frac{1}{\|R_{\bar{a}c}\|_\infty^2} = \frac{1}{\|V_{\bar{a}c}\|_\infty^2}. \tag{43}$$

Proof. It follows from

$$\|y\|^2 = \sum_{i\in\mathbb{Z}} |\langle u, h_i\rangle|^2 = \|\mathcal{G}_{ac}u\|^2$$

that

$$\begin{aligned}\beta &= \sup_{\omega\in[0,2\pi)} \bar{\lambda}(G_{ac}^{*}(e^{-j\omega})G_{ac}(e^{j\omega})) \\ &= \|G_{ac}\|_{\infty}^{2} = \|G_{\bar{a}c}\|_{\infty}^{2},\end{aligned}$$

$$\begin{aligned}\alpha &= \inf_{\omega\in[0,2\pi)} \underline{\lambda}(G_{ac}^{*}(e^{-j\omega})G_{ac}(e^{j\omega})) \\ &= \frac{1}{\sup_{\omega\in[0,2\pi)} \bar{\lambda}(G_{ac}^{-1}(e^{j\omega})G_{ac}^{-*}(e^{-j\omega}))} \\ &= \frac{1}{\|R_{ac}\|_{\infty}^{2}} = \frac{1}{\|R_{\bar{a}c}\|_{\infty}^{2}} = \frac{1}{\|V_{\bar{a}c}\|_{\infty}^{2}}.\end{aligned}$$

□

Theorem 4 shows that the tightest upper and lower bounds of the frame can be evaluated by computing the H_{∞} norm of its state space realization. Since there has been well developed software based on LMI technique for accurate and efficient computation of the H_{∞} norm of state space realizations, the result of Theorem 4 can provide a means for the analysis and evaluation of the frame bounds using the state space realization of frames.

5 Conclusions

A class of frames, with elements in the form of bi-directional infinite impulse responses of a linear time invariant (LTI) system, can be equivalently modeled as mixed causal-anticausal systems. This paper presents a direct state space approach to the analysis and computation of optimal frame bounds of this class of frames. It is shown that the lower frame bound is also equal to the inverse of the square of the operator norm of the left inverse system which can achieve perfect reconstruction.

The results obtained in this paper are applicable to a class of frames which are governed by exponential type performance behavior and can be modeled by LTI system responses in the time and frequency domains. Currently, the authors are extending the LTI state space approach presented in this paper to linear time varying (LTV) state space modeling, analysis and computation of frames. This study will enable deeper understanding and more efficient evaluation of a larger general class of frames which may not be properly analyzed and evaluated in the conventional frequency domain.

References

1. Duffin, R.J., Schaeffer, A.C.: A Class of Nonharmonic Fourier Series. Trans. of the American Mathematical Society 72(2), 341–366 (1952)
2. Daubechies, I., Grossman, A., Meyer, Y.: Painless Nonorthogonal Expansions. Journal of Mathematical Phyysics 27(5), 1271–1283 (1986)
3. Heil, C., Walnut, D.: Continuous and Discrete Wavelet Transforms. SIAM Review 31(4), 628–666 (1989)

4. Peter, G.C.: Modern Tools for Weyl-Heisenberg (Gabor) Frame Theory. Advances in Imaging and Electron Physics 115, 1–127 (2001)
5. Daubechies, I.: The Wavelet Transform, Time-frequency Localization and Signal Analysis. IEEE Trans. on Information Theory 36(5), 961–1005 (1990)
6. Daubechies, I.: Ten Lectures on Wavelets. Society for Industrial and Applied Mathematics (1992)
7. Peter, J.B., Edward, H.A.: The Lapacian Pyramid as a Compact Image Code. IEEE Trans. on Communications 31(4), 532–540 (1983)
8. Benedetto, J.J., Powell, A.M., Yilmaz, Ö.: Sigma-delta Quantization and Finite Frames. IEEE Trans. on Information Theory 52(5), 1990–2005 (2006)
9. Dragotti, P.L., Velisavljevic, V., Vetterli, M., Beferull-Lozano, B.: Discrete Directinoal Wavelet Bases and Frames for Image Compression and Denoising. In: Proc.SPIE Conf. Wavelet Applications Signal Image Processing, pp. 1287–1295 (2003)
10. Riccardo, B., Roberto, R.: Bounds on Error Amplification in Oversampled Filter Banks for Robust Transmission. IEEE Trans. on Signal Processing 54(4), 1399–1411 (2006)
11. Kovacevic, J., Chebira, A.: Life beyond bases: The advent of frames (Part I). IEEE Signal Processing Mag. 24(4), 86–104 (2007)
12. Kovacevic, J., Chebira, A.: Life beyond bases: The advent of frames (Part II). IEEE Signal Processing Mag. 24(5), 115–125 (2007)
13. Martin, V., Zoran, C.: Oversampled FIR Filter Banks and Frames in $l^2(Z)$. In: IEEE Interational Conference on Acoustic, Speech and Signal Processing Conference Proceddings, pp. 1530–1533 (1996)
14. Zoran, C., Martin, V.: Oversampled Filter Banks. IEEE Trans. on Signal Processing 46(5), 1245–1255 (1998)
15. Helmut, B., Franz, H., Hans, G.F.: Frame-theoretic Analysis of Oversampled Filter Banks. IEEE Trans. on Signal Processing 46(12), 3256–3268 (1998)
16. Helmut, B., Franz, H.: Noise Reduction in Oversampled Filter Banks Using Predictive Quantization. IEEE Trans. on Information Theory 47(1), 155–172 (2001)
17. Alfred, M.: Frame Analysis for Biothogonal Cosine-modulated Filterbanks. IEEE Trans. on Signal Processing 51(1), 172–181 (2003)
18. Li, C., Jingxin, Z., Cishen, Z.: Frame-Theory-Based Analysis and Design of Oversampled Filter Banks: Direct Computational Method. IEEE Trans. on Signal Processing 55(2), 507–519 (2007)
19. David, S., Yehoshua, Y.Z.: Frame Analysis of Wavelet-type Filter Banks. Signal Processing 67(2), 125–139 (1998)
20. Ilker, B., Ivan, W.S.: On the Frame Bounds of Iterated Filter Banks. Applied and Computational Harmonic Analysis 27(2), 255–262 (2009)
21. Christensen, O.: An Introduction to Frames and Riesz Bases. Birkhauser (2003)
22. Huang, S., Tongwen, C.: On Causality and Anticausality of Cascaded Discrete-time Systems. IEEE Trans. on Circuit and Systems I: Fundamental Theory and Applications 43(3), 240–242 (1996)
23. Li, C., Jingxin, Z., Cishen, Z., Edoardo, M.: Efficient Computation of Frame Bounds Using LMI-Based Optimization. IEEE Trans. on Signal Processing 56(7), 3029–3033 (2008)
24. Anders, R.: On the Kalman-Yakubovich-Popov Lemma. Systems and Control Letters 28(1), 7–10 (1996)

Using Chernoff's Bounding Method for High-Performance Structural Break Detection and Forecast Error Reduction

Dirk Pauli[1,*], Yann Lorion[1], Sebastian Feller[2], Benjamin Rupp[2], and Ingo J. Timm[3]

[1] Information Systems and Simulation, Institute of Computer Science,
Goethe-University Frankfurt, P.O. Box 11 19 32, 60054 Frankfurt/Main, Germany
[2] FCE Frankfurt Consulting Engineers GmbH,
Frankfurter Str.5, 65239 Hochheim/Main, Germany
[3] Department IV, Institute of Business Information Systems, Business Informatics I,
University of Trier, 54286 Trier, Germany

Abstract. In this paper, a new method for detecting multiple structural breaks, i.e. undesired changes of signal behavior, is presented and applied to artificial and real-world data. It will be shown how Chernoff Bounds can be used for high-performance change-point detection after preprocessing arbitrary time series to binary random variables using adequate transformation routines. The algorithm is evaluated on artificial time series and compared to state of the art methods. The developed algorithm is competitive to state of the art methods in terms of classification errors but is considerably faster especially when dealing with long time series. Theoretical results on artificial data from part one of this paper are applied to real-world time series from a pharmaceutical wholesaler and show striking improvement in terms of forecast error reduction, thereby greatly improving forecast quality. In order to test the effect of structural break detection on forecast quality, state of the art forecast algorithms are applied to time series with and without previous application of structural break detection methods.

Keywords: Change-point detection, Hypothesis testing, Chernoff Inequality, Binomial distribution, Additive changes, Nonadditive changes, Multiple structural break detection.

1 Introduction

Structural break detection concentrates on discovering time points at which properties of time series change significantly. This term is used e.g. in [1] but other terms like change-point, event, novelty, anomaly or abnormality detection, e.g. in [2], [3], [4], [5], and [6], refer to this problem in a similar manner. The problem itself varies with its application. For example, consider a time series with a stable trend. Despite changes in statistical moments of the distribution, this results in no loss of forecast quality if the data represents the historical demand of an article and the task is to forecast future demands. In contrast, if the same time series is a vibration signal of a gas turbine, a stable

[*] Correspondence: dirk.pauli@ymail.com

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 129–148.
springerlink.com

trend leads to an undesired state of the machine and must be detected as soon as possible, compare [7]. Further real-world applications include e.g. fraud detection in [8], anomaly detection for spacecraft in [9] or [10], detecting abnormal driving conditions in [11], and anomaly detection in multi-node computer systems in [12] to name but a few. More application domains and examples are provided in [13]. All these applications emphasize the importance and need of algorithms for change-point detection for a broad community.

Another real-world application is the forecast of future demands, which is a crucial element of calculating an optimal stock policy. In many cases, large amounts of data are available, but information cannot be retrieved completely, due to limited resources in terms of e.g. computing time or inefficient algorithms. In order to gain the full information available an automated, reliable, and efficient work flow has to be established.

In this paper, a novel approach to structural break detection is introduced in order to reduce forecast errors and thereby increase accuracy and reliability of forecasts. The algorithm is validated on a real-world data set consisting of 8002 independent time series of historical demands of articles of a pharmaceutical wholesaler. The performance of the new algorithm is measured in terms of forecast error reduction, statistical power, significance and runtime on this particular data set.

Related work, compare [14], shows that a common approach in the area of change-point detection is to divide the task at least into two parts: the first step generates residuals of the original measurements that reflect the changes of interest, e.g. the residuals are close to zero before and nonzero after the change. The second step contains the design of a decision rule based upon these residuals. The algorithm presented in this paper proceeds in a similar way. The first task is to transform an arbitrary time series $x_1, \ldots, x_s \in \mathbb{R}$ to a sequence of binary numbers, which is interpreted as the outcome of a binary stochastic process $\{Y_i\}_{i \in \mathbb{N}}$ with $\Omega := \{0, 1\}$. Afterwards, Chernoff Inequalities are used for hypothesis testing, i.e. to estimate the likelihood of structural breaks.

In the context of this paper, the new algorithm is adjusted to detect additive changes. However, the novel approach can be adapted to detect nonadditive changes as well, which is discussed in section 2.5. In e.g. [14] additive changes are defined as shifts in the mean value of a signal, while nonadditive changes are defined as changes e.g. in variance, correlations, spectral characteristics, or dynamics of the signal or system. Both definitions will be used throughout this paper. Furthermore, this paper concentrates on offline detection, i.e. it uses time series as *a posteriori* information, since it is sufficient for the current applications. An online variant of this algorithm will be discussed in section 5.

The paper is structured as follows: section 2.1 presents how Chernoff's Inequalities can be used for high-performance hypothesis testing to detect structural breaks. Section 2.2 provides the design of a transformation routine that fulfils the goal of the case study and reflects additive changes. Section 2.3 introduces boolean operations on hypothesis tests, a technique used to reduce the number of false alarms. In section 2.4 the basic algorithm is extended to multiple structural break detection. In order to show the flexibility of the novel approach, the detection of nonadditive changes is discussed in section 2.5. In section 3, model parameters are analyzed and a benchmark is performed on artificial data, where state of the art methods are compared regarding performance and diagnostic capabilities. Section 4 contains the application of the new algorithm to a real-world problem. Since forecast error reduction will be used as a key performance indicator of the new algorithm, a set of forecast methods is shortly introduced in section 4.1. Test scenarios and error estimates are defined in section 4.2. The results of the case study and performance indicators of the algorithm are presented in section 4.3. In section 5 results of this paper are discussed and potential future enhancements are suggested.

2 A Novel Approach to High-Performance Structural Break Detection

The algorithm used in this paper can be separated into two parts. The first step is to generate random variables $y_i \in \{0, 1\}$ for all $i \in [1, s]$ from the corresponding x_i in order to satisfy the requirements of the variant of Chernoff's bounding method used here. The second step is to prepare and to perform a hypothesis test. The authors of this paper decided to start with step two for reasons of clarity, therefore it will be assumed until section 2.2 that a routine $P : \mathbb{R} \to \{0, 1\}$ does exist to transform x_i adequately.

2.1 Chernoff's Bounding Method for Hypothesis Testing

In this section the application of Chernoff's bounding method to detect structural breaks in time series y_i is presented. First Chernoff's Inequality is described.

Chernoff's Inequality. Given s independent Bernoulli-experiments $y_1, \ldots, y_s$ with probability $Pr\left[y_i = 1\right] = p$ and $Pr\left[y_i = 0\right] = 1 - p$, then for each $\alpha > 0$

$$Pr\left[\sum_{i=1}^{s} y_i \geq (1 + \alpha) \cdot p \cdot s\right] \leq e^{-\frac{\alpha^2 \cdot p \cdot s}{3}} \tag{1}$$

and for each $\alpha \in [0, 1]$

$$Pr\left[\sum_{i=1}^{s} y_i \leq (1 - \alpha) \cdot p \cdot s\right] \leq e^{-\frac{\alpha^2 \cdot p \cdot s}{2}} \tag{2}$$

holds, compare [15].

In other words, large linear deviations from the expectation are highly improbable. Starting at this point, a hypothesis test can be defined as follows: it is assumed that all y_i are independent and identically distributed, therefore the sum of events y_i will only exceed each bound with probability less than γ, where

$$\gamma = e^{-\frac{\alpha^2 \cdot p \cdot s}{c}} \tag{3}$$

and $c \in \{2, 3\}$. If bounds are exceeded, the assumption is considered to be wrong and the hypothesis is rejected. The probability γ is antiproportional to the risk of making a wrong decision.

In fact, the assumption of independent variables has to be discussed, since dependence between latest and preceding samples is inherent to most real-world time series. Consider, the independence property is unique, while the dependence property is it not. The structure of dependence can be e.g. correlation, mutual information, short-range, or long-range. Any form of seasonality shows dependencies between latest and preceding samples. Some forecast models like e.g. ARIMA even rely on dependency structures. But in general the first step for change-point detection is to generate residuals, compare [14], which reflect the change of interest, and are equal to noise of underlying models, which is in general assumed to be independent and identically distributed.

However, transformation routines defined exemplarily in section 2.2 and 2.3 generate residuals and interpret or transform these residuals to binary random variables. Therefor and in the context of this paper the required assumption of independent and identically distributed random variables is considered to be fulfilled. The robustness and limitations of the algorithm in case of dependent variables will be extended in future work.

The central idea of this paper is to perform hypothesis tests for each continuous subsequence of length τ and verify whether the occurrence of events $y_i = 1$ notably differ from their expectation. If they do, the distribution of y_i has changed or differs between certain subsequences and a structural break is considered. As the distribution of 1's and 0's is assumed to be binomial, p can be estimated as follows:

$$A^0 = \{j | j \in \{1, \ldots, s\}, y_j = 0\} \tag{4}$$

$$A^1 = \{j | j \in \{1, \ldots, s\}, y_j = 1\} \tag{5}$$

and $r^0 := \frac{|A^0|}{s}$ and $r^1 := \frac{|A^1|}{s}$ lead to $p := r^1$.

The upper and lower bounds are dependent on α_u and α_l, which can be estimated for a given $\gamma \in (0, 1)$ as follows:

$$\gamma = e^{-\frac{\alpha^2 \cdot p \cdot \tau}{3}} \Leftrightarrow \alpha_u = \sqrt{-\frac{3 \cdot ln(\gamma)}{\tau \cdot p}} \tag{6}$$

$$\gamma = e^{-\frac{\alpha^2 \cdot p \cdot \tau}{2}} \Leftrightarrow \alpha_l = \sqrt{-\frac{2 \cdot ln(\gamma)}{\tau \cdot p}} \tag{7}$$

The next step is to test $\forall i \in [1, \ldots, s - \tau + 1]$, whether the distribution of $y_i, \ldots, y_{i+\tau-1}$ is likely using Chernoff bounds for an estimated p. In other words, it is checked if the sum over $y_i, \ldots, y_{i+\tau-1}$ deviates from its expectation by more than a factor of $1 + \alpha$ or $1 - \alpha$, respectively. Such a deviation of the sum from its expectation can only happen with a probability less than or equal to γ. As gamma is small, deviation leads to the hypothesis being rejected, and a structural break is assumed. If the 1's are uniformly distributed according to p, the hypothesis will hold with probability $1 - \gamma$.

Hypothesis H_i $\forall i \in [1, \ldots, s - \tau + 1]$ is tested and set as follows:

$$H_i = \begin{cases} \text{reject} & \text{if } (S \geq (1 + \alpha_u) \cdot p \cdot \tau) \\ & \quad \vee (S \leq (1 - \alpha_l) \cdot p \cdot \tau) \\ & \\ \text{accept else} & \end{cases} \tag{8}$$

with $S := \sum_{j=i}^{i+\tau-1} y_j$.

If at least for one sequence $y_i, \ldots, y_{i+\tau-1}$ the hypothesis H_i is rejected, then a clustering of 1's or 0's can be assumed and a structural break is likely. If a structural break occurs, it is valuable to know the exact time index of the break, e.g. to cut off the time series to improve forecasting methods.

Case I - If the First Hypothesis is Rejected. Find first index i for which the hypothesis is accepted. Find index j in interval $[i, i+a_1]$ for which the difference of the average over $[i, j]$ and the average over $[j+1, i+a_1]$ is maximal.

Case II - If the First Hypothesis is Accepted. Find the first index i for which the hypothesis is rejected. Continue as suggested above in order to optimize j.

Since averages on small numbers of samples might cause misleading results, it is recommended e.g. to enlarge the interval artificially by ϵ and optimize j on $[i-\epsilon, i+a_1+\epsilon]$.

2.2 Transformation Routine

Finding an adequate transformation routine of course requires a clear definition of what shall be detected as a discontinuous behavior, and consequently its design is absolutely dependent on this definition. Therefore this paper cannot provide a general answer to this problem. Instead, a strategy for the practical problem considered in the context of this paper is discussed in this section.

Consider a company with a large amount of products, whose demand needs to be forecasted day by day, e.g. a supermarket or any wholesaler. Obviously, forecasting cannot be done manually in such cases, and reliable strategies have to be chosen to solve the problem. The success of forecasting strategies depends on the quality of considered time series and on the robustness of applied methods. A reasonable approach to improving the forecast quality is to improve the quality of the input data using preprocessing methods, and a method to detect and remove structural breaks as presented here. Following a structural break means a rapid and strong shift of the mean demand of a certain article. In other words, one can find two different marginal distributions which can be separated at a certain point in time. Figure 1 shows an example for such a strong and rapid shift of the mean. In week 69 the behavior of the time series and the marginal distribution of the signal changes dramatically. The mean demand changes from 181 items based on the deliveries until week 69, to 119 items based on the deliveries starting from week 70. Since safety stock levels are often affected by variance or standard deviation, an estimation of stock level based on the complete time series can lead to overstocking in cases as described above.

Taking the previous considerations into account, the task can be summarized as follows: If a set of data is likely to correspond to two distributions with different means, is there a point in time which can be used to differentiate between both marginal distributions, or do the random numbers come alternating from both distributions? Having obtained the results from section 2.1 it is necessary to find an adequate transformation routine, which reflects the change of interest.

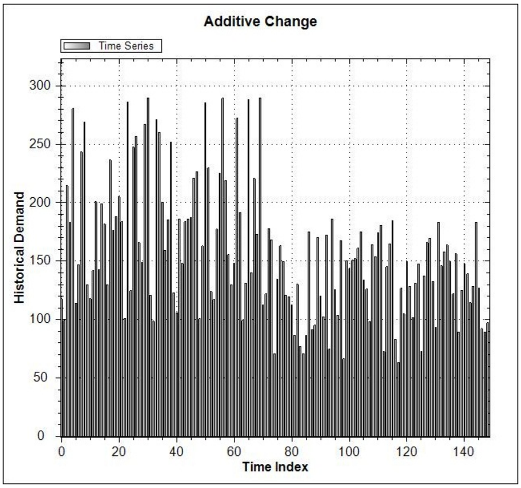


Fig. 1. Example of an additive change. At break time index 69 the arithmetic mean shifts from 181 items to 119 items.

A well-known clustering algorithm is the k-means-clustering as described e.g. in [16]. Clustering is known to be NP-hard in standard scenarios, hence polynomial clustering heuristics like k-means-clustering do not guarantee optimal solutions. Since in this case clustering is performed for only one dimension the algorithm converges to the optimum as described in [17]. The goal of the algorithm is to find k clusters in n-dimensional space, where a cluster is described by its n-dimensional mean vector. Whereas some modifications of the algorithm allow an adaptive fit of k to the data samples, the problem described above requires to set $k = 2$, as the task is to find two separate distributions of samples. Unfortunately, using exactly two clusters brings up a weakness of this method concerning outliers. In order to prevent identifying outliers as a cluster, it is recommended to remove outliers prior to the analysis, e.g. by using the 3σ rule, i.e. eliminating samples which deviate from the mean value by more than three times the standard deviation, compare [18].

Having obtained two clusters, C_0 and C_1, on basis of the marginal distributions, the transformation routine $P_C : \mathbb{R} \to \{0, 1\}$ can be defined as follows and the time series $x_i \in \mathbb{R}$ can be transformed to $y_i \in \{0, 1\}$

$$y_i = \begin{cases} 0 \; x \in C_0 \\ 1 \; x \in C_1 \end{cases} \tag{9}$$

Just as the design of the transformation routine depends on the task considered, certain parameters have to be set depending on it. Since the task in this case is to detect a clustering of samples from different distributions, it is recommended to set the length of the analyzed subsequence τ in section 2.1 to

$$\tau = min \left\{ |C_0| \, , |C_1| \right\} \tag{10}$$

by default as an intermediate result. A more detailed analysis is prepared in section 3.1. In order to reduce the number of false alarms it is helpful to define an offset. This has the effect that a time series can only be reduced to a certain minimum number of samples. Another strategy to prevent false alarms is to demand a minimum size of each cluster. Both points are justified by the goal to analyze whether the distribution of samples has reliably changed and choice of settings should depend on risks associated with increasing either type I or type II error. Another way to prevent misleading results is to combine hypothesis tests using boolean expressions.

2.3 Boolean Operations on Hypothesis Tests

Another transformation routine $P_R : \mathbb{R} \to \{0,1\}$ is needed to combine the hypothesis test from above with a second one. In order to be able to distinguish between different hypothesis tests, the notation for a hypothesis test will be as follows: $H(P, X)$ where $P : \mathbb{R} \to \{0,1\}$ and $X = \{x_i | i = 1, \ldots, s\}$.

Now the goal is to test whether or not the time series x_i follows a stable trend which can be described by a linear function. For motivation, consider such a function and recall that this time series would cause hypothesis $H(P_C, X)$ to be rejected, since the first half of the samples would be assigned to one cluster and the second half would be assigned to the second cluster.

The linear regression can be described in form

$$x_i = a \cdot i + x_0 + e_i \tag{11}$$

where e_i denotes the error term. Assuming that e_i with $i \in [0, s]$ is a white noise series, then the least squares method produces good estimates (compare [19]).

The regression line can be used as a discriminant function $P_R : \mathbb{R} \to \{0,1\}$ to transform the time series x_i with

$$y_i = \begin{cases} 0 \; a \cdot i + x_0 - x_i > 0 \\ 1 \; a \cdot i + x_0 - x_i \leq 0 \end{cases} \tag{12}$$

to $y_i \in \{0,1\}$ as demanded.

Assuming white noise or any other symmetric distribution density one can expect the regression line to separate the data set in two almost equal parts which leads to $p \approx 0.5$. If the number of samples is large enough the law of large numbers suggests that significant deviations are highly improbable. The size of the moving window should consider the total number of samples and a certain minimum, e.g.

$$\tau = max\left\{\tau_{min}, f \cdot s\right\} \ . \tag{13}$$

The factor $f \in [0,1]$ determines the accuracy of the test and τ_{min} influences the precision of the Chernoff bound. Finally, by combining both hypothesis tests

$$H = \begin{cases} \text{reject} \;\; H(P_C, X) \wedge \neg H(P_R, X) \\ \text{accept else} \end{cases} \tag{14}$$

no time series with a stable trend is truncated. In case of rejecting H the outcome of $H(P_C, X)$ can be used to shorten the time series for further investigations.

2.4 Dealing with Multiple Structural Breaks

In order to deal with multiple structural breaks, an iterative procedure of the algorithm presented within this paper is applied. Given a time series $x_1, \dots, x_s$ and the algorithm detects a structural break at time index b, the algorithm is applied again on time series $x_b, \dots, x_s$ until convergence, i.e. no further change-point is detected on the subsequence. If one is interested in identifying all change-points, the procedure can be applied to all remaining subsequence until convergence.

2.5 A Brief Note on Dealing with Nonadditive Changes

Nonadditive changes are defined in e.g. [14] as changes in variance, correlations, spectral characteristics, and dynamics of the signal or system. Hence, these types of changes are considered to be more complex to detect than additive changes, i.e. shifts in the mean value. Although additive changes play the central role in the following application on real data, the algorithm can easily be adapted to detect nonadditive changes. In order to demonstrate the flexibility of the novel approach, a rough recipe for this adaptation is provided.

The task of detecting either additive or nonadditive changes can be summarized as generating residuals of the original measurements that reflect the changes of interest, which are in this particular case of nonadditive nature. As stated above, instead of residuals the algorithm introduced within this paper demands a sequence of binary numbers, which is interpreted as the outcome of a binary stochastic process $\{Y_i\}_{i\in\mathbb{N}}$ with $\Omega := \{0, 1\}$. Afterwards, the sequence can be analyzed using Chernoff's Bounding Method as described in section 2.1.

Alternatively to the transformation routine P_C, introduced in section 2.2, one can define new routines to face nonadditive changes. Specifically when analyzing changes in variance or higher statistical moments, one challenge is to avoid problems with shifting means in time series. Therefore, preprocessing in terms of e.g. high pass or wavelet filtering is recommendable, of which [20] provides a good survey on the latter. The outcome of the preprocessing shall be denoted as $x'_1, \dots, x'_s \in \mathbb{R}$ and is assumed to be free of shifts in mean.

In a second step, the following transformation results in a reduction of the variance change detection problem to the additive change detection problem solved by the procedure defined in section 2.2.

$$\dot{x}_i = \begin{cases} \|x'_1\| & i = 1 \\ \left\|x'_i - x'_{i-1}\right\| & i \geq 2 \end{cases} \tag{15}$$

Assuming that elements of time series $x_1, \dots, x_s$ are stochastically independent and that elements x_i and x_{i+1} follow the same distribution, it is known that the variance of distributions of derivatives of two i.i.d. variables summarizes to $2 \cdot \sigma^2$, compare [21]. However, this ensures that information on shifts in variance is not destroyed by the derivation in equation 15. Furthermore, using the absolute value in equation 15 and the symmetric character of the derivatives distribution reduces the problem to the additive change detection problem.

3 Model Optimization and Performance Tests on Artificial Data

The algorithm presented in section 2.1 requires default values, the probability bound γ and the size of the sliding time window τ. Default settings have been proposed in section 2.1, and a detailed analysis is discussed exemplarily for $H\left(P_C, X\right)$ in the following section. The second goal of this section is a comparison of this algorithm to competitors.

3.1 Parameter Optimization

A generator for symmetric time series is needed to evaluate the algorithm on artificial data. In this example time series consist of 150 random samples with given expectation and variance. For simplicity it is assumed that samples are uniformly distributed. With probability p_s a structural break occurs, which means that expectation and variance change at a certain time index.

Having the information for each time series on whether it contains of a structural break and, if so, at which time index, the algorithm can be applied and evaluated. Results and previous knowledge can be compared and ratios of successful diagnostics determined. A diagnosis will be considered successful if a structural break occurred in a time series and was detected by the algorithm, or if no structural break occurred in a time series and no structural break was detected by the algorithm. In contrast, a diagnosis is not successful, if the algorithm raises a false alarm or misses a structural break. Obviously the target function maximizes the ratio of successful diagnostics.

In order to analyze the impact of γ and τ, the algorithm is applied to 1000 artificial time series using different pairs of settings for γ and τ. In figure 2 a result of the analysis is presented. The window size τ is described as a factor of the number of 1's in Y with ratio $r \in [0.05, 3]$ as

$$\tau = r \cdot \left|A^1\right| \tag{16}$$

and the probability bound γ is taken from $\gamma \in [0.01, 1]$. Dark areas in the graph represent high ratios of successful diagnostics for the corresponding pair of γ and τ, red areas represent maximal ratios. Since the maximum ratio of successful diagnostics is located around $\gamma = 0.1$ and $r = 1.0$ figure 3 shows a zoom on this section for $\gamma \in [0.01, 0.2]$ and $r \in [0.4, 1.4]$.

The analysis shows that the parameter combination of $\gamma = 0.1$ and $r = 1.0$ has highest likelihoods for successful diagnostics for the given set of time series. Further analyses have shown that changes in the databasis have only little influence on the results obtained for the parameter optimization. E.g. in one of these analyses, samples of each time series were replaced by normally distributed samples. In general, the authors of this paper recommend using default parameter settings or, if computing time is of no relevance, repeating the analysis for concerned databases.

3.2 Benchmarking

The goal of this section is to compare the novel approach to a well known algorithm concerning performance and diagnostic capability. For comparison the sequential probability ratio test (SPRT) and exact bounds, i.e. quantiles of the binomial distribution instead of Chernoff Bounds, will be used.

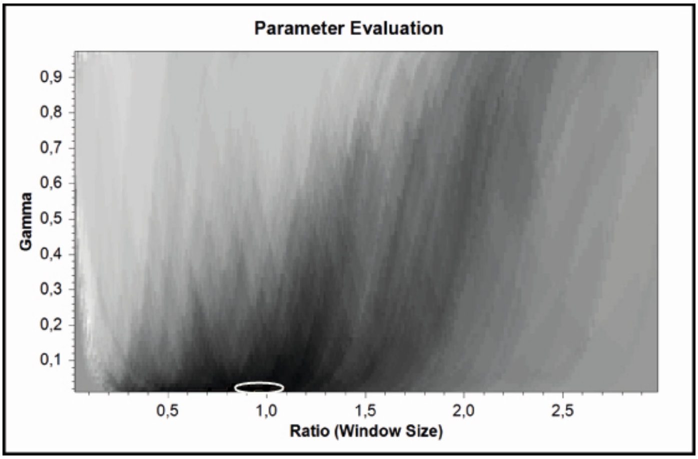


Fig. 2. Ratio of successful diagnostics for different parameter combinations. Dark areas represent high ratios, area with maximal ratios is circled.

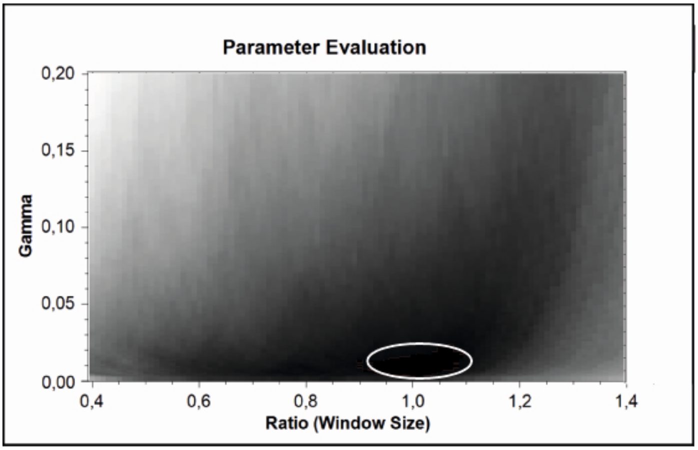


Fig. 3. Zoom on figure 2. Dark areas represent high ratios, area with maximal ratios is circled.

SPRT was invented by [22] and applied by e.g. [23] for dynamics sensor validation. The rough idea is to test whether a sequence of residuals $r_0, r_1, r_2, ..., r_n = \{R_n\}$ is probable to satisfy either hypothesis H_k or H_0

$$L_n = \frac{Pr\left\{\{R_n\}\left|H_k\right.\right\}}{Pr\left\{\{R_n\}\left|H_0\right.\right\}} \tag{17}$$

where L_n is the likelihood ratio. H_0 is the hypothesis that $\{R_n\}$ belongs to a normal distribution $\mathcal{N}\left(\mu, \sigma^2\right)$. Since a deviation of the mean is tested in this case, H_k is either the hypothesis that $\{R_n\}$ belongs to a normal distribution $\mathcal{N}\left(\mu + d, \sigma^2\right)$ or to a normal distribution $\mathcal{N}\left(\mu - d, \sigma^2\right)$, where d is the disturbance magnitude for the mean test.

The likelihood ratio is evaluated as follows: if $\log\left(L_n\right)$ is greater or equal to the logarithm of threshold A, then H_k is true. If $\log\left(L_n\right)$ is less or equal to the logarithm of B, then the null hypothesis is accepted. The thresholds A and B are related to the misidentification probabilities α, false alarm probability, and β, missed alarm probability, where

$$A = \frac{\beta}{1-\alpha} \tag{18}$$

and

$$B = \frac{1-\beta}{\alpha} \,. \tag{19}$$

The parameters μ and σ are usually estimated on a predetermined training set.

In order to compare and to estimate the advantage of using Chernoff Bounds instead of exact quantiles of the Binomial Distribution, both strategies were implemented. The exact quantiles can be calculated in time $O\left(n^2\right)$.

Test conditions for performance tests are similar to those in section 3.1. Each test consists of 1000 time series with or without structural breaks. Tests are performed for time series with length 150, 750, 1500 and 4500 and elapsed time is measured in seconds. Results are listed in table 1, which shows that the benefit of using Chernoff Bounds becomes obvious for long time series. Optimal parameter settings for all algorithms are neglected, hence the goal in this experiment is to measure execution time and not result quality.

Table 1. Runtime in seconds for 1000 time series with specified number of samples

Number of Samples	Chernoff Bounds	Binomial Distribution	SPRT
150	0.05	0.16	34.21
750	0.23	3.29	546.72
1500	0.42	17.14	1692.21
4500	1.40	289.86	10497.45

Test conditions for diagnostic capability tests are again quite similar to test conditions in section 3.1. 1000 time series with 150 samples each were generated. This procedure was repeated for samples of a Gaussian distribution and a uniform distribution. As best diagnostic capabilities are to be compared, the optimal parameter settings for each algorithm, as defined in section 3.1, were evaluated and applied. In case of time series with uniformly distributed samples, this means that optimal parameter settings for structural break detection using Chernoff Bounds are $\gamma = 0.01$ and factor window size $r = 1.0$, for exact quantiles $\gamma = 0.001$ and factor window size $r = 1.2$. For structural break detection via SPRT $\alpha = 0.001$, and $\beta = 0.05$ turn out to be the optimal parameters. The results are shown in table 2. For time series with normally distributed samples, the parameters have to be adapted: for Chernoff Bounds $\gamma = 0.015$ and factor window size $r = 0.98$ are used, for exact quantiles $\gamma = 0.001$ and factor window size $r = 0.98$ and for SPRT $\alpha = 0.001$ and $\beta = 0.05$.

Table 2. Ratio of successful and unsuccessful classifications. Basis of the test are 1000 time series with 150 samples each, following a uniform distribution.

Classification	Chernoff Bounds	Binomial Distribution	SPRT
Detections	95.21	97.79	79.56
Sensitives	92.56	80.53	94.75
Missed Alarms	4.79	2.21	20.44
False Alarms	7.44	19.47	5.25

Table 3. Ratio of successful and unsuccessful classifications. Basis of the test are 1000 time series with 150 samples each, following a normal distribution.

Classification	Chernoff Bounds	Binomial Distribution	SPRT
Detections	78.89	92.32	96.16
Sensitives	85.50	67.98	61.58
Missed Alarms	21.11	7.68	3.84
False Alarms	14.50	32.02	38.42

Table 2 and 3 present surprising results at first glance. Usually, one would expect exact bounds of the Binomial Distribution to produce better results than approximated Chernoff Bounds. But, since Chernoff Bounds are not exact, they might result in a higher threshold even for a smaller γ. This can lead to better detection and worse sensitive ratios or vice versa. Since structural breaks occur with probability $p_s = 0.5$ detection and sensitive ratios in table 2 and 3 are comparable. High ratios show that Chernoff Bounds perform well for uniform and Gaussian distributed samples.

4 Application on Real-World Data

This section provides a real-world application of the algorithm presented above. Within this section, P_C is permanently used as the transformation routine. The evaluation of the algorithm is based on 8002 real-world time series of a pharmaceutical wholesaler. These time series represent historical demands and, in their very nature, can imply seasonality, trends, slow or fast moving articles, or nonadditive changes as well as additive changes. The elements of each time series will be considered as independent and of unknown distribution, since no *a priori* information is available. Goal of this section is to show that the detection and removal of additive changes using the novel approach will reduce the forecast error significantly.

In section 4.1 forecast methods used for this evaluation are shortly introduced. In order to compare the novel approach to competitive strategies test scenarios are defined in section 4.2. Furthermore, the relative forecast error is defined as a measure to compare two given strategies. In section 4.3 results of evaluation are presented and discussed.

4.1 Forecast Methods

In order to estimate the value of preprocessing the following forecast methods have been implemented and applied on original and shortened time series.

- The arithmetic mean estimator is used as a representative of naive forecasting procedures. Additionally, this estimator should perform well on stationary time series.
- Single exponential smoothing is considered to be robust on seasonality, seasonal correlation, changing trends and suitable for forecasting in the presence of outliers as quoted in [24] and [25]. Therefore, it should perform well even in the presence of structural changes. Considered for original work are Brown and Holt in the 1950s, compare e.g. [26] and [27], and a review on exponential smoothing in general is provided in [28] or [29].
- Linear regression analysis is recommendable for predictions on basis of time series containing trends. State of the art applications are provided in [30], [31], and [32] to name but a few.

In combination, these algorithms address important issues of time series prediction. The selection procedure to decide which forecast method should be used for a particular time series can be described as best historical performance principle. This principle pretends that historical performance is an indicator for future performance. Formally speaking, the goal is to determine a method to predict $\hat{x}_{s+1}$. Each forecast method available can now be used to forecast w samples $\hat{x}_{s-w+1}, \ldots, \hat{x}_s$ of time series $x_1, \ldots, x_s$. The best method is determined e.g. with respect to the average mean squared error

$$AMSE = \frac{1}{w} \sum_{t=s-w+1}^{s} \left(\hat{x}_t - x_t\right)^2 \tag{20}$$

and used to estimate $\hat{x}_{s+1}$.

4.2 Design of Test Scenarios and Relative Error Estimates

The goal is to analyze whether preprocessing in terms of structural break detection is an improvement to forecasting or not. Hence, test scenarios will be defined which are composed of two preprocessing modes and a set of forecast functions. The preprocessing mode can be any one of the following:

- None (None). No preprocessing in the sense of structural break detection is applied at all. This mode will be used to illustrate the value of structural break detection.
- Chernoff Bounds (CB). The algorithm presented in this paper is applied for structural break detection. If a break is detected, the time series is abridged accordingly.
- Binomial distribution (BinDist). The algorithm presented in this paper is applied for structural break detection but instead of Chernoff's approximations the exact bounds of the Binomial distribution are used. This is done for comparison of both thresholds. Results from section 3.2 show that for short time series containing no more than 150 samples, the run time of the algorithm using exact thresholds rather than Chernoff Bounds can be approximated by a factor of three.
- Naive approach (NA). In order to compare accurate detection methods to a naive approach, one strategy will be to cut off the time series at point $b = \lceil s/2 \rceil$.

Furthermore, two sets of forecasting functions are defined:

- The first set (AME) only contains the arithmetic mean estimator. This is reasonable since it represents naive forecasting methods and should perform well especially on stationary time series.
- The second set (Combo) contains the arithmetic mean estimator, single exponential smoothing and linear regression for reasons given in section 4.1.

Scenarios are composed of preprocessing strategies and a set of forecast functions. Table 4 provides a list of all scenarios to be evaluated in section 4.3.

Table 4. Overview on scenarios. Each scenario contains a reviewed preprocessing strategy, a reference preprocessing strategy and a set of forecast functions applied on either preprocessed time series.

Scenario ID	Reviewed Strategy	Reference Strategy	Forecast Functions
1	CB	None	AME
2	CB	NA	AME
3	CB	BinDist	AME
4	CB	None	Combo
5	CB	NA	Combo
6	CB	BinDist	Combo
7	BinDist	None	AME
8	BinDist	NA	AME
9	BinDist	CB	AME
10	BinDist	None	Combo
11	BinDist	NA	Combo
12	BinDist	CB	Combo

The relative forecast error is estimated for each time series separately in the following way. In order to reduce type II errors or false alarms, the best historical performance principle, as introduced in section 4.1 for forecasting, is applied for the selection of the preprocessing strategy as well. If the reviewed strategy performed better in the past on $x_1, \ldots, x_{s-1}$ than the reference strategy in terms of $AMSE$, then the reviewed strategy is used for the actual forecast of sample x_s as well. If the reviewed strategy performed better in the past, then the relative error is measured. The $AMSE$ received by the reference strategy is denoted as $AMSE^{Ref}$ and the $AMSE$ received by the reviewed strategy as $AMSE^{Rev}$ and the estimates $\hat{x}_s^{Ref}$ and $\hat{x}_s^{Rev}$ at time index s, respectively. The residua are denoted as δ_s^{Ref} and δ_s^{Rev}, respectively.

Then the relative error R_η of time series η is defined as

$$R_\eta := \begin{cases} \frac{\left|\delta_s^{Ref}\right| - \left|\delta_s^{Rev}\right|}{\left|\delta_s^{Ref}\right|} & \left|\delta_s^{Rev}\right| < \left|\delta_s^{Ref}\right| \\ \\ \frac{\left|\delta_s^{Ref}\right| - \left|\delta_s^{Rev}\right|}{\left|\delta_s^{Rev}\right|} & \left|\delta_s^{Rev}\right| > \left|\delta_s^{Ref}\right| \\ \\ 0 & \text{else} \end{cases} \tag{21}$$

Consider that the $AMSE$ is estimated on $x_1, \ldots, x_{s-1}$ and the improvement might be negative, if $\hat{x}_s^{Rev}$ proves to be a worse estimator than $\hat{x}_s^{Ref}$, even if $AMSE^{Rev} < AMSE^{Ref}$. Hence, missed and false alarms will be measured as described in table 5. Whereas missed structural breaks fail to reduce forecast errors, false structural breaks increase forecast errors. Obviously it is worthwhile avoiding both of them. Formula 21 returns the percentage error decrease in case $\hat{x}_s^{Rev}$ is a better estimator than $\hat{x}_s^{Ref}$ and the percentage error increases in case of false alarms. Finally, the relative error improvement E_η of time series η is defined as

$$E_\eta := \begin{cases} R_\eta \; AMSE^{Rev} < AMSE^{Ref} \\ \\ 0 \quad \text{else} \end{cases} \tag{22}$$

Table 5. Classification of historical and present strategies. In order to reduce false alarms the best historical performance principle is applied, but on account of missed alarms. Measuring false and missed alarms indicates success of the method.

Classification	Historical Performance	Present Performance
Sensitivity	$AMSE^{Rev} > AMSE^{Ref}$	$\delta_s^{Rev} > \delta_s^{Ref}$
Specificity	$AMSE^{Rev} < AMSE^{Ref}$	$\delta_s^{Rev} < \delta_s^{Ref}$
False alarm	$AMSE^{Rev} < AMSE^{Ref}$	$\delta_s^{Rev} > \delta_s^{Ref}$
Missed alarm	$AMSE^{Rev} > AMSE^{Ref}$	$\delta_s^{Rev} < \delta_s^{Ref}$

4.3 Evaluation

The evaluation of the algorithm is based on 8002 real-world time series of a pharmaceutical wholesaler. The elements of each time series have been considered to be independent and of unknown distribution. In this section, results of test scenarios as defined in section 4.2 are discussed. In order to increase the clarity of graphical presentation, the scenarios have been subdivided into four groups of three each. The performance measures in terms of significance, statistical power, forecast error improvement and runtime are summarized in table 6 for all scenarios.

When performing the tests, it became obvious that results have been volatile to a certain extend for the following reason. Assume $AMSE^{Rev} < AMSE^{Ref}$ and $\hat{x}^{Rev}$ is taken as the next forecast, ϕ is the true distribution of $x_s \in X$ and

$$\left|E\left[X\right] - \hat{x}_s^{Rev}\right| < \left|E\left[X\right] - \hat{x}_s^{Ref}\right| \tag{23}$$

then the probability P that $\hat{x}_s^{Ref}$ is a better estimator for x_s than $\hat{x}_s^{Rev}$ is given by

$$P := \begin{cases} \int_a^\infty \phi\left(x\right) dx \; \hat{x}_s^{Rev} < \hat{x}_s^{Ref} \\ \\ \int_0^a \phi\left(x\right) dx \quad \hat{x}_s^{Rev} > \hat{x}_s^{Ref} \end{cases} \tag{24}$$

where $a = \frac{1}{2}\left(\hat{x}_s^{Rev} + \hat{x}_s^{Ref}\right)$.

In order to reduce the volatility of the results, the test sequence to determine the performance indicators has been increased from one to ten, or in other words, instead

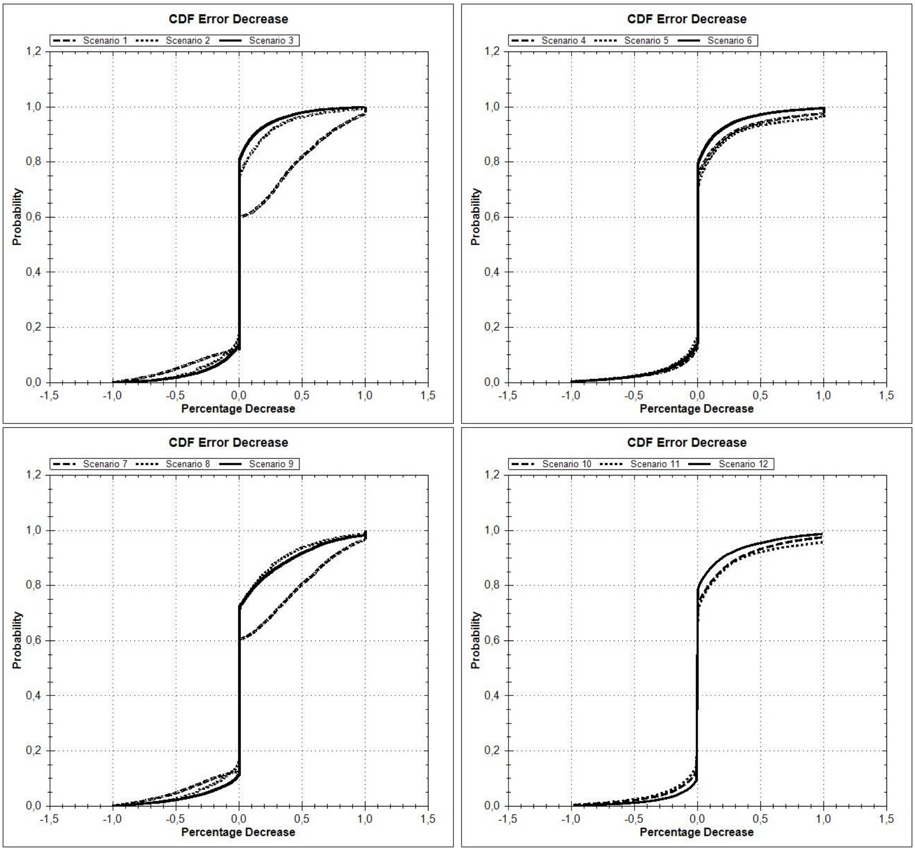


Fig. 4. CDF's of the relative forecast error reduction of scenario one to twelve

of estimating $\hat{x}_s^{Rev}$ and $\hat{x}_s^{Ref}$, the sequences $\hat{x}_{s-9}^{Rev}, \ldots, \hat{x}_s^{Rev}$ and $\hat{x}_{s-9}^{Ref}, \ldots, \hat{x}_s^{Ref}$ have been estimated.

The top left graph of figure 4 shows the cdf for the first three scenarios. In each of the three scenarios, structural break detection using Chernoff Bounds was the reviewed method and the arithmetic mean estimator was the only forecast method used. As can be seen in the figure, the Chernoff Bounds competed well against all three competitors. The cdf takes a forecast error into account if an additive change has been detected and therefore $AMSE^{Ref} \neq AMSE^{Ref}$. The relative error has been measured as depicted in equations 22 and 21. The curve shows that for scenario one about 40% of the forecasts could be improved if a structural break had been detected. In 18% of the forecasts, the error could be reduced by more than 50%. The forecast error was increased by 50% in less than 2% of the forecasts, due to false alarms. The first scenarios show that especially when dealing with naive forecast methods, structural break detection results in great improvements in terms of relative forecast errors.

The top right graph of figure 4 displays similar scenarios to those shown in the top left graph, with more sophisticated forecast methods having been used than in the former.

Comparing scenario one and four, the effect of improving forecast methods can be seen if no preprocessing has been applied before.

Scenarios one to six have been repeated, using the exact bounds of the binomial distribution instead of Chernoff's approximations. The results of scenario seven to twelve, compare bottom graphs in figure 4, are similar to those of scenario one to six, but using the exact bounds increases the relative forecast error reduction as expected.

Table 6. Sensitivity, missed alarm, specificity, and false alarm classify success in terms of forecast error reduction only if a structural break has been detected. Ratios of improved or worsened forecasts reflect success in terms of forecast error reduction proportional to the overall number of forecasts done. The runtime is standardized by the fastest scenario, which took approximately four seconds.

Scenario ID	Sensitivity	Missed Alarm	Specificity	False Alarm	Ratio of Improved FCs	Ratio of Worsen FCs	Relative Runtime
1	0.67	0.33	0.81	0.19	0.26	0.06	1
2	0.71	0.29	0.62	0.38	0.28	0.17	1
3	0.78	0.22	0.71	0.29	0.21	0.09	3
4	0.71	0.29	0.73	0.27	0.20	0.07	190
5	0.62	0.38	0.62	0.38	0.29	0.18	96
6	0.77	0.23	0.72	0.28	0.25	0.10	103
7	0.61	0.39	0.79	0.21	0.33	0.09	2
8	0.67	0.33	0.65	0.35	0.30	0.16	2
9	0.71	0.29	0.78	0.22	0.26	0.07	3
10	0.64	0.36	0.71	0.29	0.24	0.10	127
11	0.60	0.40	0.64	0.36	0.28	0.16	82
12	0.72	0.28	0.77	0.23	0.24	0.07	103

Table 6 extends the results given in figure 4. As defined in table 5, sensitivity, specificity, false alarms, and missed alarms have been measured. This classification can only take into account forecasts of time series, for which structural breaks have been detected. Since the time series represent real-life data instead of artificial ones, break dates are unknown. Therefore relative error reduction is used as performance measure. The ratio of improved or worsened forecasts takes all forecasts into account, i.e. it is the absolute number of specificities or false alarms divided by the total number of forecasts done, respectively. For example in scenario one, 26% of all forecasts have been positively influenced by using structural break detection.

The results in table 6 show the positive effect of using sophisticated preprocessing methods and diverse forecasting methods. Results of scenarios in which naive preprocessing was involved appear to be arbitrary, indicated by high ratios of both improved and worsened forecasts.

The runtime of scenarios has been standardized. Scenario one took about four seconds to complete. Results from section 3.2 show that the runtime of exact bounds differs by a factor of three in comparison to Chernoff's Inequalities for short time series and increases exponentially for longer time series. The increase of runtime when using a combination of forecasting methods is considerable. Taking into account runtime and error reduction ratios, applying preprocessing methods appears to be very worthwhile.

5 Conclusions and Future Prospects

In section 4.3, the evaluation of the novel approach to change-point detection and its impact on forecasting have been performed and discussed. Results are striking in terms of forecast error reduction and runtime, compare table 6.

The scenarios contained both naive and sophisticated forecast and structural break detection methods. Table 6 shows that using sophisticated forecast methods raises the runtime enormously in relation to its error improvement, compare scenario one and four. A possible explanation for the success of preprocessing in terms of change-point detection might be that it is more widely applicable than additional forecast algorithms. New forecasting algorithms are often designed to deal with special characteristics on certain time series, whereas prepocessing will affect forecasting performance in a wider range of time series.

The algorithm used in this paper is designed to deal with additive changes, i.e. shifts in the mean value. Nonadditive changes which occur in variance, correlations, spectral characteristics, and dynamics of the signal or system, compare [14], are the topic of future work. It will be shown that this detection problem can be solved by adequate transformation routines that reflect changes of interest. The special aim will be to design those transformation routines with respect to efficiency and robustness. Section 2.5 provided a brief note to this topic in order to show the flexibility of the novel approach.

From the theoretical point of view, the current application is an offline detection problem, but real-world problems such as asset monitoring, compare [33], or econometric applications, compare [34], demand online detection of structural changes. In future work, this algorithm will be applied to online detection problems, which will demand new performance indicators such as mean time between false alarms or mean delay for detections.

The goal of this paper is to demonstrate the applicability of the algorithm to a real-world problem and facing real-world data. Future prospects will be to analyze more general performance indicators as proposed for example in [14] such as mean time between false alarms, probability of false detections, mean delay for detection, probability of nondetection, statistical power, and required effect size to name but a few. The goal will be to answer these questions analytically and by simulation.

References

1. Perron, P.: Dealing with Structural Breaks. Palgrave handbook of econometrics 1, 278–352 (2006)
2. Kawahara, Y., Sugiyama, M.: Change-point Detection in Time Series Data by Direct Density-Ratio Estimation. In: Proceedings of 2009 SIAM International Conference on Data Mining (SDM 2009), pp. 389–400 (2009)
3. Markou, M., Singh, S.: Novelty Detection: a Review–Part 1: Statistical Approaches. Signal Processing 83, 2481–2497 (2003)

4. Guralnik, V., Srivastava, J.: Event Detection from Time Series Data. In: Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 33–42. ACM (1999)
5. Ma, J., Perkins, S.: Time Series Novelty Detection Using One-class Support Vector Machines. In: Proceedings of the International Joint Conference on Neural Networks, vol. 3, pp. 1741–1745 (2003)
6. Ibaida, A., Khalil, I., Sufi, F.: Cardiac abnormalities detection from compressed ECG in wireless telemonitoring using principal components analysis (PCA). In: 2009 5th International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 207–212. IEEE (2010)
7. Feller, S., Chevalier, R., Morsili, S.: Parameter Disaggregation for High Dimensional Time Series Data on the Example of a Gas Turbine. In: Proceedings of the 38th ESReDA Seminar, Pécs, H, pp. 13–26 (2010)
8. Murad, U., Pinkas, G.: Unsupervised Profiling for Identifying Superimposed Fraud. In: Żytkow, J.M., Rauch, J. (eds.) PKDD 1999. LNCS (LNAI), vol. 1704, pp. 251–261. Springer, Heidelberg (1999)
9. Fujimaki, R., Yairi, T., Machida, K.: An Approach to Spacecraft Anomaly Detection Problem Using Kernel Feature Space. In: Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 401–410. ACM (2005)
10. Schwabacher, M., Oza, N., Matthews, B.: Unsupervised Anomaly Detection for Liquid-Fueled Rocket Propulsion Health Monitoring. In: Proceedings of the AIAA Infotech@ Aerospace Conference. American Institute for Aeronautics and Astronautics, Inc., Reston (2007)
11. Gustafsson, F.: Estimation and Change Detection of Tire-Road Friction Using the Wheel Slip. IEEE Control System Magazine 18, 42–49 (1998)
12. Ide, T., Kashima, H.: Eigenspace-based Anomaly Detection in Computer Systems. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 440–449. ACM (2004)
13. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys (CSUR) 41, 1–58 (2009)
14. Basseville, M., Nikiforov, I.: Detection of Abrupt Changes: Theory and Application. Prentice-Hall, Inc. (1993)
15. Chernoff, H.: A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the Sum of Observations. The Annals of Mathematical Statistics 23, 493–507 (1952)
16. Press, W., Teukolsky, S., Vetterling, W., Flannery, B.: Numerical Recipes: The Art of Scientific Computing. Cambridge University Press (2007)
17. Hartigan, J., Wong, M.: Algorithm AS 136: A k-means Clustering Algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics) 28, 100–108 (1979)
18. Wadsworth, H.: Handbook of statistical methods for engineers and scientists. McGraw-Hill Professional (1997)
19. Tsay, R.: Analysis of Financial Time Series. Wiley-Interscience (2005)
20. Strang, G.: Wavelets and dilation equations: A brief introduction. Siam Review 31, 614–627 (1989)
21. Feller, W.: An introduction to probability theory and its applications. Wiley-India (2009)
22. Wald, A.: Sequential Tests of Statistical Hypotheses. The Annals of Mathematical Statistics 16, 117–186 (1945)
23. Herzog, J.: Dynamics Sensor Validation for Reusable Launch Vehicle Propulsion. Presented at 34th AIAA/ASME/ASEE Joint Propulsion Conference and Exhibit, Cleveland, OH, US, July 13-15 (1998)
24. Taylor, J.: Multi-item sales forecasting with total and split exponential smoothing. Journal of the Operational Research Society (2010)

25. Gelper, S., Fried, R., Croux, C.: Robust forecasting with exponential and Holt-Winters smoothing. Journal of Forecasting 29, 285–300 (2010)
26. Holt, C.: Forecasting trends and seasonals by exponentially weighted moving averages. ONR Memorandum 52 (1957)
27. Brown, R.: Statistical forecasting for inventory control. McGraw-Hill, New York (1959)
28. Gardner Jr., E.: Exponential smoothing: The state of the art. Journal of Forecasting 4, 1–28 (1985)
29. Makridakis, S., Wheelwright, S., Hyndman, R.: Forecasting methods and applications. Wiley-India (2008)
30. Ng, T., Skitmore, M., Wong, K.: Using genetic algorithms and linear regression analysis for private housing demand forecast. Building and Environment 43, 1171–1184 (2008)
31. Xia, B., Zhao, C.: The Application of Multiple Regression Analysis Forecast in Economical Forecast: The Demand Forecast of Our Country Industry Lavation Machinery in the Year of 2008 and 2009. In: Second International Workshop on Knowledge Discovery and Data Mining, WKDD 2009, pp. 405–408 (2009)
32. Pinson, P., Nielsen, H., Madsen, H., Nielsen, T.: Local linear regression with adaptive orthogonal fitting for the wind power application. Statistics and Computing 18, 59–71 (2008)
33. Feller, S., Pauli, D., Kmiecik, F.: Robust AAKR and Modern Signal Transformation Methods for Fault Detection and Analysis. In: Proceedings of the 37th ESReDA Seminar, Baden, CH, pp. 57–63 (2009)
34. Zeileis, A., Leisch, F., Kleiber, C., Hornik, K.: Monitoring Structural Change in Dynamic Econometric Models. Journal of Applied Econometrics 20, 99–121 (2005)

Analytical-Numerical Localization of Hidden Attractor in Electrical Chua's Circuit

Nikolay Kuznetsov[1,2], Olga Kuznetsova[1,2],
Gennady Leonov[2], and Vladimir Vagaitsev[1,2]

[1] University of Jyväskylä, MIT Department
P.O. Box 35 (Agora), FIN-40014, Jyväskylä, Finland
[2] St.Petersburg State University, Applied Cybernetics Department
Universitetsky pr. 28, St. Petersburg, 198504, Russia
nkuznetsov239@gmail.com
http://www.math.spbu.ru/nk/

Abstract. Study of hidden oscillations and hidden chaotic attractors (basin of attraction of which does not contain neighborhoods of equilibria) requires the development of special analytical-numerical methods. Development and application of such methods for localization of hidden chaotic attractors in dynamical model of Chua's circuit are demonstrated in this work.

Keywords: Hidden oscillation, Chua circuit, Hidden attractor localization.

1 Introduction

The classical attractors of Lorenz [1], Rossler [2], Chua [3], Chen [4], and other widely-known attractors are those excited from unstable equilibria. From computational point of view this allows one to use standard numerical method, in which after transient process a trajectory, started from a point of unstable manifold in the neighborhood of equilibrium, reaches an attractor and identifies it.

However there are attractors of another type: *hidden attractors, a basin of attraction of which does not contain neighborhoods of equilibria* [5]. Here equilibria are not "*connected*" with attractor and creation of numerical procedure of integration of trajectories for the passage from equilibrium to attractor is impossible because the neighbourhood of equilibrium does not belong to such attractor. The simplest examples of systems with such hidden attractors are hidden oscillations in counterexamples to widely-known Aizerman's and Kalman's conjectures on absolute stability (see, e.g., [8,10]). Similar computational problems arise in investigation of semi-stable and nested limit cycles in 16th Hilbert problem (see, e.g., [11,12,13]).

In 2010, for the first time, a *chaotic hidden attractor* was computed by the authors in generalized Chua's circuit (which can be used for hidden chaotic communication [17]) and then one was discovered in classical Chua's circuit.

Further a special analytical-numerical algorithm for localization of hidden attractors is considered.

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 149–158.
springerlink.com

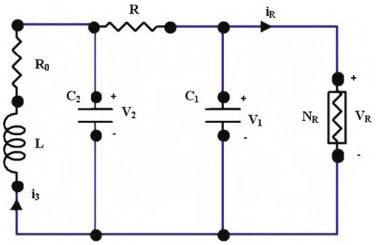


Fig. 1. Classical Chua's circuit

Chua's circuit (see Fig. 1) can be described by differential equations in dimensionless coordinates:

$$\begin{aligned} \dot{x} &= \alpha(y - x) - \alpha f(x), \\ \dot{y} &= x - y + z, \\ \dot{z} &= -(\beta y + \gamma z). \end{aligned} \tag{1}$$

Here the function

$$\begin{aligned} f(x) = m_1 x + (m_0 - m_1)\mathrm{sat}(x) = \\ = m_1 x + \frac{1}{2}(m_0 - m_1)(|x+1| - |x-1|) \end{aligned} \tag{2}$$

characterizes a nonlinear element, of the system, called *Chua's diode*; $\alpha, \beta, \gamma, m_0, m_1$ are parameters of the system. In this system it was discovered the strange attractors [14,15] called then Chua's attractors. All known classical Chua's attractors are the attractors that are excited from unstable equilibria and this makes it possible to compute such attractors with relative easy (see, e.g., attractors gallery in [16]).

The applied in this work algorithm shows the possibility of existence of hidden attractor in system (1). Note that L. Chua himself, analyzing in the work [3] different cases of attractor existence in Chua's circuit, does not admit the existence of such hidden attractor.

2 Analytical-Numerical Method for Hidden Attractors Localization

For numerical location of hidden oscillations it is turns out to be effective methods based on homotopy where a sequence of similar systems is consider such that initial data for numerical localization of periodic solution (starting periodic solution) in the first starting system can be obtained analytically and then transformation of this starting periodic solution in the transition from one system to another is followed numerically.

Consider a system

$$\frac{d\mathbf{x}}{dt} = \mathbf{P}\mathbf{x} + \boldsymbol{\psi}(\mathbf{x}), \mathbf{x} \in \mathbb{R}^n, \tag{3}$$

where $\mathbf{P}$ is a constant $n \times n$-matrix, $\boldsymbol{\psi}(\mathbf{x})$ is a continuous vector-function, and $\boldsymbol{\psi}(0) = 0$.

Define a matrix $\mathbf{K}$ in such a way that the matrix

$$\mathbf{P}_0 = \mathbf{P} + \mathbf{K} \tag{4}$$

has a pair of purely imaginary eigenvalues $\pm i\omega_0$ $(\omega_0 > 0)$ and the rest of its eigenvalues have negative real parts. We assume that such $\mathbf{K}$ exists. Rewrite system (3) as

$$\frac{d\mathbf{x}}{dt} = \mathbf{P}_0\mathbf{x} + \boldsymbol{\varphi}(\mathbf{x}), \tag{5}$$

where $\boldsymbol{\varphi}(\mathbf{x}) = \boldsymbol{\psi}(\mathbf{x}) - \mathbf{K}\mathbf{x}$.

Introduce a finite sequence of functions $\boldsymbol{\varphi}^0(\mathbf{x}), \boldsymbol{\varphi}^1(\mathbf{x}), ..., \boldsymbol{\varphi}^m(\mathbf{x})$ such that the graphs of neighboring functions $\boldsymbol{\varphi}^j(\mathbf{x})$ and $\boldsymbol{\varphi}^{j+1}(\mathbf{x})$ slightly differ from one another, the function $\boldsymbol{\varphi}^0(\mathbf{x})$ is small, and $\boldsymbol{\varphi}^m(\mathbf{x}) = \varphi(\mathbf{x})$. Using a smallness of function $\boldsymbol{\varphi}^0(\mathbf{x})$, we can apply and mathematically strictly justify [6,7,8,9,10] the method of harmonic linearization (describing function method) for the system

$$\frac{d\mathbf{x}}{dt} = \mathbf{P}_0\mathbf{x} + \boldsymbol{\varphi}^0(\mathbf{x}), \tag{6}$$

and determine a stable nontrivial periodic solution $\mathbf{x}^0(t)$. For the localization of attractor of original system (5), we shall follow numerically the transformation of this periodic solution (a starting *oscillating attractor* — an attractor, not including equilibria, denoted further by $\mathcal{A}_0$) with increasing j. Here two cases are possible: all the points of $\mathcal{A}_0$ are in an attraction domain of attractor $\mathcal{A}_1$, being an oscillating attractor of the system

$$\frac{d\mathbf{x}}{dt} = \mathbf{P}_0\mathbf{x} + \boldsymbol{\varphi}^j(\mathbf{x}) \tag{7}$$

with $j = 1$, or in the change from system (6) to system (7) with $j = 1$ it is observed a loss of stability bifurcation and the vanishing of $\mathcal{A}_0$. In the first case the solution $\mathbf{x}^1(t)$ can be determined numerically by starting a trajectory of system (7) with $j = 1$ from the initial point $\mathbf{x}^0(0)$. If in the process of computation the solution $\mathbf{x}^1(t)$ has not fallen to an equilibrium and it is not increased indefinitely (here a sufficiently large computational interval $[0, T]$ should always be considered), then this solution reaches an attractor $\mathcal{A}_1$. Then it is possible to proceed to system (7) with $j = 2$ and to perform a similar procedure of computation of $\mathcal{A}_2$, by starting a trajectory of system (7) with $j = 2$ from the initial point $\mathbf{x}^1(T)$ and computing the trajectory $\mathbf{x}^2(t)$.

Proceeding this procedure and sequentially increasing j and computing $\mathbf{x}^j(t)$ (being a trajectory of system (7) with initial data $\mathbf{x}^{j-1}(T)$) we either arrive at the computation of $\mathcal{A}_m$ (being an attractor of system (7) with $j = m$, i.e. original system (5)), or, at a certain step, observe a loss of stability bifurcation and the vanishing of attractor.

To determine the initial data $\mathbf{x}^0(0)$ of starting periodic solution, system (6) with nonlinearity $\varphi^0(\mathbf{x})$ can be transformed by linear nonsingular transformation $\mathbf{S}$ to the form

$$\begin{aligned}
\dot{y}_1 &= -\omega_0 y_2 + \varepsilon\varphi_1(y_1, y_2, \mathbf{y}_3), \\
\dot{y}_2 &= \omega_0 y_1 + \varepsilon\varphi_2(y_1, y_2, \mathbf{y}_3), \\
\dot{\mathbf{y}}_3 &= \mathbf{A_3}\mathbf{y}_3 + \varepsilon\boldsymbol{\varphi}_3(y_1, y_2, \mathbf{y}_3)
\end{aligned} \tag{8}$$

Here $\mathbf{A_3}$ is a constant $(n-2)\times(n-2)$ matrix, all eigenvalues of which have negative real parts, $\boldsymbol{\varphi}_3$ is an $(n-2)$-dimensional vector-function, φ_1, φ_2 are certain scalar functions. Without loss of generality, it may be assumed that for the matrix $\mathbf{A_3}$ there exists positive number $\alpha > 0$ such that

$$\mathbf{x}_3^*(\mathbf{A_3} + \mathbf{A_3}^*)\mathbf{x}_3 \leq -2\alpha|\mathbf{x}_3|^2, \quad \forall\, \mathbf{x}_3 \in \mathbb{R}^{n-2} \tag{9}$$

Introduce the following describing function

$$\begin{aligned}\Phi(a) = \int\limits_0^{2\pi/\omega_0} \Big[&\varphi_1\left((\cos\omega_0 t)a, (\sin\omega_0 t)a, 0\right)\cos\omega_0 t + \\ +&\varphi_2\left((\cos\omega_0 t)a, (\sin\omega_0 t)a, 0\right)\sin\omega_0 t\Big]dt.\end{aligned}$$

Theorem 1. *[9] If it can be found a positive a_0 such that*

$$\Phi(a_0) = 0 \tag{10}$$

and $\Phi'(a_0) < 0$ then there is a periodic solution in system (6) with the initial data $\mathbf{x}^0(0) = \mathbf{S}(y_1(0), y_2(0), \mathbf{y}_3(0))^*$

$$y_1(0) = a_0 + O(\varepsilon),\ y_2(0) = 0,\ \mathbf{y}_3(0) = \mathbf{O_{n-2}}(\varepsilon). \tag{11}$$

Here $\mathbf{O_{n-2}}(\varepsilon)$ is an $(n-2)$-dimensional vector such that all its components are $O(\varepsilon)$.

3 Localization of Hidden Attractor in Chua's System

We now apply the above algorithm to analysis of Chua's system with scalar nonlinearity. For this purpose, rewrite Chua's system (1) in the form (3)

$$\frac{d\mathbf{x}}{dt} = \mathbf{P}\mathbf{x} + \mathbf{q}\psi(\mathbf{r}^*\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^3. \tag{12}$$

Here

$$\mathbf{P}, \mathbf{q}, \mathbf{r} = \begin{pmatrix} -\alpha(m_1+1) & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & -\gamma \end{pmatrix}, \begin{pmatrix} -\alpha \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix},$$
$$\psi(\sigma) = (m_0 - m_1)\mathrm{sat}(\sigma).$$

Introduce the coefficient k and small parameter ε, and represent system (12) as (6)

$$\frac{d\mathbf{x}}{dt} = \mathbf{P_0}\mathbf{x} + \mathbf{q}\varepsilon\varphi(\mathbf{r}^*\mathbf{x}), \tag{13}$$

where

$$\mathbf{P_0} = \mathbf{P} + k\mathbf{q}\mathbf{r}^* = \begin{pmatrix} -\alpha(m_1+1+k) & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & -\gamma \end{pmatrix},$$
$$\lambda_{1,2}^{\mathbf{P}_0} = \pm i\omega_0,\ \lambda_3^{\mathbf{P}_0} = -d,$$
$$\varphi(\sigma) = \psi(\sigma) - k\sigma = (m_0 - m_1)\mathrm{sat}(\sigma) - k\sigma.$$

In practice, to determine k and ω_0 it is used the transfer function $W(p)$ of system (3):

$$W_{\mathbf{P_0}}(p) = \mathbf{r}^*(\mathbf{P} - p\mathbf{I})^{-1}\mathbf{q},$$

where p is a complex variable. Then $\mathrm{Im}W(i\omega_0) = 0$ and k is computed then by formula $k = -(\mathrm{Re}W(i\omega_0))^{-1}$.
By nonsingular linear transformation $\mathbf{x} = \mathbf{S}\mathbf{y}$ system (13) can be reduced to the form

$$\frac{d\mathbf{y}}{dt} = \mathbf{A}\mathbf{y} + \mathbf{b}\varepsilon\varphi(\mathbf{c}^*\mathbf{y}), \tag{14}$$

where

$$\mathbf{A}, \mathbf{b}, \mathbf{c} = \begin{pmatrix} 0 & -\omega_0 & 0 \\ \omega_0 & 0 & 0 \\ 0 & 0 & -d \end{pmatrix}, \begin{pmatrix} b_1 \\ b_2 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ -h \end{pmatrix}.$$

Further, using the equality of transfer functions of systems (13) and (14), we obtain

$$W_{\mathbf{A}}(p) = \mathbf{r}^*(\mathbf{P}_0 - p\mathbf{I})^{-1}\mathbf{q}.$$

This implies the following relations

$$\begin{aligned} k &= \frac{-\alpha(m_1 + m_1\gamma + \gamma) + \omega_0^2 - \gamma - \beta}{\alpha(1+\gamma)}, \\ d &= \frac{\alpha + \omega_0^2 - \beta + 1 + \gamma + \gamma^2}{1+\gamma}, \\ h &= \frac{\alpha(\gamma + \beta - (1+\gamma)d + d^2)}{\omega_0^2 + d^2}, \\ b_1 &= \frac{\alpha(\gamma + \beta - \omega_0^2 - (1+\gamma)d)}{\omega_0^2 + d^2}, \\ b_2 &= \frac{\alpha\big((1+\gamma-d)\omega_0^2 + (\gamma+\beta)d\big)}{\omega_0(\omega_0^2 + d^2)}. \end{aligned} \tag{15}$$

System (13) can be reduced to the form (14) by the nonsingular linear transformation $\mathbf{x} = \mathbf{S}\mathbf{y}$. Having solved the following matrix equations

$$\mathbf{A} = \mathbf{S}^{-1}\mathbf{P_0}\mathbf{S}, \quad \mathbf{b} = \mathbf{S}^{-1}\mathbf{q}, \quad \mathbf{c}^* = \mathbf{r}^*\mathbf{S}, \tag{16}$$

one can obtain the transformation matrix

$$\mathbf{S} = \begin{pmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{pmatrix}.$$

By (11), for small enough ε we determine initial data for the first step of multistage localization procedure

$$\mathbf{x}(0) = \mathbf{S}\mathbf{y}(0) = \mathbf{S}\begin{pmatrix} a_0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} a_0 s_{11} \\ a_0 s_{21} \\ a_0 s_{31} \end{pmatrix}.$$

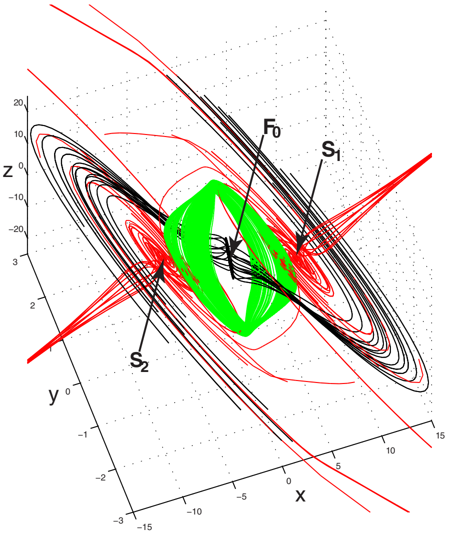


Fig. 2. Equilibrium, stable manifolds of saddles, and localization of hidden attractor

Returning to Chua's system denotations, for determining the initial data of starting solution of multistage procedure we have the following formulas

$$\begin{aligned} x(0) = a_0,\ y(0) = a_0(m_1 + 1 + k),\\ z(0) = a_0 \frac{\alpha(m_1 + k) - \omega_0^2}{\alpha}. \end{aligned} \tag{17}$$

Consider system (13) with the parameters

$$\begin{aligned} \alpha = 8.4562,\ \beta = 12.0732,\ \gamma = 0.0052,\\ m_0 = -0.1768,\ m_1 = -1.1468. \end{aligned} \tag{18}$$

Note that for the considered values of parameters there are three equilibria in the system: a locally stable zero equilibrium and two saddle equilibria.

Now we apply the above procedure of hidden attractors localization to Chua's system (12) with parameters (18). For this purpose, compute a starting frequency and a coefficient of harmonic linearization. We have

$$\omega_0 = 2.0392, \quad k = 0.2098\,.$$

Then, compute solutions of system (13) with nonlinearity $\varepsilon\varphi(x) = \varepsilon(\psi(x) - kx)$, sequentially increasing ε from the value $\varepsilon_1 = 0.1$ to $\varepsilon_{10} = 1$ with the step 0.1.

By (15) and (17) we obtain the initial data

$$x(0) = 9.4287, y(0) = 0.5945, z(0) = -13.4705$$

for the first step of multistage procedure for the construction of solutions. For the value of parameter $\varepsilon_1 = 0.1$, after transient process the computational procedure reaches the

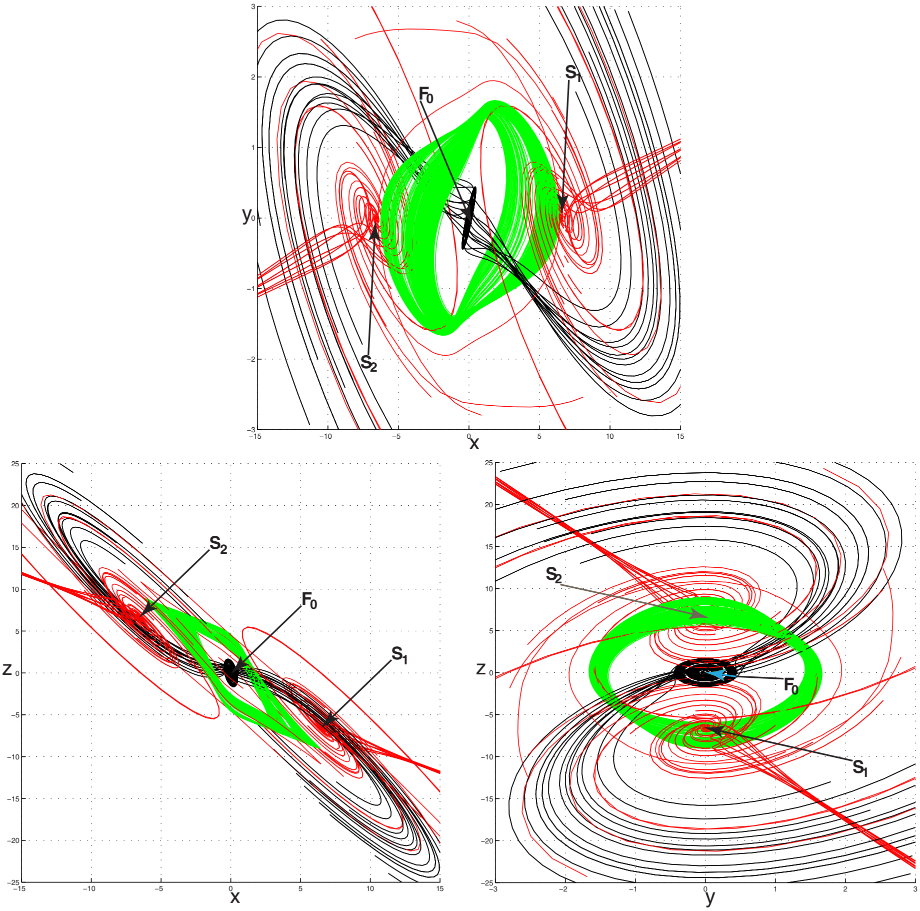


Fig. 3. Hidden attractor projections on (x, y), (x, z), and (y, z)

starting oscillation $\mathbf{x}^1(t)$. Further, by the sequential transformation $\mathbf{x}^j(t)$ with increasing the parameter ε_j, using the numerical procedure, for original Chua's system (12) the set $\mathcal{A}_{\mathrm{hidden}}$ is computed. This set is shown in Fig. 3.

The considered system has three stationary points: the stable zero point F_0 and the symmetric saddles S_1 and S_2. To zero equilibrium F_0 correspond the eigenvalues $\lambda_1^{F_0} = -7.9591$ and $\lambda_{2,3}^{F_0} = -0.0038 \pm 3.2495i$ and to the saddles S_1 and S_2 correspond the eigenvalues $\lambda_1^{S_{1,2}} = 2.2189$ and $\lambda_{2,3}^{S_{1,2}} = -0.9915 \pm 2.4066i$. The behavior of trajectories of system in a neighborhood of equilibria is shown in Fig. 3.

We remark that here positive Lyapunov exponent [18] corresponds to the computed trajectories.

By the above and with provision for the remark on the existence, in system, of locally stable zero equilibrium and two saddle equilibria (trajectories from the neighborhood of these saddles tend to zero or to infinity), we arrive at the conclusion that in $\mathcal{A}_{\mathrm{hidden}}$ a hidden strange attractor is computed.

Now let us consider localization of hidden oscillation in the Chua's circuit with modified nonlinear characteristic — discontinuous nonlinearity $\mathrm{sign}(x)$ instead of $\mathrm{sat}(x)$. For this we consider the system (12) with nonlinearity of special form

$$\theta^i(x) = \psi(x) + \frac{i}{n}\big((m_0 - m_1)\mathrm{sign}(x) - \psi(x)\big), \quad i = 1, \dots, n \tag{19}$$

and apply the same numerical procedure to the new system with $n = 10$, increasing the value of the parameter i from 1 up to 10. Projections of the solutions of the system (12) with nonlinearity (19) on the plane (x, y) for $i = 3,\ 5,\ 7,\ 10$ are shown in Figs. 4–7 respectively.

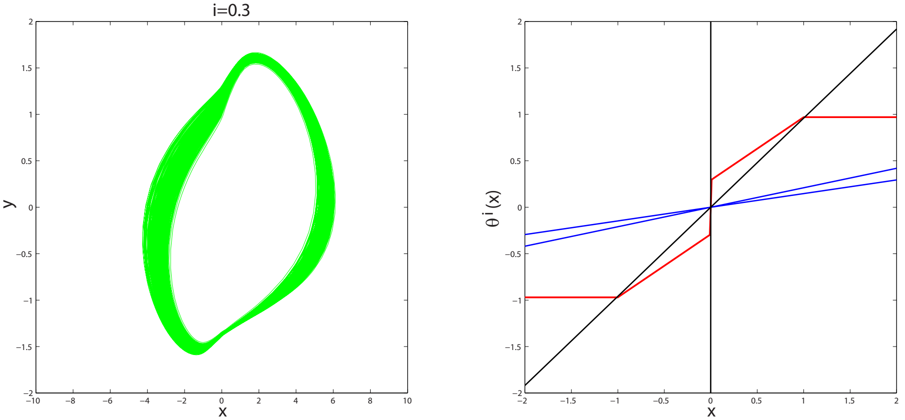


Fig. 4. Projection of the solutions of the modified Chua's system on the plane (x, y) for $i = 3$. Nonlinearity $\theta^i(x)$ and stability sectors.

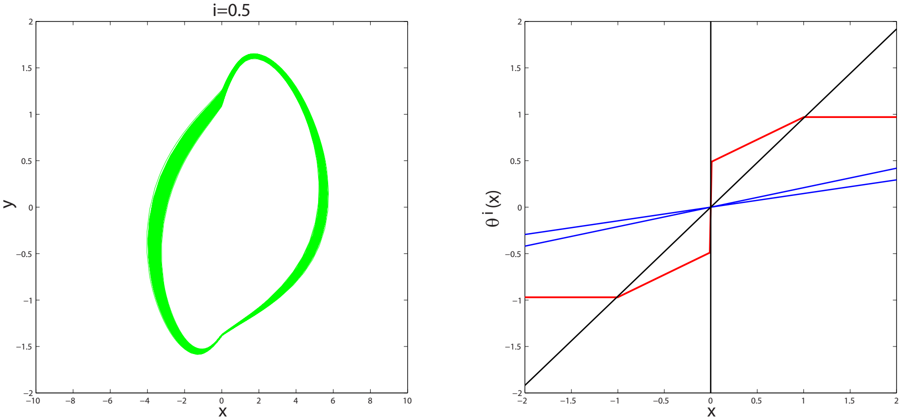


Fig. 5. Projection of the solutions of the modified Chua's system on the plane (x, y) for $i = 5$. Nonlinearity $\theta^i(x)$ and stability sectors.

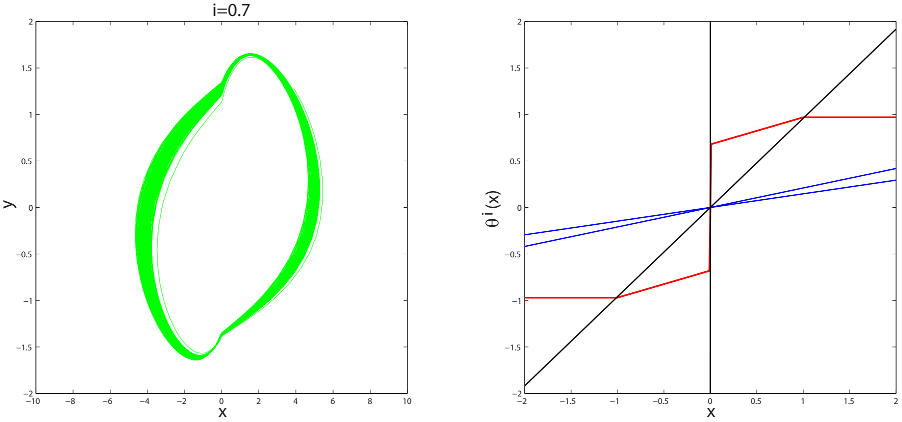


Fig. 6. Projection of the solutions of the modified Chua's system on the plane (x, y) for $i = 7$. Nonlinearity $\theta^i(x)$ and stability sectors.

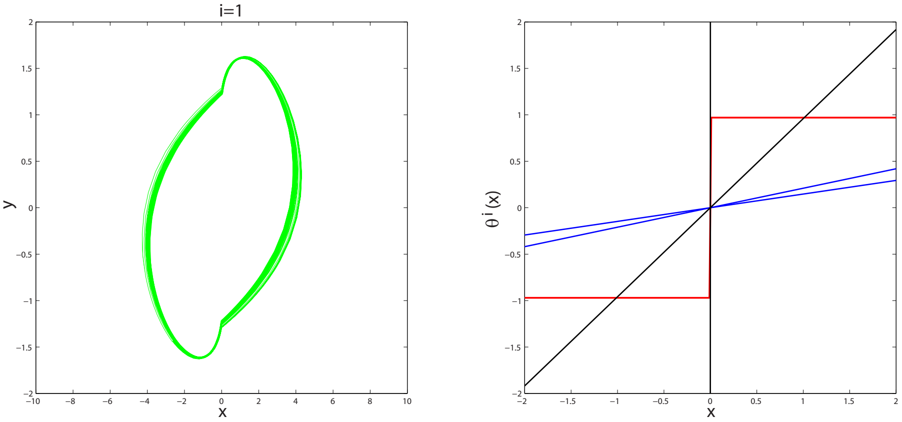


Fig. 7. Projection of the solutions of the modified Chua's system on the plane (x, y) for $i = 10$. Nonlinearity $\theta^i(x)$ and stability sectors.

4 Conclusions

In the present work the application of special analytical-numerical algorithm for hidden attractor localization is discussed. The existence of such hidden attractors in classical and modified Chua's circuits is demonstrated.

It is also can be noted that to obtain existence of hidden attractor in Chua's circuit one can artificially modify [20,19,9] diode characteristics to stabilize zero stationary point by inserting small stable zone around zero stationary point into nonlinearity.

Acknowledgements. This work was partly supported by grants from Academy of Finland, Ministry of Education and Science (Russia), RFBR, and Saint-Petersburg State University.

References

1. Lorenz, E.N.: Deterministic nonperiodic flow. J. Atmos. Sci. 20, 130–141 (1963)
2. Rossler, O.E.: An Equation for Continuous Chaos. Physics Letters 57A(5), 397–398 (1976)
3. Chua, L.O., Lin, G.N.: Canonical Realization of Chua's Circuit Family. IEEE Transactions on Circuits and Systems 37(4), 885–902 (1990)
4. Chen, G., Ueta, T.: Yet another chaotic attractor. Int. J. Bifurcation and Chaos. 9, 1465–1466 (1999)
5. Leonov, G.A., Kuznetsov, N.V., Vagaytsev, V.I.: Localization of hidden Chua's attractors. Physics Letters A 375(35), 2230–2233 (2011), doi:10.1016/j.physleta.2011.04.037
6. Leonov, G.A.: On the method of harmonic linearization. Automation and remote controle 70(5), 800–810 (2009)
7. Leonov, G.A.: On the Aizerman problem. Automation and Remote Control 70(7), 1120–1131 (2009)
8. Leonov, G.A.: Effective methods for periodic oscillations search in dynamical systems. App. Math. & Mech. 74(1), 24–50 (2010)
9. Leonov, G.A., Vagaitsev, V.I., Kuznetsov, N.V.: Algorithm for localizing Chua attractors based on the harmonic linearization method. Doklady Mathematics 82(1), 663–666 (2010)
10. Leonov, G.A., Bragin, V.O., Kuznetsov, N.V.: Algorithm for Constructing Counterexamples to the Kalman Problem. Doklady Mathematics 82(1), 540–542 (2010)
11. Kuznetsov, N.V., Leonov, G.A.: Lyapunov quantities, limit cycles and strange behavior of trajectories in two-dimensional quadratic systems. Journal of Vibroengineering 10(4), 460–467 (2008)
12. Leonov, G.A., Kuznetsov, N.V.: Limit cycles of quadratic systems with a perturbed weak focus of order 3 and a saddle equilibrium at infinity. Doklady Mathematics 82(2), 693–696 (2010)
13. Leonov, G.A., Kuznetsov, N.V., Kudryashova, E.V.: A Direct Method for Calculating Lyapunov Quantities of Two-Dimensional Dynamical Systems. In: Proceedings of the Steklov Institute Of Mathematics, vol. 272(suppl. 1), pp. S119–S127 (2011)
14. Chua, L.O.: A Zoo of Strange Attractors from the Canonical Chua's Circuits. In: Proceedings of the IEEE 35th Midwest Symposium on Circuits and Systems, vol. 2, pp. 916–926 (1992)
15. Chua, L.O.: A Glimpse of Nonlinear Phenomena from Chua's Oscillator. Philosophical Transactions: Physical Sciences and Engineering 353, 3–12 (1995)
16. Bilotta, E., Pantano, P.: A gallery of Chua attractors. World scientific series on nonlinear science (2008)
17. Shi, Z., Hong, S., Chen, K.: Experimental study on tracking the state of analog Chua's circuit with particle filter for chaos synchronization. Physics Letters A 372(34), 5575–5580 (2008)
18. Leonov, G.A., Kuznetsov, N.V.: Time-Varying Linearization and the Perron effects. International Journal of Bifurcation and Chaos 17(4), 1079–1107 (2007)
19. Savaci, F.A., Gunel, S.: Harmonic Balance Analysis of the Generalized Chua's Circuit. International Journal of Bifurcation and Chaos 16(8), 2325–2332 (2006)
20. Suykens, J.A.K., Huang, A., Chua, L.O.: A Family of n-scroll Attractors from a Generalized Chua's Circuit. AEU-International Journal of Electronics & Communications 51(3), 131–138 (1997)

Flexible Discrete Time State Space Model for Canal Pools

João Lemos Nabais[1] and Miguel Ayala Botto[2]

[1] IDMEC, Dept. of Systems and Informatics, Escola Superior de Tecnologia de Setúbal,
Campus do IPS, Estefanilha, 2910-761 Setúbal, Portugal
joao.nabais@estsetubal.ips.pt
[2] Instituto Superior Técnico, Technical University of Lisbon, Dept. of Mechanical Engineering,
CIS/IDMEC-LAETA, Av. Rovisco Pais, 1049-001 Lisboa, Portugal
ayalabotto@ist.utl.pt

Abstract. Water is a vital resource for mankind used in activities such as agriculture, industry and domestic activity. Irrigation is one of the most consuming water resources in human activity. Irrigation canals are characterized for being spatially distributed crossing different administrative regions. As water is becoming a scarce and valuable resource, efficient engineering water conveyance networks are required. In this paper a discrete state space for modeling open-channels is presented. The well known Saint-Venant equations are first linearized for a steady state and then discretized using the Preissmann scheme. The resulting model is shown to be computational simple and flexible to accommodate different type of boundary conditions, in flow, water depth or hydraulic structures dynamics, which are important features for modeling complex water conveyance systems. The hydraulic model also offers monitoring ability along the canal axis and can therefore be integrated in fault diagnosis and tolerant control strategies. The model is validated with experimental data from a real canal property of the Évora University.

Keywords: Modeling, Partial differential equations, Saint-venant equations, Open-channels, Water conveyance networks, Time delay system.

1 Introduction

Water is an essential resource for all life species, in particular human life. From agricultural to industrial applications or simple domestic activities, an efficient water conveyance network is a key factor for a sustainable development, social stability and welfare. Water can be distributed through natural irrigation canals provided by nature itself, like rivers, or be either transported by means of artificial irrigation canals generally known as water conveyance systems. These systems have usually great complexity from an automatic control point of view, since they are generally large spatially distributed systems with strong nonlinearities and physical constraints, time delays, while their operation typically requires the compatibility of multiple competing objectives. Therefore the need for an accurate dynamic hydraulic model that is sufficiently rich to incorporate the most relevant physical dynamics, while being flexible enough to be adapted to different operational setups.

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 159–171.
springerlink.com

For model base canal controller design is necessary to have a good model able to capture the main system dynamics. A simple analytical model was proposed by [1] the so-called integrator delay whose simplicity made it popular for canal modeling [2] [3]. Although being a simple model, controller design using this type of model is still a current research topic [4,5]. If more accuracy is needed then Saint Venant equations [6] are commonly used to model the dynamic behavior of the water flow in open water channels. The Saint-Venant equations consist of a pair of nonlinear hyperbolic partial differential equations. These equations are hard to be handled and so typically a linearized version around an equilibrium point is used for simulation and control purposes [7]. In [8] it is shown how a continuous multivariable dynamic model relating inflows to water depths for an open water pool is obtained. This model is specially suitable for H_∞ frequency analysis. Based on this structure, a simplified single-input single-output Integral Delay Zero (IDZ) model was shown to capture the main hydraulic dynamics [9]. However, although simple, the IDZ model lacks some accuracy [10]. A more flexible model framework that is computationally simple with efficient monitoring capacity of the canal water depths is then required.

This paper presents a discrete-time linear state space model for canal pools. The model is obtained through the discretization of the linearized Saint-Venant equations around a stationary point. The following interesting features can be found:

- a minimum computational effort is required for simulation purposes making it easily extendable to high dimensional water conveyance networks;
- monitoring water depths and flows along the canal can be easily made through an appropriate choice of the output equation;
- boundary conditions are easily integrated as flows or water depths, allowing for modular interconnection of different elements on a given canal;
- it enables the accommodation of hydraulic structure dynamics to which the pool is linked to;
- it can be easily integrated into model based control strategies [11] [12] opening the gate to its inclusion in fault tolerant control applications [13] [14] [15].

Besides, the proposed hydraulic model is further validated against real data retrieved from an experimental water delivery canal hold by the NuHCC – Hydraulics and Canal Control Center from the Évora University in Portugal.

The paper has the following structure. Section 2 presents the experimental canal. The canal model problem formulation is then presented in section 3 where the partial differential dynamic equations describing the transport phenomenon are first linearized and then discretized leading to a finite linear pool model. In section 4 a brief model numerical parameter analysis is presented. In section 5 the hydraulic model is validated with data retrieved from the experimental canal. Here it is shown the reliability and accuracy of the proposed hydraulic model. Finally, in section 6 some conclusions are drawn.

2 Experimental Canal

The experimental automatic canal [16] is located in Mitra near Évora, Portugal (Fig. 1). The canal has 4 pools with a trapezoidal cross section of 0.900m height and a total

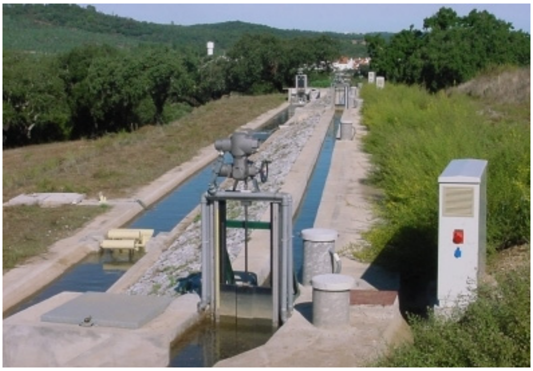


Fig. 1. NuHCC canal global view

Table 1. Experimental canal uniform parameters

Parameter	Pool 1	Pool 2	Pool 3	Pool 4
L [m]	40.7	35	35	35.2
S_0	0.0016	0.0014	0.0019	0.0004
n [m$^{-1/3}$s]	0.015	0.015	0.015	0.015
b [m]	0.15	0.15	0.15	0.15
m [m]	1:0.15	1:0.15	1:0.15	1:0.15

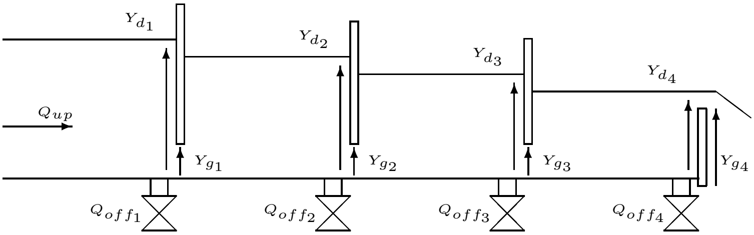


Fig. 2. Schematics of the complete facility

length of 145.9m. The geometric characteristics for each pool are shown in Table 1 where L means pool length, S_0 the bed slope, n the Manning friction coefficient, b the bottom width and m the side slope.

The 4 pools are divided by three sluice gates as shown in Fig. 2. All these sluice gates are electro-actuated and instrumented with position sensors. A rectangular overshot gate is located at the end of the canal with 0.38m width. The off-take valves, equipped with an electromagnetic flow-meter and motorized butterfly valve for flow control, are immediately located upstream of each sluice gate. Counterweight-float level sensors are distributed along the canal. At the head of the canal an electro-valve controls the canal inflow. This flow is extracted from a reservoir. The maximum flow capacity is 0.090 m^3/s. The flow within the automatic canal is regulated by another electro-valve located at the exit of a high reservoir (head of the automatic canal), simulating a real load situation. This high reservoir is filled with the recovered water pumped from a low one, which collects the flow from a traditional canal allowing a closed circuit. All electro-actuators and sensors in the canal are connected to local PLCs (Programmable Logic Controllers) responsible for the sensor data acquisition and for the control actions sent to the actuators. All local PLCs are connected through a MODBUS network (RS 485). The interaction with the Évora canal is done through 5 inputs (canal intake Q_{up} and 4 gate elevations Y_{g_i}), 4 outputs (downstream water level at each pool Y_{L_i}) and 4 disturbances Q_{off_i} (offtakes at each pool end) using a multi-platform controller interface [17].

3 Canal Pool Model

3.1 First Principles

The flow in open-channels is well described by the Saint-Venant equations,

$$\frac{\partial Q(x,t)}{\partial x} + B(x,t)\frac{\partial Y(x,t)}{\partial t} = 0 \tag{1}$$

$$\frac{\partial Q(x,t)}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2(x,t)}{A(x,t)}\right) + g \cdot A(x,t) \cdot (S_f(x,t) - S_0(x)) = 0 \tag{2}$$

where, $A(x,t)$ is the wetted cross section, $Q(x,t)$ is the water discharge, $Y(x,t)$ is the water depth, $B(x,t)$ is the wetted cross section top width, $S_f(x,t)$ is the friction slope, $S_0(x)$ is the bed slope, x and t are the independent variables. These equations are partial differential equations of hyperbolic type capable of describing the transport phenomenon. The mathematical dynamical model used is known for being able to capture the process physics namely: backwater, wave translation, wave attenuation and flow acceleration.

To solve partial differential equations it is necessary to know the initial condition along the canal axis and also two boundary conditions in time.

Initial Conditions. The flow can be classified according to the independent variables variations in time and space:

- uniform flow, when parameters do not vary along canal axis, nonuniform when parameters vary in space,
- steady flow, when parameters do not vary in time, and unsteady when parameters vary in time.

In this work the *Nonuniform Unsteady Flow* is assumed. One interesting situation is to consider gradually varied flow. This is characterized for steady conditions, which means $\frac{\partial}{\partial t} = 0$. In this case the Saint-Venant equations are reduced to Ordinary Differential Equation. If also uniform flow is to be imposed, no variations along canal axis, it is only necessary to solve $S_f(x,t) = S_0(x)$. The water depth found is also known as the normal depth Y_N. If a downstream water condition different from the normal depth is given then the water profiles presented in Fig. 3 result.

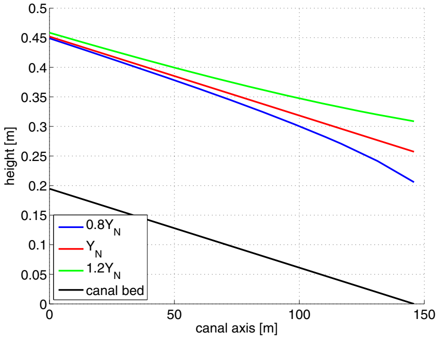


Fig. 3. Backwater for some downstream water depths with a nominal flow of $Q_0 = 0.020\text{m}^3$/s

Boundary Conditions. Partial differential equations of hyperbolic type are capable of describing the transport phenomenon. There are two waves present in the pool dynamics whose velocity are $V + C$ and $V - C$, where V is average velocity across section and C is the wave celerity. Depending on the relation between the dynamical and inertial velocity captured by the Froude number, $F_r = \frac{V}{C}$ the flow can be characterized into the following three types:

- subcritical: for $F_r < 1$ designated as fluvial and is typical of large water depths and small discharge and can be found at the river downstream;
- critical: for $F_r = 1$;
- supercritical: for $F_r > 1$ designated as torrential and is typical for small depth and large discharge and can be found at the river upstream.

In the subcritical case two waves traveling in opposite direction along the canal axis may occur. Because of this phenomenon, one boundary condition at each pool end is needed. In this work only subcritical flow is considered.

3.2 PDE Resolution

For solving numerically the partial differential equation it is required to proceed with time and space discretization. Here two approaches are valid [8]:

- Hydraulic approach: in this classical approach the equations are first discretized and then the nonlinear terms are approximated. This leads to time variant systems and requires the resolution of a set of algebraic equations, for instance through the generalized Newton method;
- Control approach: in this approach the equations are first linearized around a stationary configuration $(Q_0, Y_0(x))$. After this step the equations are discretized which allows for a time invariant state space representation.

Consider a steady state defined as $(Q_0, Y_0(x))$ where index 0 stands for steady flow configuration. The deviation variables are defined as,

$$\begin{aligned} q(x,t) &= Q(x,t) - Q_0 \\ y(x,t) &= Y(x,t) - Y_0(x) \end{aligned}$$

Assuming $A(x,t) = B_0(x)Y(x,t)$, after linearization equations (1) (2) become,

$$B_0(x)\frac{\partial y(x,t)}{\partial t} + \frac{\partial q(x,t)}{\partial x} = 0 \tag{3}$$

$$\frac{\partial q(x,t)}{\partial t} + 2V_0(x)\frac{\partial q(x,t)}{\partial x} + \delta(x)q(x,t) + \dots$$
$$+ \left[C_0^2(x) - V_0^2(x)\right] B_0(x)\frac{\partial y(x,t)}{\partial x} - \tilde{\gamma}(x)y(x,t) = 0 \tag{4}$$

where,

$$\begin{aligned}
C_0(x) &= \sqrt{g\frac{A_0(x)}{B_0(x)}} \\
\alpha(x) &= C_0(x) + V_0(x) \\
\beta(x) &= C_0(x) - V_0(x) \\
\delta(x) &= \frac{2\cdot g}{V_0(x)}\left(J_0(x) - Fr_0^2(x)\frac{dY_0(x)}{dx}\right) \\
Fr_0^2(x) &= \frac{V_0^2(x)B_0(x)}{g\cdot A_0(x)} \\
\tilde{\gamma}(x) &= V_0^2(x)\frac{dB_0(x)}{dx} + g\cdot B_0(x)\left[D(x)J_0(x) + I(x) - \left(1+2Fr_0^2(x)\right)\frac{dY_0(x)}{dx}\right] \\
D(x) &= \frac{7}{3} - \frac{4}{3}\frac{A_0(x)}{P_0(x)B_0(x)}\frac{\partial P_0(x)}{\partial y}
\end{aligned} \tag{5}$$

To complete the linearized pool model it is necessary to consider an initial condition along the spatial coordinate,

$$q(x,0) = q_0(x) = q_0 \qquad y(x,0) = y_0(x) \tag{6}$$

and two boundary conditions on each end along time,

$$q(0,t) = u_1(t) \qquad q(L,t) = u_2(t) \tag{7}$$

To simplify future analysis the Saint-Venant equations can be re-written into a more convenient alternative form. For that consider the area deviation as $a(x,t) = B_0(x)y(x,t)$. The linearized equations (3) and (4) are now given by,

$$\frac{\partial a(x,t)}{\partial t} + \frac{\partial q(x,t)}{\partial x} = 0 \tag{8}$$

$$\frac{\partial q(x,t)}{\partial t} + \left[\alpha(x) - \beta(x)\right]\frac{\partial q(x,t)}{\partial x} + \dots$$
$$+\alpha(x)\beta(x)\frac{\partial a(x,t)}{\partial x} + \delta(x)q(x,t) - \gamma(x)a(x,t) = 0 \tag{9}$$

where,

$$\gamma(x) = \frac{C_0^2(x)}{B_0(x)}\frac{dB_0(x)}{dx} + g\left[\left(1+D(x)\right)I(x) + \dots\right.$$
$$\left. - \left(1 + D(x) - (D(x)-2)Fr_0^2(x)\right)\frac{dY_0(x)}{dx}\right] \tag{10}$$

Considering the state vector $\chi(x,t) = \begin{bmatrix} q(x,t) \; a(x,t) \end{bmatrix}^T$ equations (8) and (9) may be expressed in state space form as follows,

$$A\frac{\partial}{\partial t}\chi(x,t) + B(x)\frac{\partial}{\partial x}\chi(x,t) + C(x)\chi(x,t) = 0 \tag{11}$$

where,

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad B(x) = \begin{bmatrix} 1 & 0 \\ \alpha(x) - \beta(x) & \alpha(x)\beta(x) \end{bmatrix} \quad C(x) = \begin{bmatrix} 0 & 0 \\ \delta(x) & -\gamma(x) \end{bmatrix} \tag{12}$$

3.3 Finite Dimension Model

The numerical method used to obtain a finite dimension model is the implicit method known as Preissmann Scheme. In this method Δx is the spatial mesh dimension, Δt is the time step, θ and ϕ weighting parameters ranging from 0 to 1. When using numerical methods it is important to be aware that they may introduce nonphysical behavior that is similar to the process physics and once introduced is not clear how to eliminate it [18].

The state vector for two consecutive sections is fourth dimension with both upstream and downstream flow and area deviation, $x(k) = \begin{bmatrix} q_i^k \; a_i^k \; q_{i+1}^k \; a_{i+1}^k \end{bmatrix}^T$, where index k stands for time and index i stands for space. Applying the Preissmann scheme to equation (11) after some manipulations the following discrete state space representation is obtained,

$$\begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \\ a_{13} & a_{23} \\ a_{14} & a_{24} \end{bmatrix}^T \begin{bmatrix} q_i^{k+1} \\ a_i^{k+1} \\ q_{i+1}^{k+1} \\ a_{i+1}^{k+1} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{21} \\ b_{12} & b_{22} \\ b_{13} & b_{23} \\ b_{14} & b_{24} \end{bmatrix}^T \begin{bmatrix} q_i^k \\ a_i^k \\ q_{i+1}^k \\ a_{i+1}^k \end{bmatrix} = 0 \tag{13}$$

The state space representation describes the pool dynamics between two adjacent sections. To obtain the model corresponding to a pool divided into N reaches it is necessary to use $N+1$ sections leading to $2(N+1)$ variables. Using model (13) is possible to obtain $2N$ equations. The last two equations are related to the upstream and downstream boundary conditions. The boundary conditions are imposed normally by the structure the pool is linked to. The boundary condition may be imposed in flow, usually when connected to gates, or in water depth, when the pool is connected to large reservoirs. A slightly more complex approach is when some hydraulic structure dynamics are to be incorporated into the model. The hydraulic structure linearized equation is used as a boundary condition. In this case a local linear model is constructed describing the pool plus gate dynamics. With this approach the entire canal dynamics can be obtained by means of connection local model dynamics, also called linear agents. These linear agents may be used for local model base control strategies.

In this work flow boundary conditions are assumed for constructing an open water canal simulator with the objective of validating the Saint-Venant resolution method proposed.

The total pool state vector $\mathbf{X}(k)$ is defined as,

$$\mathbf{X}(k) = \begin{bmatrix} q_1(k) \; a_1(k) \; q_2(k) \; a_2(k) \ldots q_n(k) \; a_n(k) \; q_{n+1}(k) \; a_{n+1}(k) \end{bmatrix} \tag{14}$$

Finally, the pool linear model representation can be given as,

$$\begin{aligned}\mathbf{X}(k+1) &= A\mathbf{X}(k) + B\mathbf{U}(k)\\ \mathbf{Y}(k) &= C\mathbf{X}(k)\end{aligned} \qquad (15)$$

where $\mathbf{U}(k)$ is the model input and $\mathbf{Y}(k)$ is the model output. It is important to emphasize some features of this model: i) the partial differential equations are solved only by matrices multiplications; ii) the number N of reaches inside a pool defines the number of sections $N+1$ and state space variables $2(N+1)$; and iii) all state space variables are accessible through matrix C in the output equation.

4 Parameter Analysis

The use of numerical methods for simulation may well introduce numerical oscillations and diffusion which, at the worst case, can lead to instability. Numerical methods are also known for introducing non physical dynamics which are similar to the process dynamics. After a finite dimension model is obtained, it is then crucial to proceed with parameters analysis. It is important to know how the nominal model performance is affected by the numerical parameters. This evaluation will be done for wave propagation along canal axis created by imposing a positive step input flow at the boundary condition. The analysis is made for the second canal pool. The nominal model was built with the following parameters: $L = 35\text{m}$ $N = 20$, $\varDelta x = \frac{L}{N}$, $\varDelta t \rightarrow C_r \approx 1$, $\phi = 0.5$ and $\theta = 0.5$, where C_r means the Courant number defined as,

$$C_r = \alpha \frac{\varDelta t}{\varDelta x} \qquad (16)$$

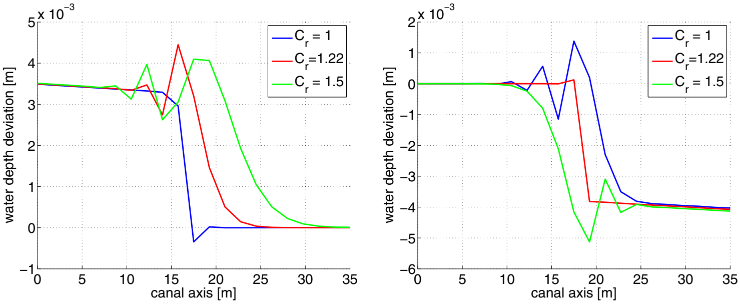
which can be seen as the ratio between numerical velocity and kinematic velocity.

4.1 Sample Time

The sample time is one of the grid dimension parameters. Reducing it means that the numerical solution is calculated faster than the dynamical velocity. As a consequence the Courant number is reduced. Different Courant numbers tested are, $C_r = \begin{bmatrix}1\ 1.22\ 1.5\end{bmatrix}$ or in time step $\varDelta t = \begin{bmatrix}0.835\ 1.02\ 1.25\end{bmatrix}$. In Fig. 4 two waves travelling along canal axis for a given time instant are shown. It is clear that the system exhibits nonphysical oscillations that are not damped by the sample time. Time step is not a tunable parameter. It must be chosen to keep the Courant number close to unity in order to have similar resolution in time and space. Contrary to what happens in continuous systems, reducing the time step does not improve the numerical solution.

4.2 Preissmann Parameters

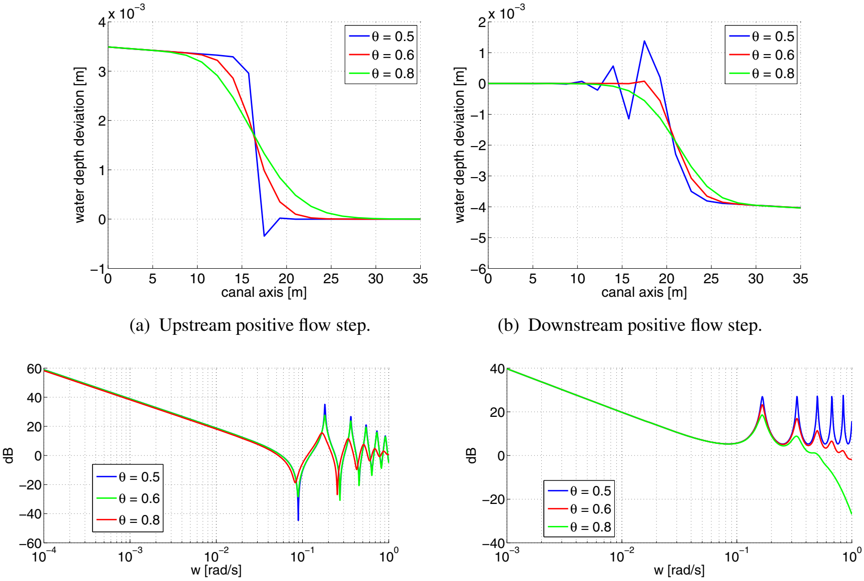
A centered approach in space is used, which means $\phi = 0.5$. Only the interpolation parameter in time θ is changed. The centered scheme is known to be unconditionally stable for $\theta \geq 0.5$. The following values were tested $\theta = \begin{bmatrix}0.5\ 0.6\ 0.8\end{bmatrix}$. In Fig. 5(a) 5(b)



(a) Upstream positive flow step.

(b) Downstream positive flow step.

Fig. 4. Wave propagation for different time step values



(a) Upstream positive flow step.

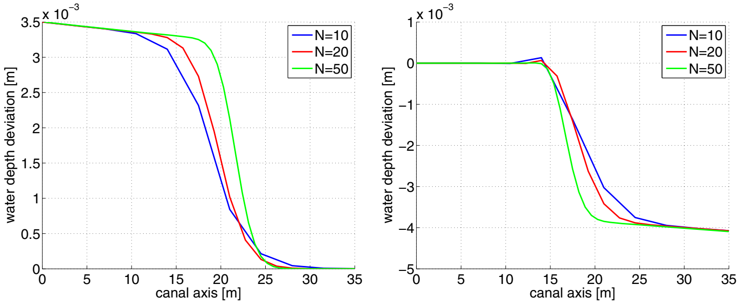
(b) Downstream positive flow step.

(c) Upstream inflow to upstream water depth.

(d) Upstream inflow to downstream water depth.

Fig. 5. Wave propagation for different θ values

the downstream and upstream wave propagation when a positive flow step is applied at the pool end is shown. The effect of increasing the θ parameter is similar: numerical oscillations are eliminated at the cost of introducing numerical diffusion. This interpretation can be confirmed in the frequency response represented in Fig. 5(c)-5(d) for the upstream flow input. The first natural frequency is kept almost unchanged while the higher frequencies are damped. Although these parameters allow for numerical oscillations elimination it may introduce too much diffusion in the model.



(a) Upstream positive flow step.

(b) Downstream positive flow step.

Fig. 6. Wave propagation for different N values

4.3 Space Step

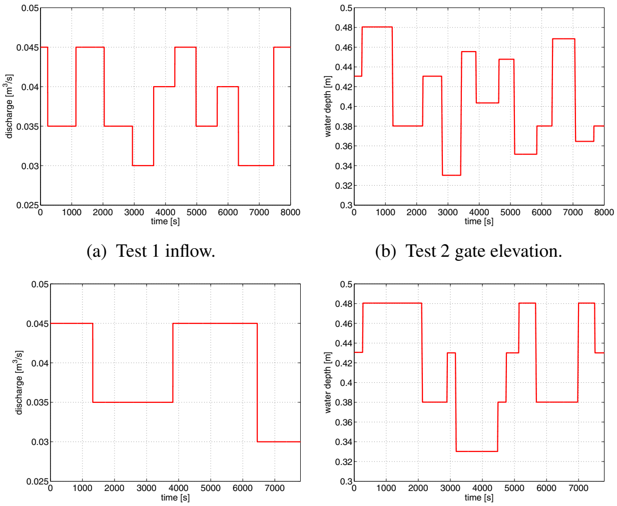
The space step is related to the number of reaches N considered in a pool. Assuming uniform space step parameter along canal axis it is practical to use $\Delta x = \frac{L}{N}$. If more resolution in the canal is desired this is the parameter to change, through the number of reaches. This is important as the space step is a constraint to the capacity of representing smaller waves as well as more abrupt changes in water profile. In Fig. 6 the downstream and upstream wave propagation when a positive flow step is applied at the pool end is shown. Establishing the N parameter is a tradeoff between accuracy and model complexity.

5 Model Validation

The model validation is done using data collected from the experimental canal. To emphasize the canal monitoring ability the canal configuration with the intermediate gates opened is used. In this case the canal is a single pool of 145.9m length. The initial condition used was $Q_0 = 0.045$m^3/s and $Y_0(L) = 0.595$m with the associated gate elevation $Y_g = 0.430$m. The linear pool model was constructed considering $N = 10$, $\theta = 0.6$ and a Courant number close to unity. In this configuration the Évora canal admits two inputs, a distant upstream flow and a local gate elevation.

Three different input scenarios were created for model validation, *test 1* – only upstream inflow, *test 2* – only local gate elevation and *test 3* – with both inputs. The scenarios run over approximately 8000s which is equivalent to 2hours and 15minutes. The input sequence was designed accordingly to the Évora canal specifications. For the inflow the interval $[0.030; 0.045]$m^3/s is tested, leading to a maximum deviation of 33% relative to Q_0. For the gate elevation the interval $[0.330; 0.480]$m is tested leading to a maximum deviation of 23% relative to $Y_0(L)$. The input sequences are represented in Fig. 7.

In Fig. 8 the system output – the downstream water depth – and three more water depths along the canal axis are shown, which proves the model canal axis monitoring ability. One is considering $x = 1/4L$, $x = 1/2L$ and $x = 3/4L$. In Table 2 the error



(a) Test 1 inflow.

(b) Test 2 gate elevation.

(c) Test 3 inflow.

(d) Test 3 gate elevation.

Fig. 7. Tests input sequences

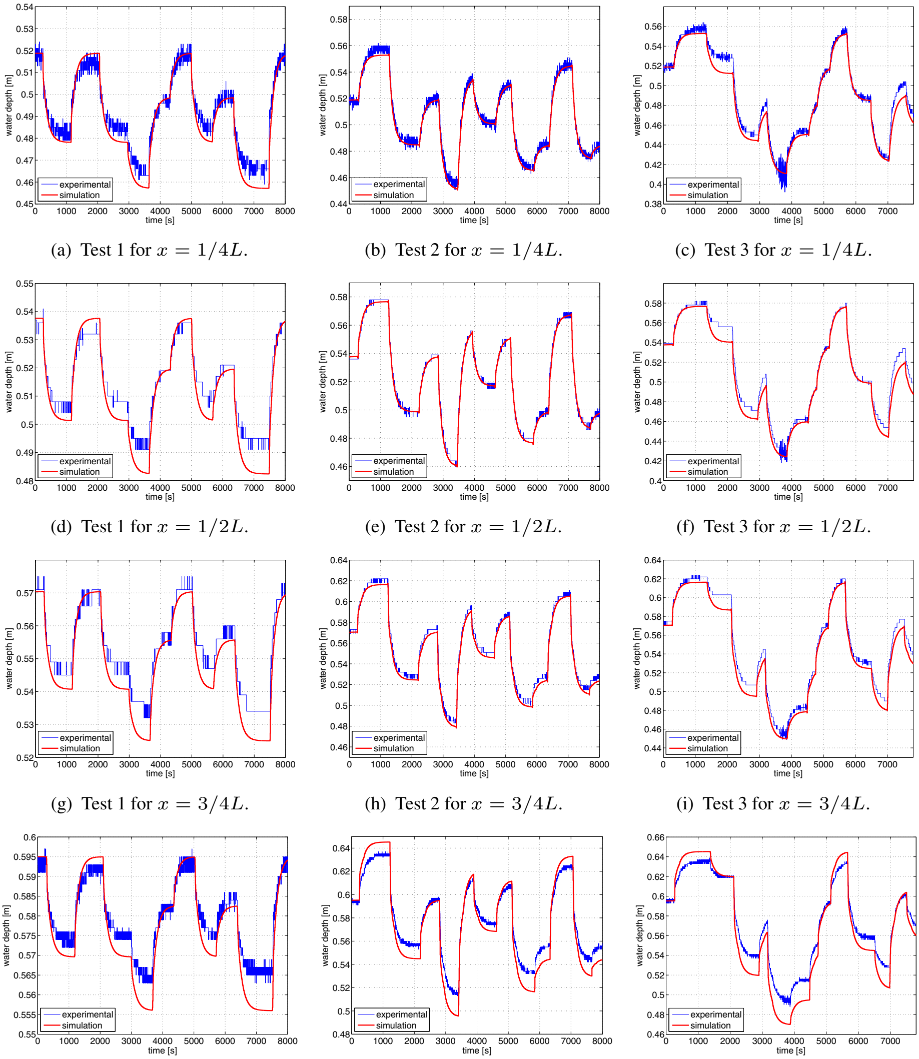
Table 2. Model performance for the different tests

Canal axis	VAF$_1$	VAF$_2$	VAF$_3$	MAE$_1$	MAE$_2$	MAE$_3$
$x = L$	80.36	91.14	91.97	0.0041	0.0092	0.0121
$x = 3/4L$	91.71	99.61	99.29	0.0066	0.0061	0.0097
$x = 1/2L$	87.43	99.58	98.50	0.0049	0.0019	0.0064
$x = 1/4L$	93.40	99.09	98.37	0.0053	0.0040	0.0074

criteria for downstream water depth as well for the intermediate points is presented. The error measurements used are the Variance Accounted For (VAF) and the Mean Absolute Error (MAE), the index refers to the test. The flow input causes small variation in downstream water depth while the opening gate is more severe. The lowest fit occurs at the downstream end, which can be explained by the experimental canal construction. The canal ends with a final reach of 7m length with rectangular section and 0.7m width. This is different from the nominal parameters considered and changes the downstream reservoir capacity. When travelling upstream the water depth tend to the normal depth which justifies the good model fit.

It is important to note that while in test 1 the water depth amplitude varies 0.030 m in test 3 due to the gate movement, a water depth amplitude variation of 0.170 m is observed, which is quite large when compared with the nominal downstream water depth.

A similar model validation was done for a model with $N = 30$ which means a space step of 5m but no increase in performance was obtained. However the computationally cost was severely increased.



(a) Test 1 for $x = 1/4L$.

(b) Test 2 for $x = 1/4L$.

(c) Test 3 for $x = 1/4L$.

(d) Test 1 for $x = 1/2L$.

(e) Test 2 for $x = 1/2L$.

(f) Test 3 for $x = 1/2L$.

(g) Test 1 for $x = 3/4L$.

(h) Test 2 for $x = 3/4L$.

(i) Test 3 for $x = 3/4L$.

(j) Test 1 for $x = L$.

(k) Test 2 for $x = L$.

(l) Test 3 for $x = L$.

Fig. 8. Water depths along the canal for tests 1,2 and 3

6 Conclusions

A finite dimension linear model for canal pools has been presented and validated with experimental data. The linearized partial differential equations describing the system are solved through matrices multiplications which requires low computational effort. This enables the model to be used for constructing water conveyance networks. The possibility to use the flow, water depth or linearized hydraulic structures as boundary conditions, augments the model applicability.

The proposed model also allows for full canal monitoring. This is an important feature that opens the scope of application to fault detection, isolation, and fault tolerant control algorithms.

Acknowledgements. This work was supported by the Portuguese Government, through Fundação para a Ciência e a Tecnologia, under the project PTDC/EEACRO/102102/2008 - AQUANET, through IDMEC under LAETA.

References

1. Schuurmans, J., Bosgra, O., Brouwer, R.: Open-channel flow model approximation for controller design. Applied Mathematical Modelling (1995)
2. Schuurmans, J., Hof, A., Dijkstra, S., Bosgra, O., Brouwer, R.: Simple water level controller for irrigation and drainage canals. Journal of Irrigation and Drainage Engineering 125, 189–195 (1999)
3. Schuurmans, J., Clemmens, J., Dijkstra, S., Hof, A., Brouwer, R.: Modeling of irrigation and drainage canals for controller design. Journal of Irrigation and Drainage Engineering 125, 338–344 (1999)
4. Overloop, P.J.: Model predictive control on open water systems. IOS Press (2006)
5. Negenborn, R., Overloop, P.J., Keviczky, T., Schutter, B.: Distributed model predictive control of irrigation canals. Networks and Heterogeneous Media 4, 359–380 (2009)
6. Akan, A.O.: Open channel hydraulics. Elsevier (2006)
7. Litrico, X., Fromion, V.: Infinite dimensional modelling of open-channel hydraulic systems for control purposes. In: 41th IEEE Conference on Decision and Control, Las Vegas, Nevada, pp. 1681–1686 (2002)
8. Litrico, X., Fromion, V.: Modeling and control of hydrosystems. Springer (2009)
9. Litrico, X., Fromion, V.: Simplified modeling of irrigation canals for controller design. Journal of Irrigation and Drainage Engineering 130, 373–383 (2004)
10. Nabais, J., Botto, M.: Qualitative comparison between two open water canal models. In: 9th Portuguese Conference on Automatic Control, Coimbra, Portugal, pp. 501–506 (2010)
11. Martinez, C.A.O.: Model predictive control of complex systems including fault tolerance capabilities: application to sewer networks. Technical University of Catalonia (2007)
12. Silva, P., Botto, M., Figueiredo, J.: Model predictive control of an experimental canal. In: European Control Conference, Kos, Greece, pp. 2977–2984 (2007)
13. Blanke, M., Kinnaert, M., Lunze, J., Staroswiecki, M.: Diagnosis and fault-tolerant control. Springer (2006)
14. Isermann, R.: Fault-diagnosis systems. Springer (2006)
15. Bedjaoui, N., Weyer, E., Bastin, G.: Methods for the localization of a leak in open water channels. Networks and Heterogeneous Media 4, 180–210 (2009)
16. Rijo, R.: Local automatic control modes in an experimental canal. Irrigation and Drainage Systems 17, 179–193 (2003)
17. Duarte, J., Rato, L., Shirley, P., Rijo, M.: Multi-Platform controller interface for SCADA application. In: IFAC World Congress, Milan, Italy (2011)
18. Szymkiewicz, R.: Numerical modeling in open channel. Springer (2010)

Part IV
Industrial Engineering, Production and Management

Virtual Sensors for Semiconductor Manufacturing: A Nonparametric Approach - Exploiting Information Theoretic Learning and Kernel Machines

Andrea Schirru[1], Simone Pampuri[1,2], Cristina De Luca[2], and Giuseppe De Nicolao[1]

[1] Department of Computer Science Engineering, University of Pavia, Pavia, Italy
{andrea.schirru,simone.pampuri,giuseppe.denicolao}@unipv.it
[2] Infineon Technologies Austria, Villach, Austria
cristina.deluca@infineon.com

Abstract. In this paper, a novel learning methodology is presented and discussed with reference to the application of virtual sensors in the semiconductor manufacturing environment. Density estimation techniques are used jointly with Renyi's quadratic entropy to define a loss function for the learning problem, relying on Information Theoretic Learning concepts. Furthermore, Reproducing Kernel Hilbert Spaces (RKHS) theory is employed to handle nonlinearities and include regularization capabilities in the model. The proposed algorithm allows to estimate the structure of the predictive model, as well as the associated probabilistic uncertainty, in a nonparametric fashion. The methodology is then validated using simulation studies and process data from the semiconductor manufacturing industry. The proposed approach proves to be especially effective in strongly nongaussian environments and presents notable outlier filtering capabilities.

Keywords: Semiconductors, Machine learning, Entropy, Kernel methods.

1 Introduction

Virtual sensors are employed in many industrial settings to predict the result of an operation (most often a measurement) when the implementation of an actual sensor would be uneconomic or impossible [11]. In general, a virtual sensor finds and exploits a relation between some easily collectible variables (input) and one or more *target* (output) variables. Virtual sensor modeling techniques range from purely physics-based approaches [8] to machine learning and statistical methodologies [14]. This paper is motivated by a specific class of virtual sensors used in semiconductor manufacturing, namely Virtual Metrology (VM) tools. The measurement operations on processed silicon wafers are particularly time-consuming and cost-intensive: therefore, only a small subset of the production is actually evaluated [15]. Conversely, Virtual Metrology tools are able to predict metrology results at process time for every wafer, relying only on process data: such predictions are expected to reduce the need for actual measurement operations and, at the same time, establish positive interactions with metrology-related equipment tools (such as Run-to-Run controllers and decision aiding tools).

A Virtual Metrology tool is expected to **(i)** find and exploit complex, nonlinear relations between process data and metrology results, and **(ii)** assess prediction uncertainty

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 175–193.
springerlink.com

in a meaningful way; in order to achieve such goals, it is key to make the right assumptions on the observed data. Remarkably, a precise characterization of the process variability is in general hard to obtain: for instance, the observed data might be distributed according to fat-tailed or strongly non-Gaussian distributions, be affected by outliers or present signs of multimodality; it is to note that such difficulties are shared among many disciplines [1]. It is intuitive that suboptimal assumptions are likely to result in ineffective predictive models. In this paper, we present a novel methodology, inspired by Information Theoretic Learning theory [9], to tackle such an issue employing a regularized Reproducing Kernel Hilbert Space (RKHS) framework jointly with nonparametric density estimation techniques. The proposed approach is able to simultaneously estimate nonlinear predictive models and the associated prediction uncertainty, enabling the delivery of probabilistic predictions. The paper is structured as follows:

- Section 2 introduces the needed elements of machine learning and Kernel methods
- Section 3 presents and justifies the proposed approach from a theoretical point of view
- Section 4 tests the proposed methodology against simulation studies and data from the semiconductor manufacturing environment

Appendix A is devoted to numerical techniques used to solve the proposed problem, while Appendix B contains mathematical proofs.

2 Machine Learning and Kernel Methods

The goal of a learning task is to estimate, from data, a relationship between an input space $\mathcal{X}$ and an output space $\mathcal{Y}$. In order to achieve such result, it is necessary to rely on a set of observations $\mathcal{S} = \{x_i \in \mathcal{X}, y_i \in \mathcal{Y}, i = 1, \ldots, N\}$. In other words, the goal is to find a map $f : \mathcal{X} \to \mathcal{Y}$ such that, given a new observation $\{x_{new} \in \mathcal{X}, y_{new} \in \mathcal{Y}\}$, $f(x_{new})$ will adequately predict y_{new}. In this framework, $\mathcal{S}$ is called a *training set* and the function f is an *estimator*. In the following, let f depend on a set of parameters θ, such that $f(x) := f(x;\theta)$; the optimization of θ with respect to some suitable criterion (function of $\mathcal{S}$ and θ) leads to the creation of a predictive model.

2.1 Regularized Machine Learning

In this paper, a regularized machine learning setting is employed to introduce and test the proposed methodology: the estimator f is found by minimizing some *loss function* $\mathcal{J}(\theta)$ with respect to θ. Such loss function is usually the sum of a *loss term* $\mathcal{L}$ and a *regularization term* $\mathcal{R}$, so that

$$\mathcal{J}(\theta) = \mathcal{L}(\theta) + \lambda\mathcal{R}(\theta) \tag{1}$$

In this framework, given a model specified by θ, $\mathcal{L}$ measures the quality of approximation on the training set $\mathcal{S}$ and $\mathcal{R}$ is a measure of the complexity of the model. Intuitively, the coexistence of $\mathcal{L}$ and $\mathcal{R}$ relates to a tradeoff between model regularity and performances on $\mathcal{S}$. The *regularization parameter* $\lambda \in \mathbb{R}^+$ acts as a tuning knob for such tradeoff: as λ grows, the order of the selected model gets lower and lower. In this paradigm, a learning algorithm is entirely specified by **(i)** the loss term $\mathcal{L}(\theta)$, **(ii)** the regularization term $\mathcal{R}(\theta)$ and **(iii)** the estimator structure $f(x;\theta)$. Remarkably, this structure assumes that the prediction of a generic y_i can be obtained, at best, up to a random uncertainty (depending on $\mathcal{L}$). In other words, adopting an additive error paradigm, it is implied that

$$y_i = f(x_i) + \epsilon_i$$

where ϵ_i is a random variable whose distribution depends on $\mathcal{L}$.

In the following, let $\mathcal{X} \equiv \mathbb{R}^p$ and, with no loss of generality, $\mathcal{Y} \equiv \mathbb{R}$. The goal is to build a map $f : \mathbb{R}^p \rightarrow \mathbb{R}$ of the relationship between an input dataset $X \in \mathbb{R}^{N \times p}$ and an array of target observations $Y \in \mathbb{R}^N$. Furthermore, let x_i be the i-th row of X, and y_i be the i-th entry of Y.

2.2 Linear Predictive Models

Perhaps the most notable example of estimation technique is the method of Least Squares, that can be traced back to Gauss and Legendre. Such methodology assumes a linear relationship between the input and output spaces, so that

$$f(x_i) := f(x_i; w) = x_i w \tag{2}$$

where w is a p-variate vector of parameters. Furthermore, let $\mathcal{L}$ be the sum of squared residuals

$$\mathcal{L}(w) = \sum_{i=1}^{N} (y_i - f(x_i))^2 \tag{3}$$

and let $\mathcal{R}(w) \equiv 0$. The optimal w^* (minimizer of $\mathcal{L}(w)$) is then

$$w^* = (X'X)^{-1} X'Y$$

When a new input observation x_{new} is available, the optimal least squares prediction of y_{new} is

$$\hat{y}_{new} = E[y_{new} | x_{new}] = x_{new} w^*$$

Equation (3) implies a Gaussian distributed ϵ_i with

$$\epsilon_i \sim N(0, \sigma^2)$$

where σ^2 is the variance of the observation uncertainty. Notably, $\hat{y}_{new}$ is independent of σ^2: it is necessary to tune the variance term only if a probabilistic output is needed (such as prediction confidence intervals). Least squares is a simple yet powerful method that suffers from two main drawbacks, namely **(i)** overfitting in high-dimensional spaces (p close to N) and **(ii)** possible ill-conditioning of the matrix $X'X$. In order to overcome such issues, a regularization term is employed: by using (2) and (3), and letting

$$\mathcal{R}(w) = \sum_{i=1}^{N} w_i^2$$

Ridge Regression is obtained. More and more stable (low sum of squared coefficients) models are selected as λ grows, at the cost of worsening the performances on the training set. The idea behind Ridge Regression is that the optimal λ allows to build a predictor that includes all and only the relevant information. The optimal Ridge Regression coefficient vector is

$$w^* = (X'X + \lambda I)^{-1} X'Y$$

Similarly to least squares, it is not necessary to explicitly address the tuning of the error variance σ^2 unless a probabilistic output is needed.

2.3 Nonlinear Predictive Models

It is apparent that (2) defines a linear relationship between the p-variate input space and the output space. In a wide variety of applications, however, a linear model is not complex enough to obtain the desired prediction performances. An unsophisticated approach would be to adopt an *expanded basis* (augmenting the input set X with nonlinear functions of its columns - for instance, polynomials) to tackle such issue. It is to note, however, that this simple approach would yield computationally intractable problems also for a relatively small values of p [5]: nonlinearities are more efficiently handled using kernel-based methodologies. In the case of Ridge Regression, consider a symmetric positive definite matrix $K \in \mathbb{R}^{N \times N}$ whose entries arise from a suitable positive definite inner product $\mathcal{K}$ (kernel function), such that

$$K_{ij} = \mathcal{K}(x_i, x_j) \tag{4}$$

Furthermore, consider the model structure

$$f(x_i) = K_i c \tag{5}$$

where K_i is the i-th row of K, and the regularization term

$$\mathcal{R}(c) = c' K c$$

In this framework, $c \in \mathbb{R}^N$ is the coefficient vector of the so-called *dual form* of the learning problem, and $\mathcal{R}$ is the norm of f in a nonlinear Hilbert space. The resulting model f exploits a nonlinear relationship (specified by $\mathcal{K}$) between X and Y. This result arises from RKHS (Reproducing Kernel Hilbert Spaces) theory and Riesz Representation Theorem: the kernel function $\mathcal{K}$ is used to establish a relationship between the p features and the N examples. Among the most popular kernel functions, the inhomogeneous polynomial kernel

$$\mathcal{K}(x_i, x_j; d) = (x_i x_j' + 1)^d$$

incorporates the polynomial span of X up to the d-th grade, and the exponential kernel

$$\mathcal{K}(x_i, x_j; \xi) = e^{-\frac{||x_i - x_j||^2}{\xi^2}}$$

relates to an infinite-dimensional feature space whose bandwidth is controlled by ξ^2. The optimal Kernel Ridge Regression coefficient vector is

$$c^* = (K + \lambda I)^{-1} Y$$

and the predictor is

$$\hat{y}_{new} = k_{new} c^*$$

with $k_{new} = [\mathcal{K}(x_{new}, x_1) \ldots \mathcal{K}(x_{new}, x_N)]$. A thorough review of kernel-based methodologies is beyond the scope of this section: the interested reader can find more information in [12].

2.4 Learning in Nongaussian Settings

It is to note that the methodologies reviewed in this section rely on Gaussian assumptions: the probabilistic interpretation of the loss function (3) is that y_{new} can be predicted, at best, with an additive Gaussian-distributed uncertainty with fixed variance. The reasons for adopting such an assumption are both historical (linking to the concept of Least Squares estimation) and methodological (Central Limit Theorem and closed form solution), but other choices are possible. For instance, the Huber loss function (used in robust statistics) implies a Gaussian distribution near the origin with Laplace tails and allows to reduce the weight of outliers in the learning process. Another notable example is the ϵ-insensitive loss function, that relates to a uniform distribution between $[-\epsilon, \epsilon]$ with Laplace tails, and is mainly used in Support Vector Machines (SVM).

Remarkably, all the loss terms described in this section rely on parametric assumptions: the uncertainty is assumed to follow a specific (known) distribution depending on a set of unknown parameters. In a real setting, however, it is often not possible (and sometimes not even desirable) to identify the uncertainty distribution in a parametric way: in such situations, a more flexible characterization is needed to achieve the best performances. In the next section, a learning method that achieves such flexibility is presented using Density Estimation techniques jointly with Entropy-related criteria.

3 Regularized Entropy Learning

In this section, an entropy-based learning technique that makes no assumptions about the uncertainty distribution is presented and discussed. The novel methodology will be referred to as ”Regularized Entropy Learning”.

3.1 Density Estimation and Learning

Consider a real-valued array $\epsilon = [\epsilon_1, \ldots, \epsilon_N]'$, where every ϵ_i is assumed to be independently drawn from the same unknown distribution. In order to obtain a nonparametric estimate of the probability density of ϵ, it is convenient to resort to *density estimation* techniques [7].

Remark: it would be more correct to use the term "*kernel* density estimation" (KDE). In order to avoid confusion (the word "kernel" has different meaning in KDE and Kernel Methods), KDE will be referred to as *density estimation* (DE).

DE techniques are able to estimate a probability density from a set of observations, using a mixture of predetermined distributions. Given the vector ϵ, the underlying distribution is estimated as

$$p_\epsilon(x) = \frac{1}{N} \sum_{i=1}^{N} g(\epsilon_i; x) \tag{6}$$

where $g(\cdot; x)$ is a nonnegative function such that

$$\int_{-\infty}^{+\infty} g(\cdot; x) dx = 1$$

It is immediate to prove that (6) is a probability distribution. In this paper, we employ the Gaussian density

$$G(\mu, \sigma^2; x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\dfrac{(x-\mu)^2}{2\sigma^2}} \tag{7}$$

so that $g(z; x) := G(z, \sigma^2; x)$. Hereby σ^2 is the *bandwidth* of the estimator, related to the smoothness of the estimated density: its tuning will be discussed in a later subsection. The density p_ϵ is rewritten as

$$p_\epsilon(x) = \frac{1}{N} \sum_{i=1}^{N} G(\epsilon_i, \sigma^2; x) \tag{8}$$

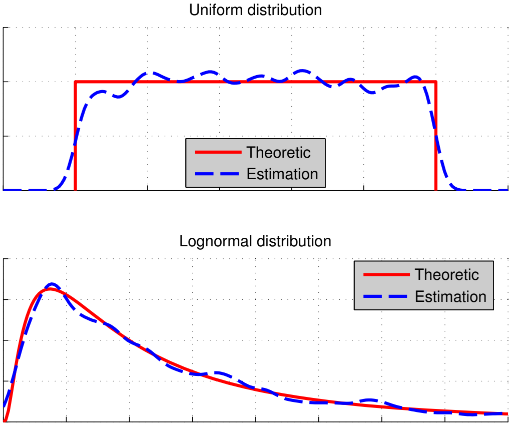
that is, a Gaussian density of variance σ^2 is centered on every observation ϵ_i. With reference to the learning setting presented in the previous section, let ϵ_i be the estimation error (residual) on the i-th sample of $\mathcal{S}$, for some value of c:

$$\epsilon_i := \epsilon_i(c) = y_i - K_i c \tag{9}$$

where K_i is the i-th row of the kernel matrix K. In the next subsection, (8) and (9) are used to define a loss term related to the concept of information entropy.

3.2 Entropy-Based Loss Term

In information theory, entropy is a measure of the uncertainty of a random variable: while an high entropy is associated to chaos and disorder, a quiet and predictable random variable is characterized by low entropy [4]. Notably, by minimizing the entropy of a random variable, a constraint is imposed on all its moments [3]. For this reason, the definition of an entropy-based loss term is desirable with respect to the Least Squares loss term, that involves only the second moment (variance). More interesting properties of such a loss term are investigated in [10].


Uniform distribution

Lognormal distribution

Fig. 1. Density estimation of uniform and lognormal distributions, using Gaussian densities

Shannon's entropy, perhaps the most notable entropy measure, is defined as the expected value of the information contained in a message. Renyi's entropy generalizes this concept to a family of functions depending on a parameter $\alpha \geq 0$. Consider a continuous random variable ϵ; its Renyi's entropy $H_\alpha(\epsilon)$ is

$$H_\alpha(\epsilon) = \frac{1}{1-\alpha} \log \int_{-\infty}^{+\infty} p_\epsilon(x)^\alpha dx \tag{10}$$

We consider the quadratic Renyi's entropy $H_2(\cdot)$ of the random variable $\epsilon|c$, as

$$H_2(\epsilon(c)) = -\log \int_{-\infty}^{+\infty} p_{\epsilon|c}(x)^2 dx \tag{11}$$

It is easily noted that $H_2(\epsilon)$ reaches its infimum when $p_\epsilon(x)$ is a Dirac Delta (complete predictability), and its supremum when $p_\epsilon(x)$ is flat over $\mathbb{R}$ (complete uncertainty). In order to define the desired loss term, we consider the following

Theorem 1. Let $A \in \mathbb{R}^{s\times s}$, $a \in \mathbb{R}^s$, $B \in \mathbb{R}^{t\times t}$, $b \in \mathbb{R}^t$ and $Q \in \mathbb{R}^{s\times t}$. Let $\mathbf{x} \in \mathbb{R}^t$ be an input variable. It holds that

$$G(a, A; Q\mathbf{x})G(b, B; \mathbf{x}) = G(a, A + QBQ'; b)G(d, D; \mathbf{x})$$

with

$$\begin{aligned} D &= (Q'A^{-1}Q + B^{-1})^{-1} \\ d &= b + DQ'A^{-1}(a - Qb) \end{aligned}$$

□

It is possible to express $H_2(\epsilon)$ in function of a weighted sum of Gaussian densities: this result is summarized in the following

Proposition 1. Applying Theorem 1 [6] and using (8) and (11), it holds that

$$H_2(\epsilon(c)) = -\log\left(\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G(y_i - y_j, 2\sigma^2; (K_i - K_j)c)\right)$$

Exploiting the symmetry of the Gaussian density, we define

$$\mathcal{H}(c) := \frac{2}{N^2}\sum_{i=1}^{N}\sum_{j=i+1}^{N} G(y_i - y_j, 2\sigma^2; (K_i - K_j)c)$$

and observe that $\mathcal{H}(c)$ is equal to $e^{-H_2(\epsilon(c))}$ up to an additive constant. Since the exponential transformation is monotonic,

$$\arg\max_c \mathcal{H}(c) = \arg\min_c H_2(\epsilon(c)) \tag{12}$$

□

Equation (12) states that a minimum entropy estimator can be obtained by maximizing a mixture of Gaussian densities with respect to the parameters vector c. In the following, since ϵ is entirely specified by c, we let $H_2(c) := H_2(\epsilon(c))$.

3.3 Regularized Entropy Learning

In this section, we consider the properties of the learning algorithm for which

$$\begin{aligned}\mathcal{L}(c) &= H_2(c)\\ \mathcal{R}(c) &= c'Kc\\ f(y_i) &= k_i c\end{aligned}$$

The novelty of the proposed approach lies in the RKHS regularization of an entropy-related loss term. Consider the following

Proposition 2. Given the loss function

$$\mathcal{J}(c) = H_2(c) + \lambda c'Kc \tag{13}$$

it holds that

$$e^{-\mathcal{J}(c)} \propto \mathcal{H}(c)G\left(0_N, \frac{K^{-1}}{\lambda}; c\right) \tag{14}$$

that is, applying an exponential transformation to $\mathcal{J}(c)$, it is possible to write it as the product between a weighted sum of Gaussian densities ($\mathcal{H}(c)$) and a Gaussian density dependent on λ. □

Furthermore, it has to be considered that $H_2(c)$ is shift-invariant: this result is discussed in the following

Proposition 3. Let $\epsilon(c)$ be a real valued vector of residuals associated to a coefficient vector c, and let $\epsilon(c^*) = \epsilon(c) + z$, where z is a real constant. It holds that

$$H_2(c) \equiv H_2(c^*)$$

□

Following Proposition 3, the expected value of the residuals represents an additional degree of freedom to be set in advance. Without loss of generality we choose to ensure that, given a random variable γ,

$$(p(\gamma|c) = p_\epsilon(x)) \rightarrow (E[\gamma] = 0) \tag{15}$$

According to Proposition 2, it is possible to write $\mathcal{J}(c)$, upon a monotonic transformation, as a sum of products of Gaussian densities. In order to define an efficient minimization strategy for $\mathcal{J}$, we consider the following

Proposition 4. It holds that

$$e^{-\mathcal{J}(c)} \propto \sum_{i=1}^{N} \sum_{j=i+1}^{N} \alpha_{ij} G(d_{ij}, D_{ij}; c) \tag{16}$$

with

$$\begin{aligned}
\alpha_{ij} &= G(y_i, 2\sigma^2 + \frac{K_{ii} - 2K_{ij} + K_{jj}}{\lambda}; y_j) \\
D_{ij} &= \left(\frac{(K_i - K_j)'(K_i - K_j)}{2\sigma^2} + \lambda K \right)^{-1} \\
d_{ij} &= D_{ij} \frac{(K_i - K_j)'(y_i - y_j)}{2\sigma^2}
\end{aligned}$$

where K_{st} is the $\{s, t\}$ entry of K, and therefore

$$\mathcal{J}(c) = -\log \left(\sum_{i=1}^{N} \sum_{j=i+1}^{N} \alpha_{ij} G(d_{ij}, D_{ij}; c) \right)$$

up to an additive constant. □

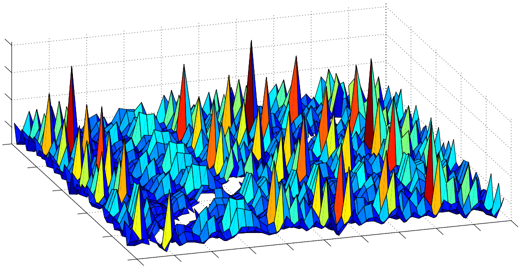


Fig. 2. Graphical representation of the matrix $\{\alpha_{ij}\}$ (surface plot)

Proposition 4 straightforwardly applies Theorem 1 to state that it is possible to write $\mathcal{J}(c)$ as the logarithm of a weighted sum of Gaussian densities. The multiplicative coefficients α_{ij} admit an interesting interpretation: α_{ij}, for which

$$0 \leq \alpha_{ij} \leq G(0, 2\sigma^2; 0) \tag{17}$$

gets monotonically closer to its supremum as two conditions are met: **(i)** y_i is close to y_j and **(ii)** $K_{ii} + K_{ij}$ is close to $2K_{ij}$. Using the definition of $K_{ij} = \mathcal{K}(x_i, x_j)$, it is immediately verified that condition **(ii)** occurs when x_i is close to x_j. Therefore, the multiplicative coefficient α_{ij} relates to the *information consistency* between the i-th and j-th sample: in other words, it is a measure of the similarity between the i-th and j-th observations. This allows for two interesting properties: **(i)** given a training set $\mathcal{S}$, it is possible to identify the most consistently informative pairs of examples (Figure 2). This information can be subsequently used, for instance, as a pruning criterion to obtain a minimal representative dataset. Furthermore, **(ii)** it is possible to use $\{\alpha_{ij}\}$ to discover mixtures in $\mathcal{S}$: indeed, if it is possible to identify two sets $\mathcal{S}_1 \subset \mathcal{S}$ and $\mathcal{S}_2 \subset \mathcal{S}$ such that, for all $(i, j \in \mathcal{S}_1)$ and $(k, z \in \mathcal{S}_2)$,

$$\alpha_{ij} \gg \alpha_{ik}$$
$$\alpha_{kz} \gg \alpha_{ik}$$

the information conveyed by $\mathcal{S}_1$ and $\mathcal{S}_2$ are significantly decoupled. Figure 3 depicts a colormap of $\{\alpha_{ij}\}$ for a toy dataset with $N = 30$, obtained by concatenating two decoupled sets of observations. As expected, the upper-left and lower-right $15x15$ submatrices show the highest values of α_{ij}.

3.4 Model Estimation

In the previous subsection, we have shown that the loss function of the proposed method is monotonically related to a weighted sum of N-variate Gaussian densities. In this subsection, an optimal (entropy-wise) regularized estimator of c is derived and employed to build a predictor for new observations. In order to obtain c^*, it is necessary to solve the following

Problem 1 (Minimization of $\mathcal{J}(c)$): find

$$c^* = \arg\min_c e^{\mathcal{J}(c)}$$

with

$$\mathcal{J}(c) = -\log\left(\sum_{i=1}^{N}\sum_{j=i+1}^{N} \alpha_{ij} G(d_{ij}, D_{ij}; c)\right)$$

Since the exponential transformation is monotonic, the minimizer of $e^{\mathcal{J}(c)}$ minimizes also $\mathcal{J}(c)$; the exponential formulation yields, however, simpler derivatives. Implementation details about the solution of Problem 1 are reported in Appendix A. The

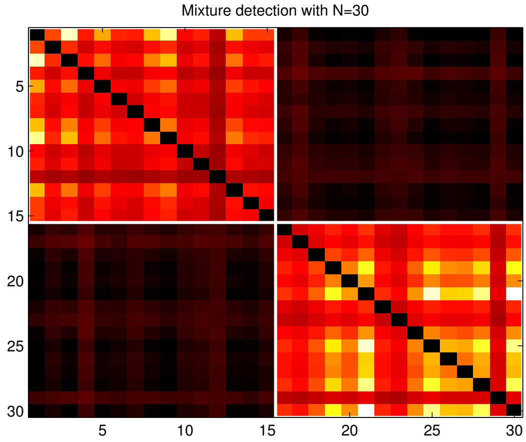


Fig. 3. Mixture recognition capabilities of the coefficients α_{ij}: the bright areas are self-consistent groups of homogeneous observations

estimate c^* represents a compromise between the RKHS norm of f, $\mathcal{R}(c)$, and Renyi's second order entropy of the estimation errors, $H_2(c)$. Additionally, $H_2(c^*)$ is the minimum reachable entropy configuration for the dataset $\mathcal{S}$ for a given value of λ. As c^* is obtained by solving Problem 1, it is necessary to set the additional degree of freedom discussed in Proposition 3: the bias term $\mathcal{B}$ is computed as

$$\mathcal{B} := \text{mean}(\{\epsilon_i\}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - K_i c^*)$$

It is then possible to compute predictions whenever a new observation x_{new} is available. Using the real-valued array

$$k_{new} = [\mathcal{K}(\tilde{x}_{new}, x_1) \ldots \mathcal{K}(\tilde{x}_{new}, x_N)]$$

the estimator $\hat{y}_{new}$ is

$$\hat{y}_{new} = E[y_{new}|x_{new}] = k_{new}c^* \tag{18}$$

Furthermore, the prediction uncertainty γ can be estimated using a Leave-One-Out approach: let

$$p(\gamma) = \frac{1}{N}\sum_{i=1}^{N} G(y_i - K_i c^*_{(i)} - \mathcal{B}_{(i)}, \sigma^2; x) \tag{19}$$

where $c^*_{(i)}$ and $\mathcal{B}_{(i)}$ solve Problem 1 using the reduced dataset $\mathcal{S}_{(i)} = \mathcal{S}\backslash\{x_i, y_i\}$. The probabilistic form of the predictor is then

$$y_{new} = \hat{y}_{new} + \gamma \tag{20}$$

where $\hat{y}_{new}$ is deterministic and γ is a random variable.

It is to be noted that the solution of Problem 1 was obtained for fixed bandwidth σ^2 and regularization parameter λ. In practice, however, it is generally needed to estimate such parameters from data as well. In the presented univariate case, the tuning of such parameters was efficiently performed by means of a Generalized Cross Validation (GCV) approach. It is also possible, if there is not enough data to perform GCV, is to tune σ^2 relying on some theoretical criterion, such as Silverman's rule [13].

4 Results

In this section, Regularized Entropy Learning (REL) is tested against simulated datasets and data coming from the semiconductor manufacturing industry. In order to understand the potential of the proposed approach, its performances are compared with Kernel Ridge Regression (KRR). The performance gap between Regularized Entropy Learning and Kernel Ridge Regression is expected to vary: intuitively, if the real uncertainty distribution is strongly nongaussian, REL is expected to outperform KRR. For all the experiments, RMSE was used as metric; the parameters of both methods were tuned by means of GCV.

4.1 Simulated Datasets

Two datasets (100 training, 50 validation and 50 test observations) were generated using strongly nongaussian uncertainty distributions. In the first case, a uniform random variable was employed; in the second case, the uncertainty was modeled as a Laplace distribution. Multiplicative (5x) outliers were inserted (with 10% probability) in each dataset to test the robustness of the proposed methodology. Table 1 presents the RMSE ratios for the four simulated experiments: as expected, the best results are obtained with the Laplace distribution (whose power law is hardly approximated by a Gaussian). Figures 4 and 5 show the uncertainty distributions considered by REL and KRR: the advantage of the proposed approach in such situations is clear.

4.2 Semiconductor Dataset

REL was tested against a set of homogeneous observations from the semiconductor manufacturing environment (courtesy of the Infineon Austria facility in Villach). Specifically, a homogeneous dataset consisting of 239 measured wafers was collected from a Chemical Vapor Deposition (CVD) equipment. Every wafer is characterized by 30 input variables and 9 thickness measurements (associated to different sites on the wafer). Four experiments were conducted:

- Predict the average layer thickness (with and without outliers)
- Predict the standard deviation (9 points) of layer thickness (with and without an outliers)

In order to produce outlier-free experiments, a standard outlier elimination technique based on Mahalanobis distance was employed. The original dataset was split in 3 consecutive groups: a training dataset of 150 wafers, a validation dataset of 50 wafers and a test dataset of 39 wafers. The validation dataset was used to tune the hyperparameters, while the algorithms were compared on the test dataset.

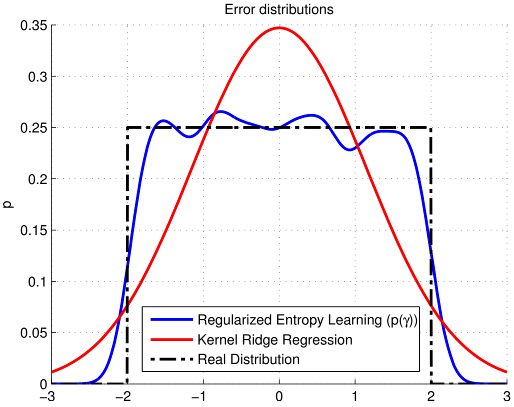


Fig. 4. Error distributions for the prediction of a simulated dataset with uniform uncertainty: the nonparametric estimation of $p(\gamma)$ allows the proposed methodology to outperform KRR

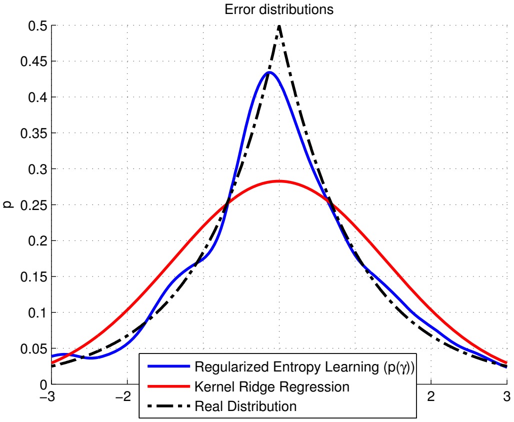


Fig. 5. Error distributions for the prediction of a simulated dataset with Laplace-distributed uncertainty: the fat tails of the Laplace distribution are correctly recognized by REL, while the Gaussian assumptions of KRR fail to achieve the best prediction results

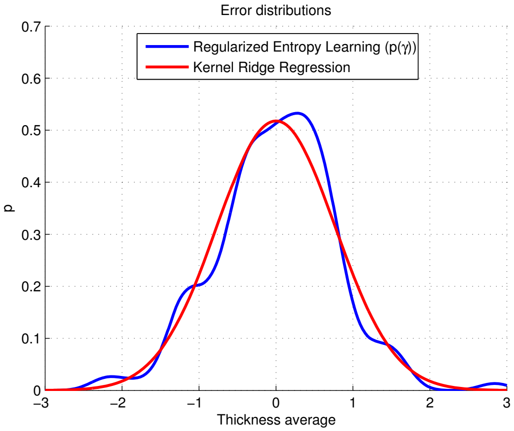


Fig. 6. Error distributions for the prediction of average thickness with no outliers: notably, in this case the Gaussian assumptions are verified, and KRR performs better than the proposed methodology

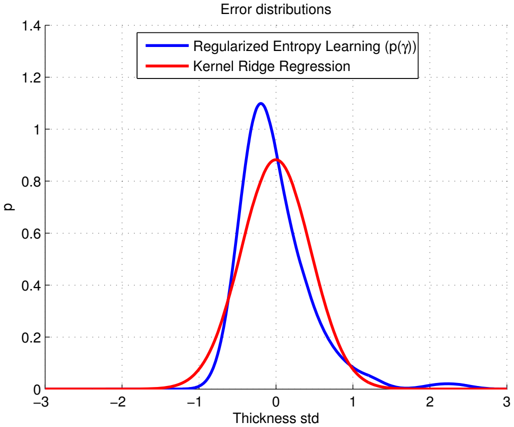


Fig. 7. In the prediction of standard deviation, the density estimated distribution is skewed and strongly nongaussian (both the distributions in this Figure has expected value 0). Thanks to this better understanding of the uncertainty distribution, the proposed methodology performs well in this setting.

The results of this experiment are reported in Table 1. Notably, the Gaussian assumptions are verified for the average thickness without outliers (Figure 6): in this case, KRR performs better than REL. On the other hand, the presence of outliers in the dataset significantly degrades the performances of KRR, while the proposed methodology proves to be naturally robust. Perhaps the most interesting result comes from the prediction of standard deviation: thanks to the skewed uncertainty distribution associated to the standard deviation measurements (Figure 7), REL outperforms KRR.

Table 1. RMSE ratios REL/KRR: the proposed methodology shows natural robustness with respect to outliers, and performs better than KRR when Gaussianity assumptions are less realistic. The best result is in bold, while the worst result is in red.

	With outliers	Without outliers
Uniform	0.61	0.67
Laplace	**0.52**	0.56
Thickness Avg.	0.98	1.17
Thickness Std.	**0.73**	0.89

5 Conclusions

In this paper, a novel learning methodology is proposed relying on information theory concepts in a regularized machine learning framework. This study is motivated by the application of Virtual Metrology in semiconductor manufacturing. The estimation of a nonlinear predictive model, jointly with the associated uncertainty distribution, is achieved in a nonparametric way using a metric based on Renyi's entropy and regularized with a RKHS norm. The proposed methodology, namely Regularized Entropy Learning (REL), has been tested with promising results on simulated datasets and process data from the semiconductor manufacturing environment. Specifically, the proposed approach presents a clear advantage in outlier-intensive and strongly nongaussian environments: for this reasons, REL is a strong candidate for the use in an industrial setting, where a flexible assessment of data variability is key to achieve good predictive performances.

References

1. Ackerman, F., Stanton, E., Bueno, R.: Fat tails, exponents, extreme uncertainty: Simulating catastrophe in DICE. Ecological Economics 69(8), 1657–1665 (2010)
2. Carreira-Perpiñán, M.: Mode-finding for mixtures of Gaussian distributions. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(11), 1318–1323 (2002)
3. Erdogmus, D., Principe, J.: An error-entropy minimization algorithm for supervised training of nonlinear adaptive systems. IEEE Transactions on Signal Processing 50(7), 1780–1786 (2002)
4. Gray, R.: Entropy and information theory. Springer (2010)
5. Hastie, T., Tibshirani, R., Friedman, J., Franklin, J.: The elements of statistical learning: data mining, inference and prediction. The Mathematical Intelligencer 27(2), 83–85 (2005)
6. Miller, K.: Multidimensional gaussian distributions. Wiley, New York (1964)
7. Parzen, E.: On estimation of a probability density function and mode. The Annals of Mathematical Statistics 33(3), 1065–1076 (1962)
8. Popovic, D., Milosavljevic, V., Zekic, A., Macgearailt, N., Daniels, S.: Impact of low pressure plasma discharge on etch rate of SiO2 wafer. In: APS Meeting Abstracts, vol. 1, p. 8037P (2009)
9. Principe, J.: Information Theoretic Learning: Renyi's Entropy and Kernel Perspectives. Springer (2010)

10. Principe, J., Xu, D., Zhao, Q., Fisher, J.: Learning from examples with information theoretic criteria. The Journal of VLSI Signal Processing 26(1), 61–77 (2000)
11. Rallo, R., Ferre-Giné, J., Arenas, A., Giralt, F.: Neural virtual sensor for the inferential prediction of product quality from process variables. Computers & Chemical Engineering 26(12), 1735–1754 (2002)
12. Scholkopf, B., Smola, A.: Learning with kernels, vol. 64. Citeseer (2002)
13. Silverman, B.: Density Estimation for Statistics and Data Analysis. Number 26 in Monographs on Statistics and Applied Probability (1986)
14. Wang, P., Vachtsevanos, G.: Fault prognostics using dynamic wavelet neural networks. AI EDAM 15(04), 349–365 (2001)
15. Weber, A.: Virtual metrology and your technology watch list: ten things you should know about this emerging technology. Future Fab International 22, 52–54 (2007)

Appendix A

Problem 1 is an unconstrained global optimization problem. In order to derive a solution, we consider the features of c^* in the following

Proposition 5. Let c^* be the global minimum of a negatively weighted sum of Gaussian densities $J(c)$. Thanks to the properties of $J(c)$, there exists a real m such that

$$||c^* - d_{ij}||^2_{D_{ij}^{-1}} \leq m$$

for at least one mean vector d_{ij}. Furthermore,

$$m \leq \log\left(\frac{C_0}{\tilde{J}}\right)$$

where C_0 is a negative constant and

$$\tilde{J} = \min_{i,j} J(d_{ij})$$

That is, m is superiorly limited by a decreasing function of the minimum value of J evaluated in the mean vectors d_{ij}. □

Proposition 5 has two notable implications: **(i)** there is at least one mean vector d_{ij} that serves as suitable starting point for a local optimization procedure, and **(ii)** the global minimum gets closer to one of the mean vectors as the computable quantity $\tilde{J}$ increases. Using these results, c^* is found by means of the following

Algorithm 1. Solution of Problem 1.

```
1. Set $c^* = 0_N$
2. For $i = 1, \ldots, N$
   (a) For $j = i + 1, \ldots, N$
         – Use a Newton-Raphson algorithm to solve the local optimization
           problem
                 $c^*_{ij} = \arg\min_c J(c)$
           using $d_{ij}$ as starting point.
         – If $J(c^*_{ij}) < J(c^*)$
         – $c^* = c^*_{ij}$
         – End if
   (b) End for
3. End for
```

Algorithm 1 was originally proposed in [2], and guarantees to find all the modes of a mixture of Gaussian distributions. It is to be noted, however, that the exhaustive search performed by Algorithm 1 might be computationally demanding, and only the global minimum of J is of interest in the presented case. It is convenient, if an approximate optimal solution is acceptable, to perform convex optimization using a reduced number of starting points. In our experiments, the best performances were obtained using the d_{ij} associated to the least (1% to 5%) values of $\{J(d_{ij})\}$. It is to be noted that this reduced version of Algorithm 1 does not guarantee to reach the global minimum (although it has been verified, via simulation studies, that the global optimum is found with a very high success rate). In order to set up the Newton-Raphson algorithm used in Algorithm 1, it is necessary to know the Jacobian and Hessian matrices associated to $J(c)$. Letting

$$
\begin{aligned}
Q_{ij}(c) &= -\frac{1}{2}(c - d_{ij})' D_{ij}^{-1} (c - d_{ij}) \\
\overline{\alpha}_{ij} &= \frac{\alpha_{ij} |D_{ij}^{-1}|^{1/2}}{(2\pi)^{N/2}}
\end{aligned}
$$

it is possible to write

$$J(c) = -\sum_{i=1}^{N} \sum_{j=i+1}^{N} \overline{\alpha}_{ij} e^{Q_{ij}(c)}$$

Therefore, the Jacobian of $J(c)$ is

$$\frac{\partial J}{\partial c} = -\sum_{i=1}^{N} \sum_{j=i+1}^{N} \overline{\alpha}_{ij} e^{Q_{ij}} \frac{\partial Q_{ij}}{\partial c}$$

and the Hessian is

$$\frac{\partial^2 J}{\partial^2 c} = -\sum_{i=1}^{N}\sum_{j=i+1}^{N} \overline{\alpha}_{ij} e^{Q_{ij}} \left[\frac{\partial Q_{ij}}{\partial c} \frac{\partial Q_{ij}}{\partial c'} + \frac{\partial^2 Q_{ij}}{\partial^2 c} \right]$$

with

$$\frac{\partial Q_{ij}}{\partial c} = D_{ij}^{-1}(d_{ij} - c)$$
$$\frac{\partial^2 Q_{ij}}{\partial^2 c} = -D_{ij}^{-1}$$

Considering the Taylor series of $J(c)|_{c=c_k}$ truncated to the second order

$$J(c)|_{c=c_k} \approx J(c_k) + (c - c_k)' \frac{\partial J}{\partial c}|_{c=c_k}$$
$$+ \frac{1}{2}(c - c_k)' \frac{\partial^2 J}{\partial^2 c}|_{c=c_k}(c - c_k)$$

the next c_{k+1} is

$$c_{k+1} = x_k - \left(\frac{\partial^2 J}{\partial^2 c}|_{c=c_k} \right)^{-1} \frac{\partial J}{\partial c}|_{c=c_k}$$

Remark: in order to evaluate $J(c)$, as well as its Jacobian and Hessian, it is not necessary to explicitly compute any matrix inversion: indeed,

$$D_{ij}^{-1} = \frac{(K_i - K_j)'(K_i - K_j)}{2\sigma^2} + \lambda K$$

Appendix B

Proof of Proposition 3. It is apparent that a global shift of x does not influence the value of H_2, thanks to the infinite integration interval:

$$\int_{-\infty}^{+\infty} p_\epsilon(x)^2 dx \equiv \int_{-\infty}^{+\infty} p_\epsilon(x + \tau)^2 dx$$

The result follows. □

Proof (by contradiction) of Proposition 5. Consider, without loss of generality, the problem of finding the global maximum x^* of a weighted sum of Gaussian densities $L(x)$, such that

$$L(x) = \sum_{i=1}^{N} \alpha_i G(\mu_i, \Sigma_i;\ x)$$

Letting

$$\overline{\alpha}_i = \frac{\alpha_i}{(2\pi)^{N/2}|\Sigma_i|^{1/2}}$$
$$||x-\mu_i||^2_{\Sigma_i^{-1}} = \frac{1}{2}(x-\mu_i)'\Sigma_i^{-1}(x-\mu_i)$$

we write $L(x)$ as

$$L(x) = \sum_{i=1}^{N} \overline{\alpha}_i e^{-||x-\mu_i||^2_{\Sigma_i^{-1}}}$$

Consider then the best function value among the $\{\mu_i\}$ as

$$\tilde{L} = \max_i L(\mu_i)$$

and let x^* be far from every μ_i so that

$$||x^*-\mu_i||^2_{\Sigma_i^{-1}} \geq m \, \forall i$$

where m is a lower bound. It is apparent that

$$\tilde{L} \leq L(x^*) \leq Ne^{-m} \sum_{i=1}^{N} \overline{\alpha}_i \tag{21}$$

and it is immediately verified that if

$$m > \log\left(\frac{N\sum_{i=1}^{N}\alpha_i}{\tilde{L}}\right)$$

inequality (21) does not hold: by contradiction, x^* is not the global maximum. Therefore, there exists at least one mean vector μ_i such that

$$||x^*-\mu_i||^2_{\Sigma_i^{-1}} < \log\left(\frac{N\sum_{i=1}^{N}\alpha_i}{\tilde{J}}\right)$$

AutomationML as a Shared Model for Offline- and Realtime-Simulation of Production Plants and for Anomaly Detection

Olaf Graeser[1], Barath Kumar[2], Oliver Niggemann[2],
Natalia Moriz[2], and Alexander Maier[2]

[1] PHOENIX CONTACT GmbH & Co. KG, Support Unit Manufacturing Solutions,
Flachsmarktstraße 8, D-32825 Blomberg, Germany
ograeser@phoenixcontact.com
[2] inIT-Institut Industrial IT, Ostwestfalen-Lippe University of Applied Sciences,
Liebigstrasse 87, D-32657 Lemgo, Germany
{barath.kumar,oliver.niggemann,natalia.moriz,
alexander.maier}@hs-owl.de

Abstract. The growing complexity of production plants leads to a growing complexity of the corresponding automation systems. Two significant challenges have to be faced: (i) the control devices have to be tested *before* they are used in the plant. (ii) The diagnosis functions within the automation systems become more and more difficult to implement; this entails the risk of undetected errors. Both challenges may be solved using a joint system model of the plant and the automation system: (i) Offline simulations and HIL tests use such models as an environment model and (ii) diagnosis functions use such models to define the normal system behaviour. Modular models like this can be represented with AutomationML. Additionally, testing and diagnosis require the simulation of these models. Therefore, a corresponding simulation system for AutomationML models is presented here. A prototypical tool chain and a model factory (MF) are used to show results for this approach.

Keywords: Automation Technology, Modelling, Simulation, HIL Test, Functional Mock-up Unit and AutomationML.

1 Introduction

The virtual planning of factories (“Digital Factory”) has been a major trend over the last years (e.g., see [1] and [2]): By using simulated factories in early development phases, system integrators and factory operators want to (*i*) minimise design and construction times, (*ii*) reduce the number of errors in the factory and (*iii*) optimise factory designs and configurations. This could lead to more complex factories, faster processes, and more integrated production chains.

So far, research in virtual factories has mainly focused on the plant and machine factory construction aspect. However, more complex factories also lead to more complex automation systems. Here, we will present a concept and first prototypes for a modelling and simulation approach for an automation-centred virtual design and

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 195–209.
springerlink.com

testing process. In other words, unlike machine-centred approaches, precise models of the automation systems are used while the machines and plants are only modelled with a limited level of details – often timed event sequences are sufficient.

A software tool chain for model-based development and simulation of a plant is introduced. This tool chain starts with an editor for plant modelling and ends in a simulation framework for offline simulations, HIL tests and online plant diagnosis. These are the three main use cases for the editor and the simulation framework.

- In the offline simulation, Soft-PLCs (Programmable Logic Controllers), i.e. the control applications, can be tested by simulating Soft-PLCs and the corresponding process.
- HIL tests can be used to test a PLC given as hardware. Unlike an offline simulation, this requires a realtime simulation.
- Online diagnosis can be used to detect changes and anomalies in the behaviour of a plant – mainly by comparing the real plant behaviour to the simulated behaviour.

2 State of the Art

For automation technology, the challenge is to model and develop control software, visualisation and diagnosis software. Such software modules should be reusable (for future projects), testable on a standalone PC and also be platform independent (with respect to the used communication hardware).

In the area of embedded systems software development, several projects and standards addressed the above-mentioned challenges. For example safety-related systems (IEC 61508, see [3]), automotive software development [4], the field of automation (IEC 61499 and IEC 61131), or several research projects like Modale [5], OrVia [6] or Medeia [7].

The introduction of explicit systems, software and plant models is an important trend in embedded software development [8, 9] as well as in the field of automation technology [5, 7, 10]. The modelling approach in this work is based on control and plant component models. For this purpose, standards such as AutomationML [11] and Modelica [12] are used and improved.

Furthermore, the concept of *separation of concerns* is used. A complete plant model consists of various aspects, like logical software models, hardware topologies, network descriptions and a system model. For a better reusability of the single models, they are modelled and stored separately. The combination of the single models is done in a later step. The approach used in this work is inspired by the Object Managements Group's (OMG's) Model-Driven Architecture (MDA) [13, 14]. In the first step, a logical structure with software modules and plant models is created. This correlates to the MDA's Platform Independent Model (PIM). Later, this structure is mapped onto specific hardware descriptions. Components of the PIM, which describe a class or a role of a device in the model, are thereby linked to real hardware. The result then correlates to the MDA's Platform Specific Model (PSM).

3 Modelling

Factories and the corresponding automation systems comprise more and more heterogeneous and interconnected components. Therefore the modelling formalism and the simulation framework must support such systems. The new standard AutomationML is able to integrate heterogeneous description formalisms and models automation systems, plants, sensor/actuators, and especially their interconnection.

In general, our modelling approach is inspired by OMG's MDA paradigm (see [14] for details):

In a first step, a logical architecture comprising only software modules and plant models is created – MDA's PIM. Only later on, this logical architecture is mapped onto a specific hardware topology, creating the system architecture — MDA's PSM. This can also be seen in Fig. 1.

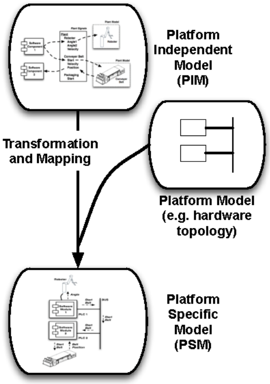

Platform Independent Model (PIM)
Transformation and Mapping
Platform Model (e.g. hardware topology)
Platform Specific Model (PSM)

Fig. 1. Model-driven architecture approach

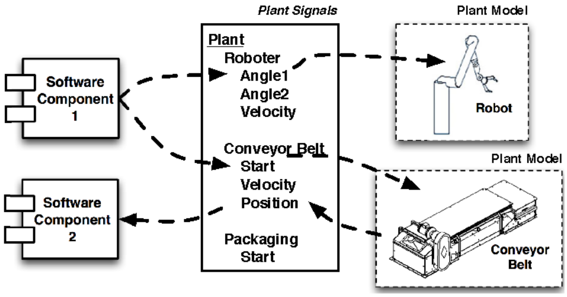


Fig. 2. Platform-independent model

3.1 Platform Independent Model

As can also be seen in Fig. 2, PIMs comprise software components (i.e. the software architecture) and plant models. Software components are control algorithms, diagnosis software algorithms, or monitoring software modules. These software components do not communicate directly with each other or with I/O drivers but via plant signals. Plant signals are variables whose semantics are defined by the plant model, e.g. in Fig. 2 software component 1 sets the robot's "angle1" signal or starts the conveyer belt by a "start" signal.

Please note that plant signals usually correspond to (physical) states of the real plant, i.e. unlike software interfaces, they exist in reality and hence need not be changed when the software architecture is modified. This increases the level of software reuse since software modules do not have to refer to software-specific interfaces anymore. Since we only consider automation systems, it is sufficient to restrict to the plant signals in most cases.

3.2 Platform-Specific Model

PIMs have the advantage that they can be reused for a large variety of hardware topologies (i.e. platforms). For this, the PIMs are mapped onto a specific hardware topology during a second step, which creates a PSM. Many features of the PSM can be derived automatically, e.g. the schedule for realtime communication networks such as I/O driver configurations or Profinet configuration (see [15] for Profinet details), the scheduling algorithm is outlined in [16]. This automatism allows developers to focus on the most difficult part of the development process: The development of the applications.

Such a PSM for the above-mentioned PIM can be seen in Fig. 3: Software component 1 has been mapped onto PLC 1, component 2 onto PLC 2. Now, unlike in Fig. 2, the signal to start the conveyor belt must be transmitted via the bus while the robot control signal can be transmitted directly to the robot.

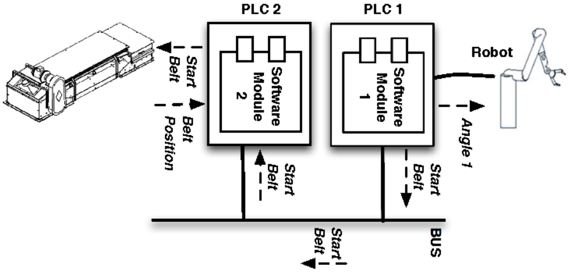


Fig. 3. Platform specific model

Please note that the same PIM can be mapped onto a different hardware topology, i.e., this approach leads to higher level of model reuse.

3.3 AutomationML System Model

As mentioned before, the key challenge lies in the modelling and simulation of heterogeneous and modular systems; systems that comprise both the automation devices and the plants. As shown on the left-hand side of Fig. 4, AutomationML is well suited for this task, since the goal of AutomationML is to describe a production plant completely, including all components in it and to support each phase and discipline of plant engineering. AutomationML is based on XML and makes use of well-established, open and free XML-based standards of the involved engineering disciplines. For example, the object model is given by CAEX [17], geometries and kinematics are described by COLLADA [18] and the behaviour of the plant is described with PLCopen XML [19]. Using these standards ensures a slim specification of AutomationML [11].

In AutomationML, a system is defined as a hierarchy of interconnected objects (bottom part of Fig. 4). Each object may be the instance of a specific class from a library — where a "RoleClass" defines an abstract type such as a PLC and a "SystemUnitClass" a specific product. From another point of view, a "RoleClass" can be seen as the requirements for an object, and a “SystemUnitClass” is an implementation of these requirements. Furthermore, objects communicate via well-defined interfaces.

Within the concept outlined here, interfaces were defined for plant modules, sensor/actuators, communication networks, and PLCs. These interfaces mainly define incoming and outgoing signals and timing information.

AutomationML is able to include arbitrary XML files. Here, an XML-based description of FMUs is used to link objects in AutomationML to simulation models. FMUs are a new standard from the Modelisar project [20] to provide a uniform API to arbitrary simulations models such as TheMathwork's Simulink or Dassault's Dymola. In this project, a mapping between our interfaces in AutomationML and FMU interfaces has been developed (right-hand side of Fig. 4). This allows for mapping between AutomationML objects and FMU-based simulation models, i.e., users are able to provide heterogeneous simulation models for all our AutomationML objects, hence supporting the required level of support for heterogeneous and modular systems.

The interface to map FMUs to AutomationML objects is similar to all other external data mappings like Collada and PCLopen XML. Both inherit from AutomationML's interface class “ExternalDataConnector”. In addition to the “ColladaInterface” and the “PLCOpenXMLInterface”, a “FMUInterface” is defined, which refers to an external FMU file. The in- and output variables used in the FMU are described in the “modelDescription.xml” inside the FMU file. For now, the mapping to the variables in the AutomationML format is done by name equality.

In this work, a drag and drop editor was developed for a PIM as described above. The editor is based on the Visual Studio 2010 Visualization and Modeling SDK [21]. The graphical user interface can be seen in Fig. 6. Models created with this editor abstract from the communication hardware used in the plant. As described above, software components (e.g. PLC applications) are connected to sensors and actuators via named signals.

The model of the plant is stored in AutomationML format. Each component of the plant model is mapped to an element of the AutomationML role class lib.

As mentioned before, the idea behind the role class lib is that each component in a plant has to fulfil a function and play one or more roles. Role classes can be understood as a requirement definition, which a component has to satisfy.

The use of a graphical editor (Fig. 6) has the advantage over an XML editor that new components (like role classes) can be added to the model per drag-and-drop and the relationships of the components and the flow directions of the communication signals are easy to identify.

The use if AutomationML increases the reusability of the models, because it can be expected that many software tools in the field of industrial automation will at least support AutomationML import and export.

Currently, the plant is only modelled as a PIM. Future versions of the editor will also include the PIM to PSM transformation. This can be done by assigning a real hardware description to each AutomationML role class lib-based component of the PIM.

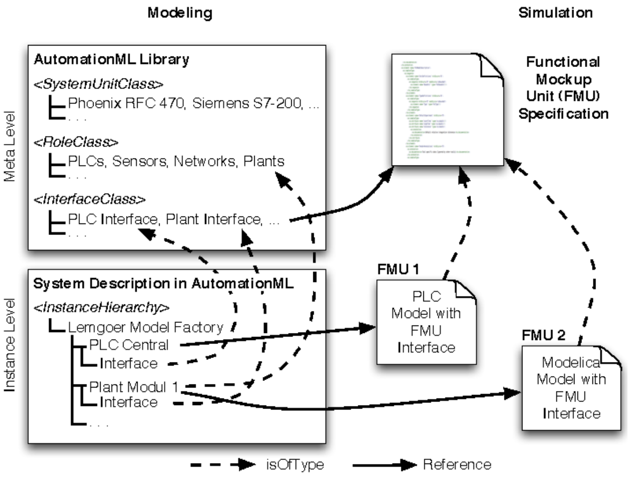

Modeling
Simulation
Meta Level
Instance Level
AutomationML Library
<SystemUnitClass>
Phoenix RFC 470, Siemens S7-200, ...
. . .
<RoleClass>
PLCs, Sensors, Networks, Plants
. . .
<InterfaceClass>
PLC Interface, Plant Interface, ...
. . .
System Description in AutomationML
<InstanceHierarchy>
Lemgoer Model Factory
PLC Central
Interface
Plant Modul 1
Interface
. . .
Functional Mockup Unit (FMU) Specification
FMU 1
PLC Model with FMU Interface
FMU 2
Modelica Model with FMU Interface
isOfType
Reference

Fig. 4. Modelling and simulating heterogeneous modular systems with AutomationML

3.4 Modelica Behaviour Models

The behavioural model of the whole plant consists of several submodels, and is modular in nature. As mentioned in section 3.1 the plant signal information (i.e., incoming, outgoing signals and timing information) are often sufficient to provide the necessary abstraction needed for modelling. One possibility is to use Finite State Machines or Automata (FSM) for constructing these behavioural models. However, FSMs do not support modelling complex timed hybrid systems. Hence, our formalism 'Probabilistic regression automata' (PRA) described in [22], which is an extension of hybrid timed automata [23], [24] is used for this purpose. In PRA, an event can be triggered when the value of one or several signals cross the specified threshold value(s) within a certain time window reflecting the actual signal timing of the real plant. In addition to the above, probabilistic information is used to handle non-deterministic scenarios (i.e. an automata can show different behaviours for the same set of inputs). A typical case of non-determinism is the occurrence of errors, which is inevitable (especially, in industrial automation system) - however the occurrence of these errors is less probable then the occurrence of the correct scenarios. Moreover, our formalism supports continuous signals and allows the arbitrary (not only linear) signal value changes within a state.

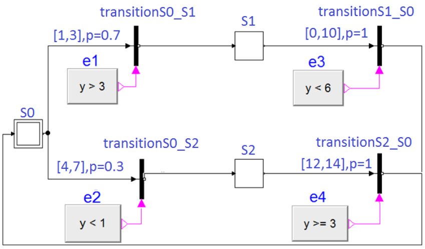


Fig. 5. Finite state machine modelled with Dymola

The automata used in this work were implemented in Dymola [25]. Dymola is suitable for the requirements of modelling and simulating production plants, because this tool supports hierarchical model composition, many libraries of reusable components from several engineering domains, faster simulation and open interfaces to other programs (FMI). Dymola enables users to develop their own components. In our tool chain, the basis for the PRA prototype is developed with the *StateGraph* and *StateGraph2* Modelica-standard libraries. This basis is extended as follows: (*i*) the interface is added (the names of the variable are the same as in the PLC program), (*ii*) the signal behaviour is integrated, (*iii*) the timing and probability information is added. Currently, our own library for PRA is under development. A PRA prototype is represented in Fig. 5.

Created behavioural models are stored as FMUs. FMUs are executables, which implemented the Functional Mock-Up Interface (FMI). The FMI functions are called by a simulator (here the “FMU Simulation Component”) to create and execute one or more instances of the FMU. These FMUs can be referenced in the above-mentioned PIM-editor.

4 The Simulation Framework

As mentioned before, the simulation framework must support heterogeneous and interconnected components – mirroring AutomatioML's capability to model such systems. Here, a corresponding simulation framework is described which uses the new standard FMU. This standard encapsulates heterogeneous behaviour models such as Modelica or the Mathwork's Simulink behind a standardised API.

The use cases of the project are offline simulations for testing the control algorithms, HIL tests of PLCs (and plant components), and anomaly detection in the operating plant. For these use cases, a flexible Simulation Framework was developed, in which Simulation Components can be easily removed, replaced or added. The Simulation Framework consists mainly of the Simulation Manager, which is responsible for the instantiation and scheduling of Simulation Components, and the Process Data Manager, which manages the process data of the plant model and distributes them to the Simulation Components (Fig. 6).

The Simulation Framework makes use of the Managed Extensibility Framework (MEF) [26] to connect all available Simulation Components to the Simulation Framework. MEF is part of Microsoft's .NET framework and supports the dynamic plug and play concept of software components. Which Simulation Components are needed, and in which quantity, is defined in the AutomationML plant description file coming from the Modelling Tool (Section 3). The Simulation Components are identified by their class name (e.g. “FMU Simulation Component”). If more than one Simulation Component of the same class is needed, an additional specific instance name is used.

The Process Data Manager is part of the Simulation Manager. If a Simulation Component is instantiated by the Simulation Manager, a process data input list and an output list has to be registered at the process data manager. The Process Data Manager then compares the process data in the lists with the already registered process data by name. New process data will be added, already existing process data are linked together (Fig. 8).

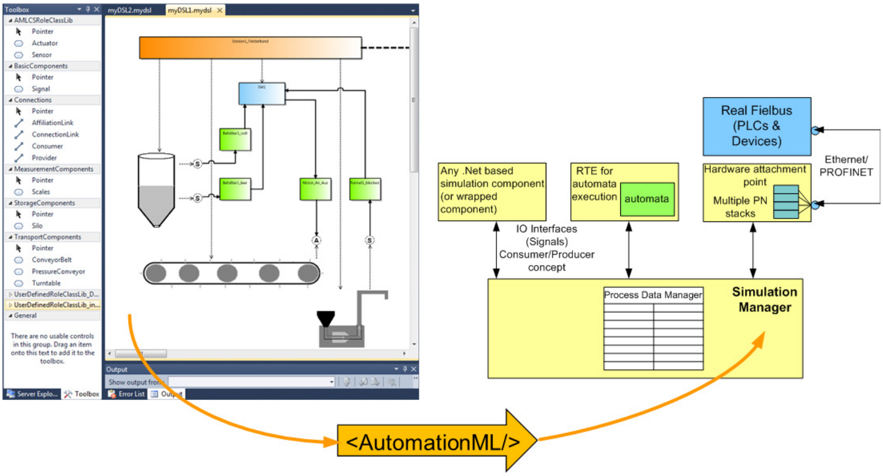


Fig. 6. The PIM Editor stores the plant model as AutomationML file. This file is then loaded into the Simulation Framework to instantiate all needed Simulation Components.

4.1 Scheduling

One important task of the Simulation Manager is to schedule all Simulation Components. For this purpose, a simple static scheduling algorithm is used. Every Simulation Component has to declare, with what time step the component needs to be scheduled. For all of these time steps, the least common multiple (LCM) is calculated. Every Simulation Component is scheduled at the beginning (*t_0*). Furthermore, each Simulation Component is scheduled again with its specific time step as long as the time step sum is smaller than the LCM. For example, in Fig. 7, three Simulation Components (*SC_1, SC_2, SC_3*) are scheduled. *SC_1* with a time step of 10ms, *SC_2* with a time step of 20ms and *SC_3* with a time step of 40ms. The LCM is therefore 40ms. The resulting schedule is cyclic which means as soon as the end of the schedule is reached, the schedule starts all over again.

Currently, the Simulation Framework is not capable of parallel processing of Simulation Components. Hence, the components are processed in sequence. It is the responsibility of the Process Data Manager to maintain two states of each process data (Fig. 8). The first state contains the input process data of the components. After the processing of a Simulation Component, the resulting process data are written to the second state.

After all parallel Simulation Components have been processed; the resulting process data (second state) are written to the first state and are now available as input values for the next simulation step. Using this double buffer concept makes sure that two or more Simulation Components, which are scheduled at the same time, work on the same set of process data.

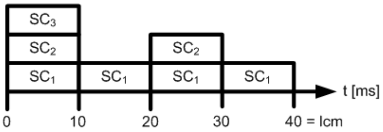


Fig. 7. Simulation manager example schedule

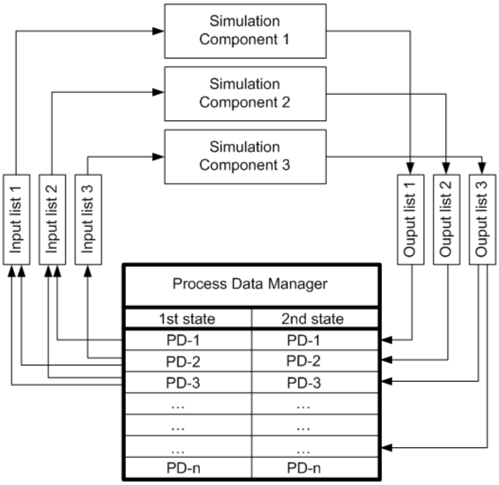


Fig. 8. The process data manager

4.2 Offline Simulation

For the offline simulation, the AutomationML plant description file is loaded and all plant behaviour models, represented by FMUs, are loaded and instantiated as FMU Simulation Components. Because the Simulation Component “FMU Simulation Component” is used more than once, the instances have to be identified by their instance names, which can be given by the file names of the loaded FMUs. After the creation of a schedule for all necessary simulation components, the components are executed. The actual process data in each time step are managed by the Process Data Manager and can be logged by a specialised simulation component.

4.3 HIL Simulation

Hardware-In-the-Loop (HIL) is a method for testing hardware devices such as PLCs or plant modules – these devices are then called unit under test (UUT). HIL differs from realtime or offline simulations by considering the ‘actual’ device in the test loop. In a HIL test, a simulator (e.g. a special PC) simulates a model of the environment for the attached UUT (i.e. all other communicating partners of the overall system, except the UUT); this can be seen in Fig. 9. The purpose of the HIL test is to make the UUT believe that it is in fact connected to the real system. By using a HIL test, UUTs can be tested with realistic plant signals providing the necessary sensor/actuator triggers to fully exercise the UUT – without requiring the real system.

In automation, several parts of the overall system may become the UUT; e.g., when several PLCs are tested. Modular models as described in this paper, allow for a fast creation of the environment model by removing those parts from the overall model that correspond to the UUT.

Fig. 10 shows a typical HIL case in the domain of industrial automation: A real PLC is tested; for this, the PLC model is removed from the overall model to create the environment model. The PLC is then tested for correct responses, i.e., in terms of (*i*) setting its actuator variables and (*ii*) performing its duties in meeting the realtime requirements of the overall system. The appropriate sensor inputs needed by the PLC are sent at the right time by simulating the environment model exploiting the input sample space of the PLC under test.

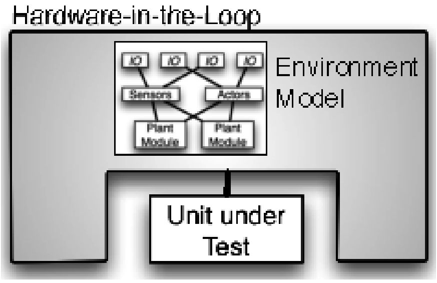


Fig. 9. Hardware-in-the-loop (HIL) test setup

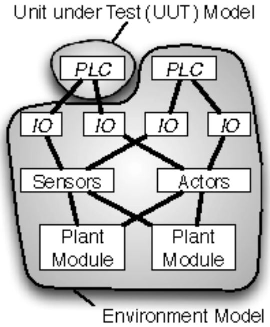


Fig. 10. Typical HIL case for automation

Some typical errors in the automation domain for which a PLC has to be tested are: (*i*) functional input space errors, i.e., does the PLC respond correctly when a blocked funnel is sensed? And, (*ii*) time domain errors i.e., even if the PLC takes the right action; Was it done within the required time limit? For example, one such industrial automation test question covering the above points would be: if the funnel is blocked, does the PLC stop the conveyor belt? – and does it do it in time?

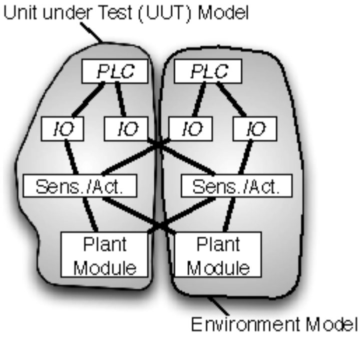


Fig. 11. Another typical HIL case for automation

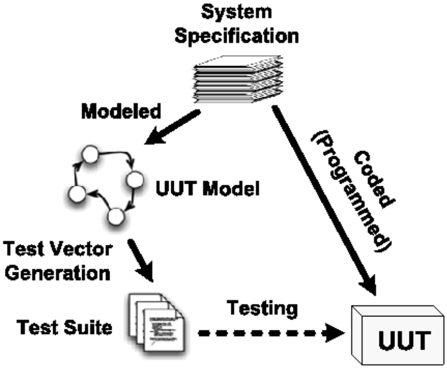


Fig. 12. 2-Model Approach

Another typical automation testing scenario is shown in Fig. 11: A whole plant module including its PLC, IO devices, and the plant is being tested.

For testing, two key questions have to be answered: (*i*) where do the test cases come from? And (*ii*), who defines whether a test case passed or failed – this is the so-called test verdict. Generally speaking, two solution approaches exist:

1-Model Approach: 1-Model Approach considers one model for both test vector generation and code generation for the UUT (e.g. PLC); the test verdict is generated by comparing the test output with the prediction of the UUT model. The key issue with this approach is: since the implementation and the test cases are derived from the same model, only the code generator can be tested.

2-Model Approach: 2-Model Approach (also see Fig. **12**) on the other hand, considers a separately implemented unit under test (UUT), i.e., the UUT in programmed using conventional programming languages (e.g. IEC 61131 languages, in case of PLC) based on an existing informal specification. In addition to this, a specific formal testing model is developed based on the same informal specification and is used for test vector generation; the test verdict is created by comparing the test output with the prediction of this test model. Since the generated vectors and the UUT code are not derived from the same model, this approach provides the necessary redundancy needed for testing. For our HIL test, we use the 2-Model Approach.

Our prototypical HIL test bed consists of a manually-programmed (e.g., using IEC61131) UUT and the testing model. The test model can be obtained in two ways:

(i) by making use of the modular separation concept of AutomationML (as shown in Fig. 10) - i.e., in this case the environment model (overall model minus UUT model) is also the testing model. The simulation framework can then be used to simulate all the individual components (For example, other PLC models, IOs and plant models, etc.) of the test model. Such a testing scenario results in a closed-loop test of the UUT.

(ii) By generating test vectors from a single automaton which is obtained by merging the automata of all the individual components of the environmental model. Such a test model can be used for an open-loop test.

In both of the above two scenarios, the test model is used as a bases to extract ‘abstract test input sequences’ (our publication [27] details on the algorithmic concepts of test extraction) either for simulating the environmental model (closed loop) or for test execution (open loop). These extracted input sequences are first serialised and then built into appropriate industrial bus protocol frames (in our case Profinet IO frames). The HIL sends these sensor values to the UUT (PLC) at appropriate times, simulating the actual execution environment of the UUT taking the real time requirements into consideration. After the UUT reacts to these inputs, the outputs are constantly monitored and compared with the results predicted by the testing model.

Empirical Results: The HIL framework was used for both closed-loop and open-loop tests for our initial test studies using the MF shown in Fig. **13**. This plant is used to produce and pack up bulk goods (e.g., popcorn). This modular plant comprises of 8 modules which include a storage system, several transportation systems, a weighing mechanism, a bottling station, a robot, a heating facility, and a packaging module. The automation solution comprises of several distributed PLCs to control the plants, buses such as Profinet, and approx. *250* sensor and actuator signals.

For the first results, we performed open-loop test with just 2 modules of the MF. To test the credibility of the generated test vectors - 24 mutants (16 functional (i.e., actuator errors) and 8 timing errors) were introduced into the PLC control logic, which were positively identified. A test model for closed-loop test with 2 modules of the MF were modelled using Modelica FMUs; these FMUs were simulated using the proposed simulation framework to imitate the execution environment of the PLC. Further tests were conducted for ideal plant scenarios and with 4 functional and 2 timing errors; which were correctly detected by the HIL framework. Table 1 summaries HIL test's initial results. Even though the first results are positive - further studies are needed to check complex test scenarios; For example, including all 8 modules of the MF and by including error with an exhaustive exploitation of mutant sample space.

Table 1. HIL test results

Type of Test	Inserted mutants		Detected mutants
	Functional mutants	Timed mutants	
Closed Loop	4	2	6
Open Loop	16	8	24

4.4 Anomaly Detection

The detection of system deterioration and of non-normal system behaviour will prove essential for the improvement of system quality and reliability: Production facilities could detect failures earlier and inspections can be planned whenever the first signs of deterioration appear.

The key idea is rather simple: Let *M* be a system model, which captures all fault relevant system aspects; here we use the model from section 3. Whenever *M*'s behaviour deviates from the system's current behaviour, the user will be alarmed.

Fig. 13. A model factory

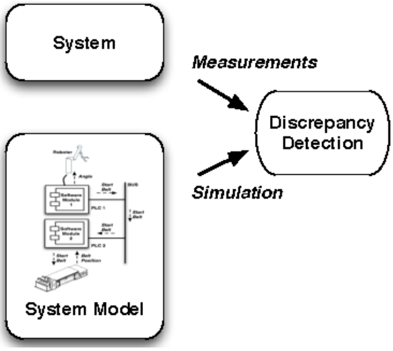


Fig. 14. Detection of system deterioration

This can also be seen in Fig. 14: During system runtime, the simulations of the model *M* are constantly compared to the system behaviour and discrepancies are detected.

Generally speaking, the offline simulation from section 4 can be used for the anomaly detection. There are three main differences. The first difference is a simulation slowdown so that the clock in the simulation matches the clock of the real world communication system. The second difference is the use of a specialised simulation component, which is able to read all process data (inclusive time stamps) from a real plant. The third difference is that the simulated process data and the real world process data are compared in every single simulation step. The idea is, that if the simulated process data are different to the real world process data, an anomaly is supposed to be detected (compare [28]).

This approach is currently tested for the application of anomaly detection: The predictions of the model are compared to measurements from the real plant; if a discrepancy occurs, an anomaly is signalled. For this, the MF discussed in section 4.3 is used.

For a first verification of the approach, we only use 3 of the plant modules and we define 3 types of errors: a blocked sensor, a sensor communication error, and the effects of deterioration of the conveyer belt. Then, for each of the 4 cases (3 errors, one OK case), we recorded 30 production cycles and tried to detect the errors. All errors were identified correctly.

5 Conclusions

Even though the development of the PIM editor and the Simulation Framework is still in progress, this software tool chain already enables a user to create an abstract and modular PIM model of a production plant and to store it in an AutomationML-based file format. AutomationML is well suited for this purpose because of its hierarchical CAEX structure and its extendable interface concept. These interfaces are currently used to describe the behaviour of the individual components in the plant model by state automatons as (self-executing) FMUs, but the modular concept of Simulation Components also allows for different approaches.

The usage of the FMI is at the moment a necessary requirement to execute these automatons in the Simulation Framework. The Simulation Framework reads the required communication connections from the AutomationML plant model and instantiates all needed Simulation Components with the corresponding input and output lists of process data. The process data are then managed by the Process Data Manager, which enables the individual Simulation Components to communicate with each other, and which provides them with the process data appropriate to the clock of the corresponding components.

For the purpose of demonstration, this tool chain has already been used for the offline simulation of a production plant, a HIL test of a single component (a PLC) and for the anomaly detection (diagnosis) in an operating production plant. Future work will extend the PIM editor with the functionality of mapping real hardware to the model. As a result, the editor will then be a PIM/PSM editor.

References

1. Brecher, C., Fedrowitz, C., Herfs, W., Kahmen, A., Lohse, W., Rathjen, O., Vitr, M.: Durchgängiges Production Engineering – Potenziale der digitalen Fabrik. In: Brecher, C., Schmitt, F.K.R., Schuh, G. (eds.) Wettbewerbsfaktor Produktionstechnik: Aachener Perspektiven, Aachen, AWK (2008)
2. Kühn, W.: Digital factory: simulation enhancing the product and production engineering process. In: WSC 2006: Proceedings of the 38th Conference on Winter Simulation, Winter Simulation Conference, pp. 1899–1906 (2006)
3. Gall, H.: Functional safety IEC 61508 / IEC 61511 the impact to certification and the user. In: AICCSA 2008: Proceedings of the 2008 IEEE/ACS International Conference on Computer Systems and Applications, pp. 1027–1031. IEEE Computer Society, Washington, DC (2008)
4. Steiner, P., Schmidt, F.: Anforderungen und Architektur zukünftiger Karosserieelektroniksysteme. VDI Berichte Nr. 1789 (2003)
5. Szulman, P., Assmann, D., Doerr, J., Eisenbarth, M., Hefke, M., Soto, M., Trifu, A.: Using ontology-based reference models in digital production engineering integration. In: Proceedings of the 16th IFAC WORLD CONGRESS, Prague (2005)
6. Stein, S., Kühne, S., Drawehn, J., Feja, S., Rotzoll, W.: Evaluation of OrViA Framework for Model-Driven SOA Implementations: An Industrial Case Study. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 310–325. Springer, Heidelberg (2008)
7. Strasser, T., Sunder, C., Valentini, A.: Model-driven embedded systems design environment for the industrial automation sector. In: Proceedings of the 6th IEEE International Conference on Industrial Informatics, Daejeon, South Korea (2008)
8. Niggemann, O., Stroop, J.: Models for model's sake: why explicit system models are also an end to themselves. In: ICSE 2008: Proceedings of the 30th International Conference on Software Engineering, pp. 561–570. ACM, New York (2008)
9. Niggemann, O., Otterbach, R.: Durchgehende Systemverifikation im Automotiven Entwicklungsprozess. In: Tagungsband des Dagstuhl Workshops Modellbasierte Entwicklung eingebetteter Systeme IV (MBEES), Schloss Dagstuhl, Germany (2008)
10. Streitferdt, D., Wendt, G., Nenninger, P., Nyssen, A., Lichter, H.: Model driven development challenges in the automation domain. In: 32nd Annual IEEE International Computer Software and Applications, COMPSAC 2008, pp. 1372–1375 (2008)
11. Drath, R., Weidemann, D., Lips, S., Hundt, L., Lüder, A., Schleipen, M.: Datenaustausch in der Anlagenplanung mit AutomationML. Springer (2010)
12. Modelica Association, Modelica (2011), `https://www.modelica.org/`
13. Allen, P. (ed.): The OMG'sModel Driven Architecture, volume XII of Component Development Strategies. The Monthly Newsletter from the Cutter Information Corp. on Managing and Developing Component-Based Systems (2002)
14. Mellor, S., Scott, K., Uhl, A., Weise, D.: MDA Distilled: Principles of Model-Driven Architecture. Addison-Wesley (2004)
15. PNO: Profinet specification iec 61158-5-10 (v2.1) (2007)
16. Graeser, O., Niggemann, O.: Planung der Übertragung von Echtzeitnachrichten in Netzwerken mit Bandbreitenreservierung am Beispiel von Profinet IRT. In: Echtzeit 2009 - Software-intensive verteilte Echtzeitsysteme GI-Fachauschuss, Boppard, Germany (2009)

17. IEC 62424: Festlegung für die Darstellung von Aufgaben der Prozessleittechnik in Fliessbildern und für den Datenaustausch zwischen EDV-Werkzeugen zur Fliessbilderstellung und CAE-Systemen (2008)
18. Khronos Group: Collada - 3d asset exchange schema (2011), `http://www.khronos.org/collada/`
19. PLCopen (2011), `http://www.plcopen.org/`
20. Modelisar project: Functional mock-up interface (2011), `http://functional-mockup-interface.org/`
21. MSDN: Visual studio visualization and modelling sdk vsvmsdk (2011), `http://code.msdn.microsoft.com/`
22. Kumar, B., Niggemann, O., Jasperneite, J.: Statistical models of network traffic. In: International Conference on Computer, Electrical and Systems Science (2010)
23. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P.-H., Nicollin, X., Olivero, A., Sifakis, J., Yovine, S.: The algorithmic analysis of hybrid systems. Theoretical Computer Science 138, 3–34 (1995)
24. Sproston, J.: Decidable model checking of probabilistic hybrid automata (2000)
25. DASSAULT SYSTEMES: Dymola (2011), `http://www.dymola.com`
26. Microsoft: Managed extensibility framework (2011), `http://mef.codeplex.com/`
27. Kumar, B., Niggemann, O., Jasperneite, J.: Test generation for hybrid, probabilistic control models. In: Entwurf komplexer Automatisierungssysteme (EKA 2010), Magdeburg, Germany (2010)
28. Supavatanakul, P., Lunze, J., Puig, V., Quevedo, J.: Diagnosis of timed automata: Theory and application to the damadics actuator benchmark problem. Control Engineering Practice 14(6), 609–619 (2006)

An Open-Loop Approach for a Stochastic Production Planning Problem with Remanufacturing Process

Oscar Salviano Silva Filho

Center for Information Technology Renato Archer
Rod. D. Pedro I (SP 65), Campinas-SP, Brazil
Oscar.salviano@cti.gov.br

Abstract. A stochastic linear production planning problem with chance-constraints is introduced in order to provide a production plan that optimizes a reverse logistics system. Such a system is composed of two channels: in the forward channel, new and remanufactured products are produced and stored into a serviceable unit, while, in the reverse channel, used products collected from market are stored into a returnable unit. The uncertainties about fluctuation of demand and amount of returnable are the reason of the stochastic nature of the problem. Since global optimal solution is not easy to be achieved, near-optimal solutions become viable alternatives to be explored. In this paper, an equivalent deterministic version of the stochastic problem is presented. This equivalent problem provides an open-loop solution that is used to provide production scenarios. An example illustrates the applicability of this methodology by creating two distinct production scenarios in order to answer the following question: "Is it possible to reduce the total production cost of a company by increasing the return rate of collecting, even knowing that the cost for remanufacturing is greater than the cost for manufacturing new products?

Keywords: Reverse Logistics, Supply Chain, Remanufacturing, Optimization.

1 Introduction

Planning and control the flow of used-products throughout the reverse channel has become increasingly crucial for the success of any supply chain. The reason of this is in the pressure that governments and society have imposed to companies for preservation of the environment. In this way, the increase of activities that involve processes of collecting and recovering is essential to reduce the necessity of new extractions of raw materials; and, as a consequence, these activities help to preserve the environment. Thus, reverse logistic issue has become an important topic of the supply chain area.

Conceptually, reverse logistics term can be understood as a process of dealing with different activities that involve, for instance: the collecting of used-products from the market for remanufacturing, recycling or disposing them. More generically, the objective of reverse logistics is to move used-products from the market to their final destination with the aim of recapturing value, or simply disposing.

In the literature, there are several papers related to logistic reverses issues. Most of them are based on quantitative models that are used to describe processes of

J.-L. Ferrier et al. (Eds.): Informatics in Control, Automation and Robotics, LNEE 174, pp. 211–225.
springerlink.com

collecting, recycling or remanufacturing in the reverse channel. Fleischmann et al [5] provide a typology of quantitative models for reverse logistics. It is based on three classes of problems: (i) reverse distribution problems; (ii) inventory-production control problems in systems with return flows; and (iii) production planning problem with reuse of parts and materials. In short, the first class of this typology is concerned with the collection and transportation of used-products and packages. According the authors: "the reverse distribution can take place through the original forward channel, through a separate reverse channel, or through combinations of the forward and the reverse channel". The second class is related to appropriate control mechanism that allows returning the used products into the market; and, the third class is associated with the planning of the reuse of items, parts and products without any additional process of remanufacturing.

This study is addressed to the second class of problems above cited. Particularly, the idea here is to formulate an inventory-production planning problem that provides optimal production scenarios, which allow the user to get insights about how to manage a reverse logistic system.

In the literature, inventory-production planning problems are formulated in deterministic or stochastic patterns. In the reverse logistics field, stochastic patterns are more common to find. Traditional techniques of stochastic inventory control theory can be reapplied for solving problems that include reverse systems. Based on this, the paper considers a sequential linear stochastic production planning problem with chance constraints on decision variables. Such a problem considers a stochastic production-inventory system that includes a special structure of collecting, remanufacturing and disposing. The stochastic nature of this system is a consequence of uncertainties related to demand fluctuation for serviceable products and return rate of used-products. It is assumed that these random variables are generated from uncorrelated normal stationary processes. Thus, as a result, the stochastic production-remanufacturing system is statistically well-defined by its mean and variance evolution over periods of the planning-horizon.

Since a global optimal solution for the stochastic problem is not easy to obtain, near-optimal solutions are viable alternatives. They are rough approximation of the optimal global solution, but are very important since they allow anticipating information about the system operation. Among the near-optimal solutions, the simplest one is the open-loop solution.

In this study, the certainty-equivalence principle is considered here to transform the stochastic problem into an equivalent deterministic problem, which provides an open-loop solution. As previously commented, this solution can be used to create production scenarios. These scenarios are developed by varying some parameters of the problem, such as: percentage of the return rate, lead time of collecting, customer satisfaction level, costs and so on.

An example is introduced to show the applicability of this methodology of modeling for solving a kind of production planning problem related to logistic reverse. In this context, two distinct production scenarios are developed to answer the following question: "Is it possible to reduce the total production cost of a company by increasing the return rate of collecting, even knowing that the cost for remanufacturing used-products is greater than the cost for manufacturing new products?" The first scenario considers that only 20% of used-products are effectively collected from the market, while the second considers a total of 100%.

The paper is stated as follows: initially, section 2 introduces brief comments on stochastic remanufacturing literature. In the section 3, a reverse logistic system is proposed. Follows then section 4 with formulation the stochastic problem. Next, section 5 presents the transformation process of the original stochastic problem to equivalent deterministic one. Finally, section 6 illustrates by an example the applicability of the matters discussed previously.

2 Literature Review

In recent years, companies have begun to seek for solutions that increase the life cycle of their products. Many reasons can explain such an interest, but the most important is that the scarcity of resources (i.e., the environment has a finite availability of raw-material). Thus, companies are rethinking their current ways of producing, introducing new processes to recover, recycle and remanufacture used products. As a consequence, planning and inventory control of these new reverse processes have become an essential task. Fortunately, all modeling and solving techniques that have been used to plan and control the production process in the forward channel can be reapplied to the reverse channel. A brief review of the literature is introduced now. Note that it highlights aspects of planning and inventory control of dynamic models that include remanufacturing processes.

Glancing at the literature, we find that the remanufacturing problems can be classified as deterministic and stochastic. This classification will depend on the hierarchical decision level where the problem is being formulated. For example, at the operational level (short term decisions) a good part of the problems are deterministic, while at the strategic level (long term decisions), they are stochastic. As mentioned in the last section, the production planning problem discussed in this paper follows a stochastic pattern. Some papers, directly or indirectly related to this problem are discussed in next paragraph, but it is worth mentioning that the processes of modeling and solving of problems are not necessarily similar to the one proposed here.

Some papers that are directly or indirectly related to the problem proposed here are due to: Shi et al. [13] discuss a stochastic production planning problem for a multi-product closed-loop system and solve it using a Lagrangian relaxation based approach; Wei et al. [14] propose an inventory control model for the return and remanufacturing processes under uncertain of the demand and returns. They propose a linear programming approach to deal with the uncertainty of the problem; Ahiska and King [1] use a Markov decision process to model and solve an inventory problem of single product recoverable manufacturing system; Inderfurth [7] considers a multi-period stochastic closed-loop supply chain with remanufacturing. A heuristic is proposed that allows evaluating this environment; Nakashima et al. [8] study a stochastic control problem of a remanufacturing system, which is modeled and solved via Markov decision process approach; and Dobos [4] formulates a quadratic production planning problem to deal with a reverse logistic system, and uses control theory to solve it.

3 A Reverse Logistics System

In this section, the mathematical model that represents a production-remanufacturing system is discussed. Figure 1 illustrates schematically how this system works, and it is possible to summarize this operation in three actions: first, there is a manufacturing unit that produces periodically lots of a single product. They are stored in the serviceable unit (storehouse 1) and wait there until to meet demand. Second, used-products are collected, periodically from the market, and stored to the returnable unit (storehouse 2). After an inspection process, they are released for remanufacturing or proper disposal. Third, after remanufacturing process, all recovered products are transferred to serviceable unit (see dashed line in the figure). Note also that a certain amount of products can be supplied by a subcontractor (i.e., a third-party vendor).

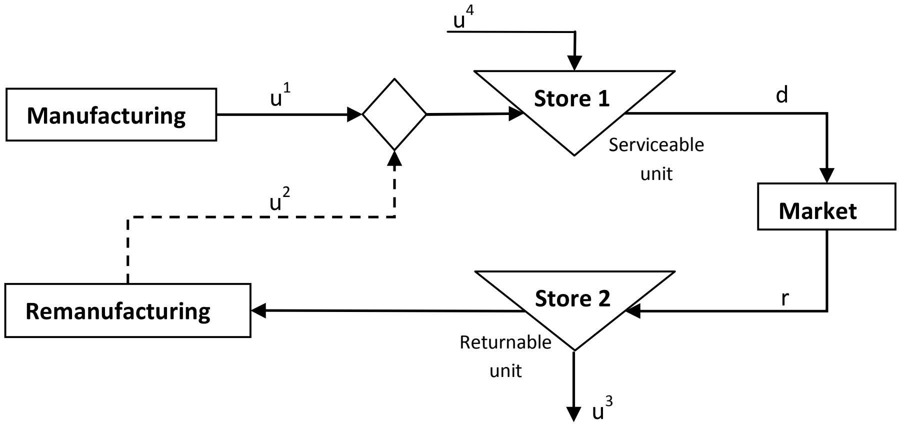


Fig. 1. Diagram of a closed-loop system with uncertain demand and return rate

Some features and properties of the system described by the Figure 1 are: (a) fluctuations of demand for serviceable products are met by the combination of both manufactured and remanufactured products; (b) demand level and return rate are non-correlated stationary random variables that follow normal distribution functions, whose their first and second statistical moments are perfectly known; (c) there are upper limits for both manufacturing and remanufacturing capacities, but not for inventory units; (d) there are two main reasons to discard used-products: the first has a technical justification, which is related to the quality of used-product, which may be inappropriate for remanufacturing. The second reason has a financial justification that says that remanufacturing a large quantity of used-products can overflow the serviceable unit causing unnecessary holding inventory costs; and (e) companies that manufacture bottles, cans, containers are typical examples of organizations that can adopt this kind of process.

4 The Stochastic Problem

The idea here is to formulate a linear-stochastic production planning problem with constraints that provides a sequential optimal plan that allows, in a strategic sense, operating the system exhibited by figure 1. Such a plan must encompass goals like: (i) an optimal inventory levels of serviceable and returnable units; (ii) an optimal production rates for manufacturing and remanufacturing processing units; (iii) an acceptable levels of subcontracting; and (iv) an estimative of disposal. Thus, this specific programming model is given by:

$$\underset{u_1,u_2,u_3,u_4}{Min}\ \mathrm{h}_1\mathrm{E}\left\{x_N^1\right\}+\mathrm{h}_2\mathrm{E}\left\{x_N^2\right\}+\sum_{\mathrm{k}=0}^{\mathrm{N}-1}\left\{\mathrm{h}_1\mathrm{E}\left\{x_k^1\right\}+\mathrm{h}_2\mathrm{E}\left\{x_k^2\right\}+c_1u_1+c_2u_2+c_3u_3+c_4u_4\right\}$$

s.t.

$$\begin{aligned}
&\mathrm{x}_\mathrm{k}^1 = x_0^1+\sum_{i=1}^{k}\left[u_i^1+u_i^2+u_i^4\right]-D_k && (a)\\
&\mathrm{x}_\mathrm{k}^2 = x_0^2-\sum_{i=1}^{k}\left(u_i^2+u_i^3\right)+R_k && (b)\\
&\mathrm{Prob.}(\mathrm{x}_\mathrm{k}^1 \geq 0)\geq\alpha && (c)\\
&\mathrm{Prob.}(\mathrm{x}_\mathrm{k}^2 \geq 0)\geq\beta && (d)\\
&0\leq \mathrm{u}_\mathrm{k}^1\leq\bar{u}_1;\ 0\leq \mathrm{u}_\mathrm{k}^2\leq\bar{u}_2;\ \mathrm{u}_\mathrm{k}^3\geq 0;\ 0\leq \mathrm{u}_\mathrm{k}^4\leq\bar{u}_4 && (\mathrm{e})
\end{aligned} \qquad (1)$$

where the notation of variables and parameters of problem (1) are given as follows:

x_k^1	-	serviceable inventory level at period k;
x_k^2	-	returnable inventory level at period k;
u_k^1	-	manufacturing rate at period k;
u_k^2	-	remanufacturing rate at period k;
u_k^3	-	disposal rate at period k;
u_k^4	-	subcontracting level at period k;
d_k	-	demand level at period k;
r_k	-	return rate at period k;
D_k	-	cumulative level of demand at period k;
R_k	-	cumulative return rate at period k;
h_1	-	serviceable holding cost;
h2	-	remanufacturable holding cost;
c_1	-	production cost;
c_2	-	remanufacturing cost;
c_3	-	disposal cost;
c_4	-	subcontracting cost; and

α; β – probabilistic indexes that belong to interval [0,1).

Note also that $\mathrm{D}_\mathrm{k}=\sum_{\mathrm{i}=1}^{\mathrm{k}}\mathrm{d}_\mathrm{i}$ and $\mathrm{R}_\mathrm{k}=\sum_{\mathrm{i}=1}^{\mathrm{k}}\mathrm{r}_\mathrm{i}$ denote cumulative random variables. The first variable is related to demand level d_i (i.e. number of orders placed at period i); and the second cumulative variable is related to number of returnable products r_i (i.e., return rate). It is assumed that these random variables are stationary, non correlated and they are completely defined by normally distributed probability functions denoted as Φ_D and Φ_R . Theses distributions are defined by their first and second statistical moments:

$$\begin{cases} \hat{D}_k = \sum_{i=1}^{k} \hat{d}_i ; & \Theta_k^D = \sqrt{k}\,\sigma_d \\ \hat{R}_k = \sum_{i=1}^{k} \hat{r}_i ; & \Theta_k^R = \sqrt{k}\,\sigma_r \end{cases} \tag{2}$$

where the pairs $(\hat{d}_i, \hat{r}_i)$ and (σ_d, σ_r) denote, respectively, mean values and finite standard deviations of demand level d_i and return rate r_i.

It is worth emphasizing that the normal distribution is usually a good approximation for a variety of other types of distributions. For example, the Poisson distribution is commonly used to represent events that involve arrivals; this is the case of the number of orders that arrival over periods of planning horizon (i.e. demand fluctuation). Whenever the flow of arrive becomes intense, the Poisson process can be approximated by a Normal process; see [3], [4]and [12].

At last, it is important to say that the problem (1) follows economic aspects of an ordinary productive environment. In other words, this means that a reverse process implementation is economically viable when its costs associated with transport, holding inventory and production are acceptable in terms of the budget of the company. For instance, if the cost of remanufacturing is less than the cost of manufacturing new products, the chances of reverse logistic system to be economically viable is relatively high. Besides, it is important to have in mind that costs with raw-material and components to manufacture new products can increase significantly the production cost of new products. In short, the production/ manufacturing policy, provided by problem (1), must be interpreted from cost viewpoint, in order to conclude something about its economic viability.

4.1 A Brief Note on the Solution of the Problem (1)

Finding an optimal global solution to the problem (1) is not an easy task [2]. Two main reasons of this difficulty are the stochastic nature of the inventory balance system and chance-constraints on inventory variables. As a consequence, equivalent versions of this problem can be formulated in order to provide near optimal solutions. Thus, these suboptimal solutions are rough approximation of the optimal global solution, but are very important because they allow anticipating information about the use of resources to operate the system (1.a)-(1.b).

The normality assumption made to random variables of problem (1) associated with the certainty-equivalence principle allows transforming the problem (1) into an equivalent, but deterministic problem, which is entirely based on the first and second statistical moments of random variables; see [12]. This equivalent formulation can be used to provide an optimal open-loop production plan. In the section 5.5, this kind of suboptimal solution will be discussed.

5 Transformation Process

The certainty-equivalence principle is commonly employed to reduce the complexity of stochastic problems as that formulated in (1). Such a principle considers that the mean value problem is a reasonable approximation for the problem (1). This deterministic problem is formulated on the assumption that all random variables of the problem (1) are replaced by their average values.

For this study, first and second statistical moments of the random variables of the systems (1.a)-(1.b) are considered in the transformation process. Note that the probability distributions of x_k^1 and x_k^2 (given by Φ_{x^1} and Φ_{x^2}) are originated from a normal process, whose mean and variance are known and given by $\left(\text{i.e., } \hat{x}_k^i = E\left\{x_k^i\right\}; \ i = 1,2\right)$ $\left(\text{i.e., } \sigma_{x_k^i}^2 = E\left\{(x_k^i - \hat{x}_k^i)^2\right\}; \ i = 1,2\right)$ respectively. It is important to realize that, in reason of the linearity of the systems (1.a)-(1.b), the distributions of probability Φ_{x^1} and Φ_{x^2} are similar to distribution of probability of Φ_D and Φ_R , described by (2).

5.1 The Equivalent System

For the transformation process, the mathematical expectation operator (i.e. E{.}) is applied to equations (1.a) and (1.b). This is equivalent to replace the random variables of these equations by their mean values. As a result follows then:

$$\begin{aligned} \hat{x}_k^1 &= \hat{x}_0^1 + \sum_{i=1}^{k}\left[u_i^1 + u_i^2 + u_i^4\right] - \hat{D}_k \qquad &(a)\\ \hat{x}_k^2 &= \hat{x}_0^2 - \sum_{i=1}^{k}\left(u_i^2 + u_i^3\right) + \hat{R}_k \qquad &(b) \end{aligned} \qquad (3)$$

where $\hat{x}_k^1 = E\left\{x_k^1\right\}$ and $\hat{x}_k^2 = E\left\{x_k^2\right\}$ are the expected levels of serviceable and remanufacturable units. Note also that $\hat{x}_0^1 = x_0^1$ and $\hat{x}_0^2 = x_0^2$ are known and provided a prior by manager.

5.2 The Equivalent Criterion

Similar to 5.1, the mathematical expectation operator is applied to the criterion of the problem (1):

$$Z = h_1 E\left\{x_N^1\right\} + h_2 E\left\{x_N^2\right\} + \sum_{k=0}^{N-1}\left\{h_1\, E\left\{x_k^1\right\} + h_2 E\left\{x_k^2\right\} + c_1 u_1 + c_2 u_2 + c_3 u_3 + c_4 u_4\right\}$$

As a consequence of the linearity of the criterion, results that:

$$\hat{Z}=h_1\hat{x}_N^1+h_2\hat{x}_N^2+\sum_{k=0}^{N-1}\left\{h_1\hat{x}_k^1+h_1\hat{x}_k^2+c_1u_1+c_2u_2+c_3u_3+c_4u_4\right\} \tag{4}$$

where $\hat{x}_k^1$ and $\hat{x}_k^2$ follow the same description given above.

5.3 The Equivalent Constraints

The transformation of chance-constraints (1.c) and (1.d) to deterministic-equivalent formats is performed as follows: Firstly, using the inventory balance equation (1.a), the chance-constraint (1.c) can be rewritten as follows:

$$\text{Prob.}(x_0^1+y_k\geq D_k)\geq\alpha \tag{5}$$

or, also, in the form:

$$\text{Prob.}(\varepsilon_d\leq\frac{x_0^1+y_k-\hat{D}_k}{\Theta_k^D})\geq\alpha \tag{6}$$

where $y_k=\sum_{i=1}^{k}\left[(1-\rho)u_i^1+u_i^2\right]$ and ε_d is a normal random variable related to the variable D_k. Thus, the variable ε_d have the same cumulative distribution function (c.d.f) of D_k, denoted here as Φ_D.

From statistical theory, the constraint (6) can be easily handled and rewritten as:

$$\frac{x_0^1+y_k-\hat{D}_k}{\Theta_k^D}\geq\Phi_D^{-1}(\alpha)\quad\Rightarrow\quad x_0^1+y_k-\hat{D}_k\geq\Theta_k^D\cdot\Phi_D^{-1}(\alpha) \tag{7}$$

Taking into account the deterministic system (3.a) and remembering that $x_0^1=\hat{x}_0^1$ is given, it is possible to show that $\hat{x}_k^1=x_0^1+y_k-\hat{D}_k$. Thus, follows from (7) that the equivalent constraint related to (1.c) is given by:

$$x_k^1\geq\Theta_k^D\cdot\Phi_D^{-1}(\alpha)\ \Rightarrow\ x_k^1\geq\sqrt{k}\cdot\sigma_d\cdot\Phi_D^{-1}(\alpha) \tag{8}$$

where Φ_D^{-1} (.) is the inverse distribution of probability of the variable D_k.

Proceeding in a similar way for the chance-constraints (1.d), it is found that

$$x_k^2\geq\Theta_k^R\cdot\Phi_R^{-1}(\beta)\ \Rightarrow\ x_k^2\geq\sqrt{k}\cdot\sigma_r\cdot\Phi_R^{-1}(\beta) \tag{9}$$

where Φ_R^{-1} (.) is the inverse distribution of probability of variable R_k.

5.4 The Equivalent Problem

Gathering together all the transformed parts above, an equivalent deterministic problem is then formulated to represent the stochastic problem (1) as follows:

$$\underset{u_1,u_2,u_3}{\text{Min}}\ \hat{Z}=h_1\hat{x}_N^1 + h_2\hat{x}_N^2 + \sum_{k=0}^{N-1}\left\{h_1\hat{x}_k^1 + h_2\hat{x}_k^2 + c_1u_1 + c_2u_2 + c_3u_3 + c_4u_4\right\}$$

s.t.

$$\begin{aligned}
&\hat{x}_k^1 = \hat{x}_0^1 + \sum_{i=1}^{k}\left[u_i^1 + u_i^2 + u_i^4\right] - \hat{D}_k \\
&\hat{x}_k^2 = \hat{x}_0^2 - \sum_{i=1}^{k}\left(u_i^2 + u_i^3\right) + \hat{R}_k \\
&\hat{x}_k^1 \geq \sqrt{k}\cdot\sigma_d\cdot\Phi_D^{-1}(\alpha) \\
&\quad \hat{x}_k^2 \geq \sqrt{k}\cdot\sigma_r\cdot\Phi_R^{-1}(\beta) \\
&0 \leq u_k^1 \leq \bar{u}_1;\ \ 0 \leq u_k^2 \leq \bar{u}_2;\ \ u_k^3 \geq 0;\ \ 0 \leq u_k^4 \leq \bar{u}_4
\end{aligned} \tag{10}$$

Note that classical techniques of linear programming, available in the literature can be applied to provide static solution.

5.5 A Note on Suboptimal Approaches

As said previously, the problem (10) can be solved by any available linear programming algorithm. As a result, a suboptimal open-loop solution can be provided. However, the "structure" of this open-loop solution can change in reason of the type suboptimal approach used to solve (10). For instance, there are two very simple approaches to deal with (10); they are known, in the literature, as Open-loop No-updating and Open-loop Updating, see [9]. A brief description of them follows next.

The Open-Loop No-updating (OLN) approach applied to (10) provides a sequential optimal plan that depends only on the initial states of the systems (1.a) and (1.b), that is, x_0^1 and x_0^2 known a prior. Other information about the states of the systems (1.a) and (1.b) available for k>0 periods are completely ignored. As a consequence, OLN approach does not take into account new information of the systems when it provides a feasible solution for the problem (10).

The Open-Loop Updating (OLU) approach is very similar to the OLN, except that sequential optimal production plan is periodically adjusted as soon as new information about the states of the systems (1.a) and (1.b) becomes available. In fact, for each period k, the current states of the systems (1.a) and (1.b) are measured and used to solve the problem (10).Thus, before providing an optimal sequential plan, the problem (10) must be repeated N times.

Both OLN and OLU approaches can be used to develop production scenarios. These scenarios are very useful for managers to get insights about the use of the resources of the company. These scenarios are developed by varying some parameters of the problem, such as: percentage of the return rate, lead time of collecting, customer satisfaction level, costs and so on.

6 An Example

Let's consider here a company that have a closed-loop production system as shown in the Figure 1. In the forward direction of this system, manufactured, remanufactured and subcontracted products are regularly stored into a serviceable unit to meet weekly demands. On the other hand, in the backward direction, used-products are weekly collected from the market and stored into returnable unit. After inspected, they are sent to remanufacture processor unit or proper disposal.

The weekly demand for serviceable products and the weekly return rate of used-products are non correlated random variables that follow stationary processes. Such processes are well-defined by mean of normally distributed probability functions that are completely described by their first and second statistics moments as given in (2).

The company wants to invest more in their reverse chain. She has no interest in expanding its current production capacity, but aims to reduce costs, particularly with the acquisition of third party products. In this way, the company wants to answer the following question: "is it possible to reduce the total production cost of a company by increasing the return rate of collecting, even knowing that the cost for remanufacturing used-products is greater than the cost for manufacturing new products?". More specifically, this question can be complemented by another question: what is the impact of the increase of the return rate in the performance of system (1.a)-(1.b)?

To carry out such an analysis and answer these questions, the company decide to apply the OLN approach to solve the problem (10) under a planning horizon of twelve weeks (i.e. N=12). Table 1 resumes the main data of the problem:

Table 1. Problem's data

k (week)	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th
$\hat{d}_k$	105	98	101	102	97	99	101	105	101	102	97	99

Standard-deviations:	$\sigma_d = 2$ and $\sigma_r = 3$
Initial inventories:	$x_0^1 = 20$ and $x_0^2 = 20$
Serviceable and returnable costs:	$h_1 = \$\,2{,}00$ and $h_2 = \$\,1{,}00$
Manufacturing and remanufacturing cost:	$c_1 = \$\,1{,}00$ and $c_2 = \$\,1{,}20$
Disposal cost:	$c_3 = \$\,0{,}14$
Subcontracting cost:	$c_4 = \$\,2{,}00$
Upper bounds of manufacturing:	$\bar{u}_1 = 60$
Maximum number of subcontracting	$\bar{u}_4 = 30$
Probabilistic indexes:	$\alpha = 90\%$ and $\beta = 80\%$
Others parameters: $\bar{x}_a = \bar{x}_2 = \bar{u}_2 = \bar{u}_3 = \infty$ (i.e., they are not limited)	

Based on these data, two scenarios will be constructed as a result of solution of the problem (10): in the first one, it is considered that the weekly mean value level of the return rate $\hat{r}_k$ is around 20% of the mean value level of demand $\hat{d}_k$. This means that the flow of returnable products is 20% smaller than the flow of orders placed by customers (i.e. demand level for serviceable products). On the other hand, the second scenario considers that the weekly mean value of return rate is equivalent to weekly mean value level of demand (i.e., 100% of returnable products is expected).

It is worth mentioning that the weekly mean value of return rate $\hat{r}_k$ is given by:

$$\hat{r}_k = \rho \cdot \hat{d}_k + \sigma_r \cdot \varepsilon_k \tag{11}$$

where ε_k is a normally distributed pseudorandom numbers; and ρ denotes the percentage rate of average demand d_k. Note that, for the scenarios analysis purpose, the sequence of values of $\{\varepsilon_1, \varepsilon_{2,} ..., \varepsilon_{12}\}$, which was generated by a random device, is set a prior.

6.1 1st Scenario: Around ρ=20% Returnable

Figures 2 and 3 illustrate optimal inventory-production trajectories for forward and reverse channels of the system illustrated generically in the Figure 1. It is interesting to note that inventory levels for serviceable and returnable units show similar tendencies in their optimal trajectories. In fact, except for the first week (k=1), when inventory-production process (1.a) and (1.b) are still being adjusted to the initial conditions of the problem (i.e., $x_0^1 = x_0^2 = 20$), the subsequent weeks (k>1) show a continuous growing of inventory levels in the service and returnable units. Such a result is caused by the dynamic behavior of deterministic constraints (8) and (9) that depend directly on the period k and the probabilistic indexed α and β. Note that right side of these inequalities represent safety-stock levels that increase with periods k. This feature of the problem (10) aims to protect the systems (1.a) and (1.b) against backorders and backlogs occurrences, respectively. For instance, the idea of a safety-stock for a serviceable unit aims to guarantee ready delivery of products to customers; and, a safety-stock in the returnable unit aims to provide a buffer of used products to be remanufactured in future periods, protecting the reverse channel against delays of collecting.

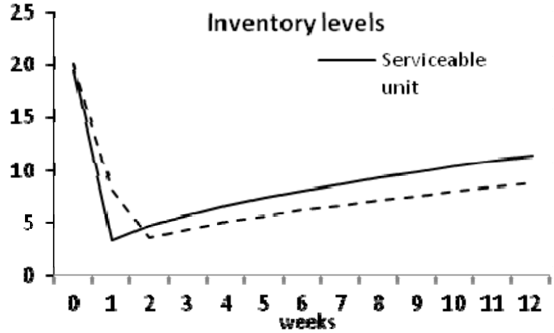


Fig. 2. Serviceable and remanufacturable levels

The figure 3 shows the weekly evolution of optimal rates of manufacturing and remanufacturing processing units, as well as, the weekly level of subcontracted products from third-part supplier. It is interesting to observe that the manufacturing unit operates at full load. Indeed, the production of new products is made in the maximum weekly level (i.e., $\hat{x}_k^1 = \overline{x}^1 = 60; k \geq 1$). In order to meet the weekly demand, manufactured and subcontracted products are considerer complimentarily, but the monthly rate of remanufactured also operates at its maximum value that is around 20% of mean value level of weekly demand.

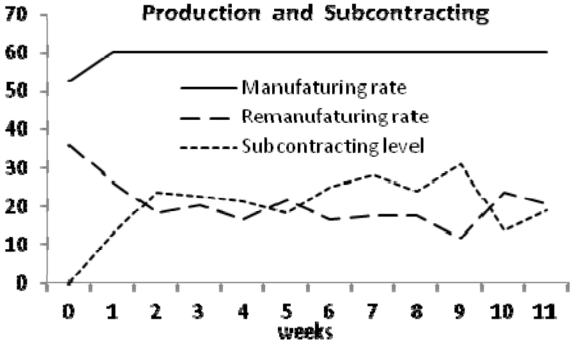


Fig. 3. Production rates

6.2 2nd Scenario: Around ρ=100% Returnable

In this scenario, the weekly return rate of used-products varies around the weekly level of fluctuation of demand for new products, that is, $\hat{r}_k \approx \hat{d}_k$. The optimal inventory and production trajectories for forward and reverse channels are respectively exhibited in Figures 4 and 5.

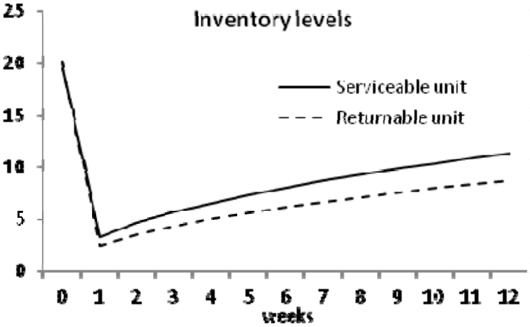


Fig. 4. Levels of inventory with 100% of returnable

Note that the inventory trajectories given by Figure 4 are similar to inventory trajectories showed in Figure 2, but with the difference that now they are a little bit closer. Once again it is important to emphasize that such a behavior is a typical characteristic of the problem (10) that creates safety-stock to protect the inventory units of systems (1.a)-(1.b).

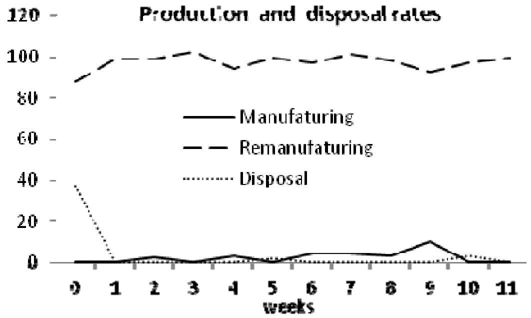


Fig. 5. Production and disposal rates

In contrast with the weekly behavior of inventories trajectories that do not change with the increase of the return rate (from ρ=20% to ρ=100%), the weekly behavior of manufacturing, remanufacturing, subcontracting and disposal variables change completely. In fact, while in the first scenario the weekly amount of remanufactured and subcontracted products just only complements the level of new manufactured products, in the scenarios 2 (see figure 5) the weekly level of remanufactured products meet near-completely the weekly demand. Note that the company practically stopped to manufacture new products and buy products from third-party suppliers. In short, this means that the company is only operating with its reverse channel.

6.3 Costs

Table 2 contains the costs associated with two scenarios.

Table 2. Total costs

Costs	**1st Scenario**	**2nd Scenario**
Serviceable holding	232,44	232,44
Remanufacturing holding	99,43	93,85
Manufactured rate	712,39	28,33
Remanufactured rate	295,50	1404,10
Disposal rate	0,00	21,09
Subcontracting	479,53	0,00
Total cost	**1.819,29**	**1.779,81**

Note that holding costs for serviceable inventory unit are exactly the same for the two scenarios, but for the returnable inventory unit, these costs are only very close. This explains why the optimal inventory trajectories (shown in Figures 2 and 4) did not vary when the return rate increases.

Note also that in the second scenario a major part of the cost is concentrated in the remanufacturing process. Even considering that the cost of remanufacturing used-products is 20% greater than the cost of manufacturing new-products, the total optimal cost provided by scenario 2 was 2.2% lower than the total cost of scenario 1.

Based on the above results, it is possible to answer the question proposed initially by company's manager: yes, it is possible to adopt a remanufacturing scheme based on high rate of returnable products that is profitable for the company, even when the cost of remanufacturing is greater than the cost of manufacturing (i.e. $c_2>c_1$). However, the same result shows that, in this case, the manufacturing process may have to be excluded from the closed-loop system given in Figure 1.

At last, it is important to say that one of advantage of the second scenario is the reduction of costs with purchase of raw-material required to manufacture new products. In fact, the increase of the return rate of used-products, together with the increase of amount of remanufactured products, may result in a significant reduction of new products being manufactured, which implies probably in a big reduction of inventory level of raw-material.

6.4 Further Extensions of the Model

Future studies include an extended version of the stochastic planning problem (1). Such a version should be idealized to allow a greater realism in the formulation. Thus, some desirable improvements are, for instance: a) to consider non-linear costs, but assuring convexity properties that allow the transformation process discussed in section 5; b) to include upper storage limits for serviceable and remanufacturable inventory units; c) to consider multi-products, and to include new constraints on the remanufacturing and manufacturing processes; and d) to implement a sequential open-loop updating (OLU) approach to solve (10), as discussed in the section 5.5. This approach is also known as a rolling horizon technique, which allows optimal adjustment of the generated plans in order to follow actual demand fluctuations during each period of the planning horizon, see [10].

With these improvements, the model will be closer to reality, and, as a result, it will be possible to develop more accurate studies, and to provide more efficient plans for management purposes.

7 Conclusions

A linear stochastic inventory-production planning problem with constraints on main decision variables was proposed to model a closed-loop system that includes a reverse channel.

The main difficulty here is that a global optimal solution to the stochastic problem is very complex. Thus, a suboptimal alternative that includes first and second statistical moments of the random variables of the system was proposed to convert the stochastic problem into an equivalent deterministic. As a result, an open-loop optimal solution was provided that allowed creating productions scenarios.

An example was proposed to illustrate the applicability of the use of the equivalent deterministic problem to create production scenarios. Thus, using the model proposed, it was possible to answer the following question considered in the beginning of this paper: is it possible to reduce the total production cost of a company by increasing the return rate of collecting, even knowing that the cost for remanufacturing is greater than the cost for manufacturing new products?. The answer is simply “yes”, but it was found some consequences of this that is discussed in the section 6.3.2.

In short, this model seems to be an interesting management tool not only to develop a production plan for use within the hierarchy of business decisions, but also to allow managers to get insights about the use of production resources.

References

1. Ahiska, S.S., King, R.: Inventory optimization in a one product recoverable manufacturing system. Int. J. Production Economics 124, 11–19 (2010)
2. Bertsekas, D.P.: Dynamic programming and optimal control, vol. 1. Athena Scientific, USA (2007)
3. DasGupta, A.: Fundamentals of probability: a First course, Springer Texts in Statistics. ch. 10, pp. 229–231. Springer (2010)
4. Dobos, I.: Optimal production-inventory strategies for HMMS-type reverse logistics system. Int. J. Production Economics 81-82, 351–360 (2003)
5. Fleischmann, M., Bloemhof-Ruwaard, J.M., Dekker, R., Van der Laan, E., Van Nunen Jo, A.E.E., Van Wassenhove, L.N.: Quantitative models for reverse logistics: A review. European Journal of Operational Research 103, 1–17 (1997)
6. Graves, S.C.: A single-item inventory model for a non-stationary demand process. Manufacturing & Service Operations Management 1(1) (1999)
7. Inderfurthn, K.: Impact of uncertainties on recovery behavior in a remanufacturing environment. International Journal of Physical Distribution & Logistics Management 3(5), 318–336 (2005)
8. Nakashima, K., Arimitsu, H., Nose, T., Kuriyama, S.: Optimal control f a remanufacturing system, Int. J. Int. J. of Production Research 42(17), 3619–3625 (2004)
9. Pekelman, D., Rausser, G.C.: Adaptive Control: Survey of Methods and Applications, Applied Optimal Control. TIMS Studies in the Management Science, vol. 9, pp. 89–120. North-Holland (1978)
10. Pereira, F.B., De Sousa, J.B.: On the receding horizon hierarchical optimal control of manufacturing systems. Journal of Intelligent Manufacturing 8, 425–433 (1997)
11. Roy, A., Maity, K., Kar, S., Maiti, M.: A production- inventory model with manufacturing for defective and usable items in fuzzy-environment. Computers & Industrial Engineering 59, 87–96 (2009)
12. Silva Filho, O.S.: An open-loop solution for a stochastic production-remanufacturing planning problem In: Proceeding of 8th International Conference on Informatics in Control, Automation and Robotics, Noordwijkerhout, Holland, July 28-31, pp. 28–31 (2011)
13. Shi, J., Zhang, G., Sha, J.: Optimal production planning for a multi-product closed loops system with uncertain demand and return. Computers & Operations Research 38, 641–650 (2011)
14. Wei, C., Li, Y., Cai, X.: Robust optimal policies of production and inventory with uncertain returns and remand. International Journal of Production Economics (2010), doi:10.1016

Author Index