Advances in Intelligent Systems and Computing 386

Zdzisław Kowalczuk *Editor*

Advanced and Intelligent Computations in Diagnosis and Control

Springer

Advances in Intelligent Systems and Computing

Volume 386

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Advisory Board

More information about this series at http://www.springer.com/series/11156

Zdzisław Kowalczuk
Editor

Advanced and Intelligent Computations in Diagnosis and Control

Springer

Editor
Zdzisław Kowalczuk
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gdańsk
Poland

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-23179-2 ISBN 978-3-319-23180-8 (eBook)
DOI 10.1007/978-3-319-23180-8

Library of Congress Control Number: 2015946755

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Preface

Since the late 1980s technical diagnostics has been an area of major scientific interest and serious research. It covers many established and emerging topics in control and systems engineering, robotics, aerospace, applied mathematics and statistics, decision sciences, signal processing, and artificial intelligence, to mention only a few of the most widely recognized. The developments have been followed by a great number of applications of fault diagnosis methods in both industrial and medical areas. Certainly, the increasing complexity of automation and systems, and the need to ensure the highest level of reliability and safety require continuing research and the development of innovative approaches to fault diagnosis, including fault-tolerant, and reconfigurable control.

This book contains selected papers presented at the 12th International Conference on *Diagnostics of Processes and Systems* (DPS), held in Ustka, Poland, from 6 to 9 September 2015. The conference was organized by the Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics, Dept. of Robotics and Decision Systems, with the support of the University of Zielona Góra and the Warsaw University of Technology.

The previous conferences took place in Podkowa Leśna (1996), Łagów Lubuski (1997), Jurata (1998), Kazimierz Dolny (1999), Łagów Lubuski (2001), Władysławowo (2003), Rajgród (2005), Słubice (2007), Gdańsk (2009), Zamość (2011), and Łagów Lubuski (2013) and attracted a large number of participants and internationally recognized speakers. In fact, the conference series is a two-year continuation of annual conferences on *Diagnostics of Industrial Processes* organized in the years 1996–1999 by those three Polish universities.

The main subject matter of the *DPS* conference series is related to the demands of research and industrial centers for diagnostics, monitoring, and decision-making systems. DPS welcomes combinations of domains of engineering knowledge for diagnosis, including detection, isolation, localization, identification, diagnostics, reconfiguration, and control. The conference is open to new challenges, including medical and industrial diagnosis, computer systems diagnosis, and non-industrial applications, providing a forum for exchanging experience and for sharing solutions between the academic and industrial communities.

There are thus six principal topics of DPS interests: (i) Fault detection, isolation and identification; (ii) Fault-tolerant control systems; (iii) Process safety, quality and reliability; and (iv) Medical diagnostics; as well as (v) methodologies based on mathematical modeling, parameter identification and state estimation, qualitative models, statistical and signal processing, artificial intelligence, fuzzy logic and rough sets, expert systems, neural networks; and (vi) industrial applications of diagnostic systems in fault tolerant problems, safety, monitoring and alarming, quality control, computer systems and networks, diagnostic software, software reliability, medicine and therapy, environment protection, production control, and other industries, chemistry, electronics, and power systems, etc.

The subject area of the DPS conferences corresponds to the topic of the IFAC symposia on *Fault Detection, Supervision and Safety for Technical Processes* (*SAFEPROCESS*), as well as to the newly established international conferences on *Control and Fault-Tolerant Systems*.

The book is divided into six parts:

Part I Fault Detection and Isolation
Part II Estimation and Identification
Part III Robust and Fault Tolerant Control
Part IV Industrial and Medical Diagnostics
Part V Artificial Intelligence
Part VI Expert and Computer Systems.

I wish to thank all participants, and reviewers from the International Program Committee for their scientific and personal contribution to the conference.

My particular appreciation goes to the authors of the papers published in this collective book of Springer and the other book of PWNT Gdańsk, as well as to the plenary speakers for delivering their lectures:

- Henrik Niemann, Technical University of Denmark: *Fault Diagnosis by using Modification of the Feedback Controller*
- Michał Bartyś, Warsaw University of Technology: *Diagnosing Single and Multiple Faults from FDI Perspective*
- Paolo Castaldi, Nicola Mimmo, Silvio Simani, Universities of Bologna and Ferrara: *Issues of Fault Diagnosis and Fault Tolerant Control for Aero-space Systems*
- Philippe Weber and Didier Theilliol, University of Lorraine: *Bayesian Networks Application to the Dependability of Multi-State Systems*
- Youmin Zhang, Concordia University: *Challenges and Development on Fault Diagnosis and Fault-Tolerant Cooperative Control Techniques with Applications to Unmanned Systems*.

Acknowledgments

I am grateful to the DPS 2015 Organizing Committee Chair, Mariusz Domżalski, for his and his co-workers' effort put into making the conference a successful scientific event. I would also like to acknowledge the editorial support of Maria Kowalczuk, Janusz Kozłowski, Marek Tatara, and the technical and administrative support of Anna Osadowska and Michał Czubenko.

Gdańsk Zdzisław Kowalczuk
June 2015

Organization

DPS 2015 was organized by Polish Consultants Society, the Gdańsk Branch, and the Department of Robotics and Decision Systems, Faculty of Electronics, Telecommunications and Informatics of the Gdańsk University of Technology, Poland.

DPS Program Committee Chairs

General Chair
Zdzisław Kowalczuk, Gdańsk University of Technology

Co-chairs
Jan M. Kościelny, Warszawa University of Technology
Józef Korbicz, University of Zielona Góra

DPS Program Committee

Christophe Aubrun (Nancy)
Jerzy Balicki (Gdańsk)
Andrzej Bartoszewicz (Łódź)
Wojciech Batko (Kraków)
Gildas Besancon (Grenoble)
Piotr Bielawski (Szczecin)
Liliana Byczkowska-Lipińska (Łódź)
Joao Calado (Lisbon)
Eduardo Camacho (Sevilla)
Alessandro Casavola (Cosenza)
Paolo Castaldi (Bologna)
Czesław Cempel (Poznań)

Wojciech Cholewa (Gliwice)
Piotr Chrzan (Gdańsk)
Vincent Cocquempot (Lille)
Steven Ding (Duisburg)
Stefan Domek (Szczecin)
Krzysztof Fujarewicz (Gliwice)
Janos Gertler (Fairfax)
Jerzy Głuch (Gdańsk)
Jan Sokołowski (Nancy)
Wojciech Jędruch (Gdańsk)
Ireneusz Jóźwiak (Wrocław)
Jacek Kabziński (Łódź)
Janusz Kacprzyk (Warszawa)
Andrzej Kasiński (Poznań)
Roman Kaszyński (Szczecin)
Jacek Kluska (Rzeszów)
Józef Korbicz (Zielona Góra)
Kazimierz Kosmowski (Gdańsk)
Jan M. Kościelny (Warszawa)
Zdzisław Kowalczuk (Gdańsk)
Krzysztof Kozłowski (Poznań)
Dušan Krokavec (Kosice)
Andrzej Królikowski (Poznań)
Piotr Kulczycki (Kraków)
Antoni Ligęza (Kraków)
Jan Lunze (Bohum)
Jan M. Maciejowski (Cambridge)
Krzysztof Malinowski (Warszawa)
Letitia Mirea (Iasi)
Wojciech Mitkowski (Kraków)
Wojciech Moczulski (Gliwice)
Henrik Niemann (Lyngby)
Mariusz Nieniewski (Warszawa)
Andrew Ordys (London)
Cezary Orłowski (Gdańsk)
Stanislaw Osowski (Warszawa)
Krzysztof Patan (Zielona Góra)
Andrzej Pieczyński (Zielona Góra)
Vicenc Puig (Barcelona)
Joseba Quevedo (Barcelona)
Ewaryst Rafajłowicz (Wrocław)
Leszek Rutkowski (Częstochowa)
Jose Sa da Costa (Lisbon)
Dominique Sauter (Nancy)
Jurek Sąsiadek (Ottawa)

Silvio Simani (Ferrara)
Piotr Skrzypczyński (Poznań)
Janusz Sosnowski (Warszawa)
Piotr Szczepaniak (Łódź)
Roman Śmierzchalski (Gdańsk)
Mirosław Świercz (Białystok)
Andrzej Świerniak (Gliwice)
Piotr Tatjewski (Warszawa)
Didier Theilliol (Nancy)
Leszek Trybus (Rzeszów)
Dariusz Ucinski (Zielona Góra)
Tadeusz Uhl (Kraków)
Wiesław Wajs (Kraków)
Bogdan Wiszniewski (Gdańsk)
Marcin Witczak (Zielona Góra)
Marian Wysocki (Rzeszów)
Youmin Zhang (Montreal)
Cezary Zieliński (Warszawa)
Jacek M. Żurada (Louisville)

DPS Organizing Committee

Chair
Mariusz Domżalski

Vice-Chair
Michał Czubenko

Secretary
Anna Osadowska

Patrons and Sponsoring Institutions

Polish Consultants Society
Gdańsk University of Technology
Faculty of Electronics, Telecommunications and Informatics
Polish Academy of Sciences, Automatic Control and Robotics Committee
Polish Society for Measurement Automatic Control and Robotics (POLSPAR)

Contents

Part I
Fault Detection and Isolation

Fault Diagnosis Based on Controller Modification

Henrik Niemann

Abstract Detection and isolation of parametric faults in closed-loop systems will be considered in this paper. A major problem is that a feedback controller will in general reduce the effects from variations in the systems including parametric faults on the controlled output from the system. Parametric faults can be detected and isolated using active methods, where an auxiliary input is applied. Using active methods for the diagnosis of parametric faults in closed-loop systems, the amplitude of the applied auxiliary input need to be increased to be able to detect and isolate the faults in a reasonable time. A negative effect of increasing the amplitude of the auxiliary input is that the disturbances in the external output will be increased and consequently reduce the closed-loop performance. This problem can be handled by using a modification of the feedback controller. Applying the YJBK-parameterization (after Youla, Jabr, Bongiorno and Kucera) for the controller, it is possible to modify the feedback controller with a minor effect on the closed-loop performance in the fault-free case and at the same time optimize the detection and isolation in a faulty case. Controller modification in connection with both fault detection and isolation will be discussed. Also passive fault diagnosis methods based on controller modification will be discussed.

Keywords Active fault diagnosis · Parametric faults · Feedback control · Controller parameterization · Controller modification

1 Introduction

The problem of detection and isolation of parametric faults has been considered in a large number of publications, see e.g. [3, 5–7, 17], where different methods have been described.

H. Niemann (✉)
Department of Electrical Engineering, Automation and Control
Technical University of Denmark, 2800 Kgs. Lyngby, Denmark
e-mail: hhn@elektro.dtu.dk

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_1

One of the applied methods is to transform the parametric faults into additive faults and then use standard fault methods for additive faults. Let's consider a system including an actuator fault described by a change of the actuator gain given as

$$y = G_0(s)(1 + \theta_a)u \tag{1}$$

where θ_a is a parametric fault describing the change in the actuator gain. $\theta_a = 0$ describes the fault-free case and $\theta_a = -1$ describe the system with a total loss of the actuator. Equation (1) can be rewritten as

$$y = G_0(s)(u + f_a(u)) \tag{2}$$

where the additive fault f_a is given by

$$f_a(u) = \theta_a u$$

It is then possible to detect the parametric fault θ_a using additive method. It is also clear from (2), it is only possible to detect the fault when the input u is non-zero.

The second issue is that the system is typically applied in closed-loop. The feedback controller will in many cases reduce the effects of parametric faults in the system, especially for small faults. This can be seen from the following simple calculations. Let's consider the output error signal in the open-loop case given as

$$\begin{aligned} e_o &= y - y_0 = G_0(s)(1 + \theta_a)u - G_0(s)u \\ &= G_0(s)\theta_a u \end{aligned} \tag{3}$$

The error signal e_o is proportional to the parametric fault θ_a and the nominal system.

Now, let the system be controlled by a proportional feedback controller with the gain k_P. The error signal for the closed-loop system is then given by:

$$e_{cl} = y - y_0 = \frac{G_0(s)(1 + \theta_a)k_P}{1 + G_0(s)(1 + \theta_a)k_P} r_{ref} - \frac{G_0(s)k_P}{1 + G_0(s)k_P} r_{ref}$$

or

$$e_{cl} = \frac{G_0(s)k_P}{1 + G_0(s)k_P} \frac{1}{1 + G_0(s)(1 + \theta_a)k_P} \theta_a r_{ref} \tag{4}$$

where r_{ref} is the reference input. Assume that the controller gives good reference tracking, i.e. $G_0(s)k_P$ has large gain. Then (4) is approximated with

$$e_{cl} \approx \frac{1}{G_0(s)k_P} \theta_a r_{ref} \tag{5}$$

for θ_a small.

Equation (5) shows that the feedback controller will in many cases reduce the effect from parametric faults significantly and then fault detection and isolation gets more difficult than in the open-loop case.

The next approach is to apply an active fault diagnosis method, see e.g. [1, 2, 4, 8, 10, 19, 22]. In active methods, auxiliary input signals η are injected into the open or closed-loop system to get faster and more reliable diagnosis. The auxiliary inputs are designed with respect to detection and isolation of specific parametric faults in the systems. Using the same closed-loop system from above and letting the auxiliary input signal be injected at the reference input, the closed-loop output error is then given by:

$$e_{cl} = \frac{G_0(s)k_P}{1 + G_0(s)k_P} \frac{1}{1 + G_0(s)(1 + \theta_a)k_P} \theta_a (r_{ref} + \eta) \tag{6}$$

The auxiliary input η is designed such that the effect from a parametric fault in the output error signal is optimized. Using a specific auxiliary input, the fault detection and isolation is done by an investigation of the signature from η in e_{cl}. The frequency content and the amplitude for η are design parameters.

As it can be seen from the above small example, fault diagnosis in closed-loop system can in many cases be difficult. Applying the active approach to the closed-loop system will give a more systematic method, because the diagnosis is based on dedicated auxiliary inputs. Both fault detection as well as fault isolation is based on a detailed investigation of the signatures from η in the closed-loop output error. Increasing the amplitude of η will reduce the diagnosis time. Further, different auxiliary inputs can be applied for detection and isolation. A drawback of active based diagnosis methods is that auxiliary input signals applied for diagnosis will also disturb the controlled outputs. This will give some limitation on the applied auxiliary inputs, so the disturbances on the controlled outputs are limited.

The main reason that fault diagnosis is difficult in closed-loop systems is the effect of the feedback controller as shown above. Using active fault diagnosis approach simplify the diagnosis to a certain point, but not without a negative effect from the applied auxiliary inputs on the controlled outputs. As described above, an auxiliary input is injected into the system when active methods are used. The fault diagnosis can then be based on the signature from the auxiliary input in an output from the system or in a residual output from a residual generator, i.e. active fault diagnosis can be formulated for both open-loop as well as for closed-loop systems. The setup for active fault diagnosis in closed-loop systems described in [10, 11, 18] is based on the YJBK controller architecture. The set-up includes a nominal feedback controller. By including the feedback controller in the set-up, the effect on the fault detection from the feedback controller will be included in the analysis and design of auxiliary inputs to the system. The general active fault diagnosis set-up is based on a coprime factorization of the system and controller in the general case, but can be simplified for stable systems. As described in [10, 11], the set-up is strongly connected with the YJBK parameterization of all stabilizing controllers and the dual YJBK parameterization of all systems stabilized by a given nominal feedback controller.

When it is possible to reduce the effects from parametric faults in the closed-loop system using feedback controllers, it will also be possible to increase the effect by using feedback controllers. The problem is now to modify the controller in a way such that it has a minor effect on the closed-loop performance in the fault-free case and at the same time optimize the fault detection and isolation in the faulty case. As it will be shown in this paper, the setup applied for active fault diagnosis based on the YJBK controller architecture can directly be applied for diagnosis based on controller modifications.

The concept of fault diagnosis based on controller modification will be described in the rest of this paper. Both fault detection as well as fault isolation will be considered. First, the general setup for active fault diagnosis based on the YJBK architecture is introduced. Based on this setup, the residual vector is analyzed with respect to a controller modification. Then a description of the fault detection problem follows, including a discussion of how to modify the controller with respect to optimizing the fault detection. At last, the fault isolation problem is considered. Both a passive based and an active based fault isolation methods are described.

2 The YJBK Architecture

The YJBK architecture is the basis for the fault diagnosis methods based on controller modification considered in this paper.

Let the dynamic system given by:

$$\Sigma_P \; : \; \begin{cases} z = G_{zw}w + G_{zd}d + G_{zu}u \\ e = G_{ew}w + G_{ed}d + G_{eu}u \\ y = G_{yw}w + G_{yd}d + G_{yu}u \end{cases} \tag{7}$$

where $d \in \mathcal{R}^r$ is an external disturbance input vector, $u \in \mathcal{R}^m$ the control input signal vector, $e \in \mathcal{R}^q$ is the external output signal vector to be controlled and $y \in \mathcal{R}^p$ is the measurement vector. Further, $w \in \mathcal{R}^k$ and $z \in \mathcal{R}^k$ are external input and output vectors. The connection between z and w is given as

$$w = \theta z$$

where θ is a diagonal matrix represents the parametric faults in the system. θ_i, $i = 1, \ldots, k$, in the diagonal of θ represent the k single parametric faults in the system. $\theta = 0$ represent the fault-free case, see e.g. [11].

Closing the loop from w to z in Σ_P by using θ, the resulting system can be realized [23] by the upper linear fractional transformation (LFT) in θ

$$\Sigma_{P,\theta} = \mathcal{F}_u(\Sigma_P, \theta)$$

where $\Sigma_{P,\theta}$ is given by

$$\Sigma_{P,\theta} \; : \; \begin{cases} e = G_{ed}(\theta)d + G_{eu}(\theta)u \\ y = G_{yd}(\theta)d + G_{yu}(\theta)u \end{cases} \tag{8}$$

The system is controlled by a stabilizing feedback controller given as

$$\Sigma_C \; : \; \{ u = Ky \tag{9}$$

2.1 The YJBK Parameterization

The coprime factorization of the nominal system G_{yu} from (7) and the stabilizing controller K from (9) are given by:

$$\begin{aligned} G_{yu} &= NM^{-1} = \tilde{M}^{-1}\tilde{N},\ N, M, \tilde{N}, \tilde{M} \in \mathcal{RH}_\infty \\ K &= UV^{-1} = \tilde{V}^{-1}\tilde{U}, \quad U, V, \tilde{U}, \tilde{V} \in \mathcal{RH}_\infty \end{aligned} \tag{10}$$

where the eight matrices in (10) must satisfy the double Bezout equation [21].

Based on the coprime factorization of the system and the controller, it is possible to give a parametrization of all controllers that stabilize the system in terms of a stable matrix transfer function Q, i.e. all stabilizing controllers are given [21] by

$$K(Q) = (\tilde{V} + Q\tilde{N})^{-1}(\tilde{U} + Q\tilde{M}),\ Q \in \mathcal{RH}_\infty \tag{11}$$

The above controller parametrization can also be realized as a lower Linear Fractional Transformation (LFT) in the parameter Q:

$$K(Q) = \mathcal{F}_l\left(\begin{pmatrix} UV^{-1} & \tilde{V}^{-1} \\ V^{-1} & -V^{-1}N \end{pmatrix}, Q\right) = \mathcal{F}_l(J_K, Q) \tag{12}$$

where J_K is given by:

$$J_K = \begin{pmatrix} UV^{-1} & \tilde{V}^{-1} \\ V^{-1} & -V^{-1}N \end{pmatrix} \tag{13}$$

Equivalently, it is possible to derive a parametrization in terms of a stable matrix transfer function S of all systems that are stabilized by one controller, i.e. the dual YJBK parametrization. The parameterization is given by [21]:

$$G_{yu}(S) = (\tilde{M} + S\tilde{U})^{-1}(\tilde{N} + S\tilde{V}),\ S \in \mathcal{RH}_\infty \tag{14}$$

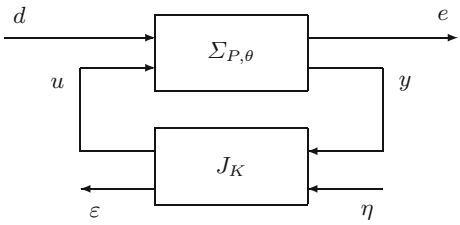


Fig. 1 Setup for AFD in the closed-loop system. The auxiliary input vector is η and the external output vector is ε, i.e. the residual vector

Further, it can be shown that S can be described using an upper LFT [21]:

$$S = \mathcal{F}_u(J_K, G_{yu}(S)) \tag{15}$$

The matrix transfer function S is a function of the system variations. Here variations in the system in terms of the parametric faults described by θ, i.e. $S = S(\theta)$ will be considered. Assuming that $\theta = 0$ is the nominal value of θ, then there exist the following simple relation [9]:

$$S(\theta) = 0, \text{ for } \theta = 0 \tag{16}$$

By testing if $S(\theta)$ is zero or non-zero, parametric faults can be detected. From (15) we have that S is given directly as the matrix transfer function between the lower input and output of J_K when the upper inputs and outputs are closed by the system as shown in Fig. 1.

2.2 Closed-Loop System

The dual YJBK parameter S is the matrix transfer function between the auxiliary input η and the external output ε. As a consequence of this, ε can be used as a residual vector in connection with active fault diagnosis. Moreover, it has been shown in [10, 11] that ε is a residual vector satisfying the decoupling conditions. Further, in the case where it is based on a full order observer based feedback controller, ε is the output estimation error vector or the innovation vector when a Kalman filter is applied. In [11], $S(\theta)$ is also called the *fault signature matrix*.

From the set-up that is shown in Fig. 1, the closed-loop system [11] is

$$\Sigma_{P,K} \; : \; \begin{cases} e = P_{ed}(\theta)d + P_{e\eta}(\theta)\eta \\ \varepsilon = P_{\varepsilon d}(\theta)d + \; S(\theta)\eta \end{cases} \tag{17}$$

where

$$P_{ed}(\theta) = G_{ed}(\theta) + G_{eu}(\theta)(\tilde{V} - \tilde{U}G_{yu}(\theta))^{-1}\tilde{U}G_{yd}(\theta)$$
$$P_{e\eta}(\theta) = G_{eu}(\theta)(\tilde{V} - \tilde{U}G_{yu}(\theta))^{-1}$$
$$P_{\varepsilon d}(\theta) = \tilde{M}G_{yd}(\theta) + (\tilde{M}G_{yu}(\theta) - \tilde{N})(\tilde{V} - \tilde{U}G_{yu}(\theta))^{-1}\tilde{U}G_{yd}(\theta)$$

3 Active Fault Diagnosis

This section deals with detection of parametric faults in systems. The results in this section are the basic for the isolation results given in the next section.

The second output ε from J_K given by (13) (also input to Q) is also a residual vector as shown in [10] for using in connection with (passive) fault diagnosis. The residual vector is given by:

$$\varepsilon = \tilde{M}y - \tilde{N}u \tag{18}$$

A simple way to implement the above residual generator is to use a full order observer or a Kalman filter, see [10]. The feedback controller K included the residual vector ε and the input vector η is shown in Fig. 1.

Based on the set-up shown in Fig. 1, the matrix transfer function from the external input d to the residual vector ε can now be calculated. This gives [11] the following residual vector:

$$\varepsilon = P_{\varepsilon d}(S)d + S(\theta)\eta \tag{19}$$

where

$$P_{\varepsilon d}(\theta) = \tilde{M}G_{yd}(\theta) + (\tilde{M}G_{yu}(\theta) - \tilde{N})(I - KG_{yu}(\theta))^{-1}KG_{yd}(\theta)$$
$$\begin{aligned} S(\theta) &= (\tilde{M}G_{yu}(\theta) - \tilde{N})(I - KG_{yu}(\theta))^{-1}\tilde{V}^{-1} \\ &= \tilde{M}G_{yw}\theta(I - (G_{zw} + G_{zu}U\tilde{M}G_{yw})\theta)^{-1}G_{zu}M \end{aligned}$$

Using

$$(I - KG_{yu}(\theta))^{-1} = (M + US(\theta))\tilde{V}$$

in (19) gives:

$$\varepsilon = P_{\varepsilon d}(S)d + S(\theta)\eta = (\tilde{M} + S(\theta)\tilde{U})G_{yd}(\theta)d + S(\theta)\eta \tag{20}$$

In the case when there is no parametric faults in the system ($S(0) = 0$), the residual vector ε in (20) is given by:

$$\varepsilon = \tilde{M}G_{yd}(0)d \tag{21}$$

3.1 Controller Based Detection

Now, let us consider the closed loop system $\Sigma_{P,K}$ given by (17). Further, applying the feedback controller

$$\eta = Q\varepsilon \tag{22}$$

gives the closed loop setup shown in Fig. 2, where η_Q and ε_Q are the auxiliary input and residual vector for the closed-loop system.

The system $\bar{\Sigma}_{P,K}$ in Fig. 2 is given by (modification of (17)):

$$\bar{\Sigma}_{P,K} \; : \; \begin{cases} e &= P_{ed}(\theta)d + P_{e\eta}(\theta)\eta_Q + P_{e\eta}(\theta)\eta \\ \varepsilon_Q &= P_{\varepsilon d}(\theta)d + S\eta_Q + S\eta \\ \varepsilon &= P_{\varepsilon d}(\theta)d + S\eta_Q + S\eta \end{cases} \tag{23}$$

Closing the loop around $\bar{\Sigma}_{P,K}$ with Q gives

$$\bar{\Sigma}_{P,Q} \; : \; \begin{cases} e &= \bar{P}_{ed}(Q,S)d + \bar{P}_{e\eta}(Q,S)\eta_Q \\ \varepsilon_Q &= \bar{P}_{\varepsilon d}(Q,S)d + S_Q(\theta)\eta_Q \end{cases} \tag{24}$$

where

$$\begin{aligned} \bar{P}_{ed}(Q,S) &= P_{ed}(\theta) + P_{e\eta}(\theta)Q(I - S(\theta)Q)^{-1}P_{\varepsilon d}(\theta) \\ \bar{P}_{e\eta}(Q,S) &= G_{eu}(\theta)(M + US(\theta))(I - QS(\theta))^{-1} \\ \bar{P}_{\varepsilon d}(Q,S) &= (I - S(\theta)Q)^{-1}(\tilde{M} + S(\theta)\tilde{U})G_{yd}(\theta) \\ S_Q(\theta) &= (I - S(\theta)Q)^{-1}S(\theta) \end{aligned}$$

and $P_{ed}(S)$, $P_{e\eta}(S)$ and $P_{\varepsilon d}(S)$ are given by (17).

In the fault-free case, the closed-loop system in (24) is given as

$$\bar{\Sigma}_{P,Q} \; : \; \begin{cases} e &= (G_{ed}(0) + G_{eu}(0)M(\tilde{U} + Q\tilde{M})G_{yd}(0))d + G_{eu}(0)M\eta_Q \\ \varepsilon_Q &= \tilde{M}G_{yd}(0)d \end{cases} \tag{25}$$

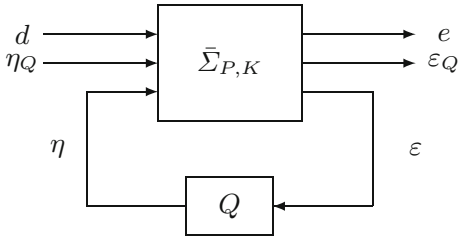


Fig. 2 The closed loop feedback system Q as the feedback controller

The closed-loop residual output vector ε_Q from (24) is now

$$\begin{aligned}\varepsilon_Q &= (I - S(\theta)Q)^{-1}(\tilde{M} + S(\theta)\tilde{U})G_{yd}(\theta)d + (I - S(\theta)Q)^{-1}\eta_Q \\ &= S_Q(\theta)(\tilde{M} + S(\theta)\tilde{U})G_{yd}(\theta)d + S_Q(\theta)\eta_Q\end{aligned} \tag{26}$$

The closed-loop residual vector ε_Q depend on the external input d, the auxiliary η_Q, the parametric fault θ, the nominal system and feedback controller and the YJBK matrix transfer function Q.

The closed-loop residual vector ε_Q given by (26) can now be analyzed with respect to disturbance input d, the auxiliary input η_Q and parametric faults θ. Four different cases are considered below.

A: *Disturbance-Free Case*. In the disturbance-free case and with no auxiliary input, both the residual vector ε and the closed-loop residual vector ε_Q given by (19) and (26), respectively, are zero and independent of possible faults in the system, i.e.

$$\varepsilon_Q = \varepsilon = 0 \text{ for } d = 0,\ \eta_Q = 0,\ \forall\theta,\ \forall Q \tag{27}$$

B: *Nominal Case*. We have that $\theta = 0$ in the nominal or fault-free case results in that the fault signature matrix is also zero, i.e. $S(0) = S_Q(0) = 0$. This gives the following relation:

$$\varepsilon_Q = \varepsilon \text{ for } d \neq 0,\ \eta_Q \neq 0,\ \theta = 0,\ \forall Q \tag{28}$$

where ε is given by (21).

C: *Faulty Case*. In the faulty case where $\theta \neq 0$, the associated fault signature matrix will be non-zero, i.e. $S(\theta) \neq 0$. Modifying the controller using the YJBK matrix transfer function Q will change the residual vector in general. This gives the following relation between the two residual vectors ε and ε_Q:

$$\varepsilon_Q \neq \varepsilon \text{ for } d \neq 0,\ \eta_Q = 0, \theta \neq 0,\ \forall Q,\ S(\theta)Q \neq 0 \tag{29}$$

D: *Faulty Case*. In the faulty case where $\theta \neq 0$, the associated fault signature matrix will be non-zero, i.e. $S(\theta) \neq 0$. Modifying the controller using the YJBK matrix transfer function Q will change the residual vector in general. This gives the following relation between the two residual vectors ε and ε_Q:

$$\varepsilon_Q \neq \varepsilon \text{ for } d \neq 0,\ \eta_Q \neq 0,\ \theta \neq 0,\ \forall Q,\ S(\theta)Q \neq 0 \tag{30}$$

From case A we see that if there is no external input to the system (d and η_Q), the residual output will be zero. As a consequence of this, it will be not possible to detect parametric faults in the system at all. Therefore, it is required that d or η_Q is non-zero to be able to detect parametric faults in the system using the passive method.

An important observation from case B is that the closed-loop residual vector ε_Q is independent of a change in the controller via the YJBK matrix transfer function Q.

It will not change the residual vector when there are no parametric faults in the system. Further, from case C and D, we have that closed-loop residual vector ε_Q depend on Q in the faulty case.

It is important to point out that with case B and case C, it is possible to discriminate between the effect from the external input d and parametric faults, i.e. a detection of parametric faults in the system.

4 Fault Detection

The fault detection case of [15] is based directly on the four cases described above. A direct result of Case B, Case C and Case D, the residual vector ε_Q will be changed by modifying the feedback controller by using a Q when θ is non-zero and unchanged in the fault-free case. It is possible to detect parametric faults using both the passive concept ($\eta_Q = 0$) or the active concept ($\eta_Q \neq 0$).

The question is then how to design Q such that the fault detection is optimized. On the other hand, including Q in the controller will also change the closed-loop performance of the system, both in the nominal case as well as in the faulty case as it can be seen from (24) and (25). An extreme design has been considered in [20], where Q is designed such that the closed-loop system is stable in the nominal case and unstable in the faulty case. Q designed in that way will not in general be acceptable for on-line use both due to the stability aspect, as well as due to major performance degradation in the nominal case.

Both in passive and active methods, it is required that the performance degradation by including Q is minimal in the fault-free case. The consequence of this is that Q need to be designed as a decoupling controller or designed such that the dynamic of $G_{eu}(0)MQ\tilde{M}G_{yd}(0)$ (see (25)) is small or outside the bandwidth of the nominal closed-loop system.

In the passive approach, it might be difficult to obtain a reasonable change in the residual vector ε_Q when Q with dynamic above the bandwidth of the closed-loop system is applied. The detection time might then be unacceptably large. One way to compensate this is to only include Q in certain time periods and then accept a larger performance degradation in these periods. Further, it will be only possible then to detect faults in these periods.

Using active method for the detection, we will have one degree of freedom more compared with the passive approach. Here both the YJBK matrix transfer function Q and the auxiliary input vector η_Q are free for design. Note also that the matrix transfer function from auxiliary input η_Q to the external output e in (24) is independent of Q in the nominal case. Q can be included now in the feedback controller for optimizing the fault detection speed without increasing the effect from the auxiliary input vector on the external output.

Using the YJBK matrix transfer function Q to detune the feedback controller in a way such that it will increase the sensitivity from η_Q to the residual output ε_Q. Using a periodic auxiliary input η_Q as applied in e.g. [10, 11, 18], this detuning only needs

to be done in the frequency range around the frequency for the auxiliary harmonic input. This can be done using a band-pass filter given by (for the SISO case but can be generalized to the MIMO case)

$$Q(s) = K_q \frac{\zeta s}{s^2 + 2\zeta\omega_q s + \omega_q^2} \tag{31}$$

where ω_q is the center frequency and ζ is the reciprocal quality factor.

Including band-pass filter given by (31) in the fault-free closed-loop system in (25) will only have an effect around the center frequency ω_q in the band-pass filter. ζ can then be used for tuning the filter with respect to minimize the effect on the fault-free closed-loop system.

The selection of the center frequency ω_q is based on analysis of the fault signature matrix $S(\theta)$. The frequency is selected in the frequency range where the fault signature matrix $S(\theta)$ has its maximal amplitude. It is important that the frequency ω in the auxiliary input and the center frequency ω_q should be the same. Thus it is possible to reduce the amplitude of the auxiliary input without reducing the detection time or reducing the detection time.

$S(\theta)$ might not have maximal amplitude for all faults θ_i in the same frequency range. Therefore, we will in such cases need to use a number of YJBK matrix transfer functions designed with respect to one or more faults. In the fault detection, we will need to change between a number of YJBK matrix transfer functions with associated auxiliary inputs.

For MIMO systems, detailed analysis is given in [13] using active fault detection without controller modification. These results can be directly applied in connection with the design of Q for MIMO systems. Here, it is important to select the optimal auxiliary input directions and residual output directions.

A small SISO example of using band-pass filters for Q has been shown in [15]. The nominal system considered is given as

$$G(s) = \frac{s^2 + 1}{(s+1)(s+2)(s+3)}$$

and the faulty system is

$$G_f(s) = \frac{s^2 + 1}{(s+1)(s+2)(s+4)}$$

(i.e. under the influence of this error, the pole moves from the point -3 to the point -4). Further, a standard LQG controller $K(s)$ is applied.

Including a band-pass filter $Q(s)$ given by (31) that is designed with a center frequency around the frequencies where $S(\theta)$ is maximal in the presence of the parametric fault in the system.

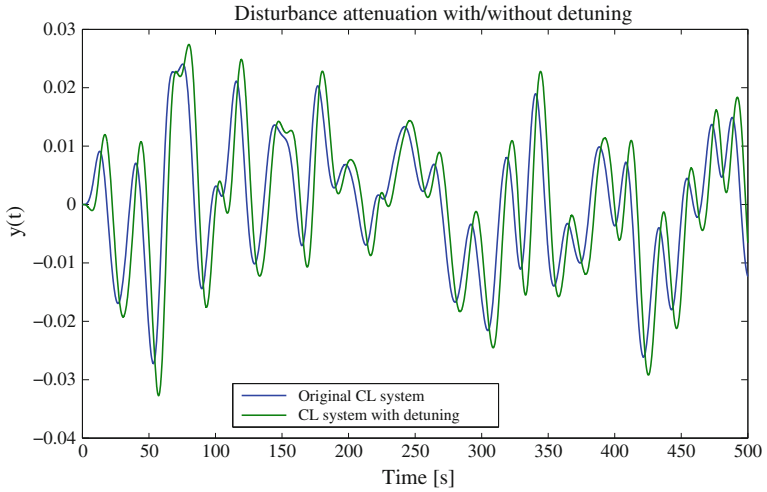


Fig. 3 Simulation showing disturbance attenuation at output both with and without the detuning parameter Q. It can be seen that the detuning causes a slight decrease in performance. This performance loss can be made arbitrarily small by controlling the quality factor of the bandpass filter of Q

The result of the detuning with respect to performance and fault detection is shown in Figs. 3 and 4. From Fig. 3, it can be seen that the ability to suppress high frequency disturbances have been slightly decreased.

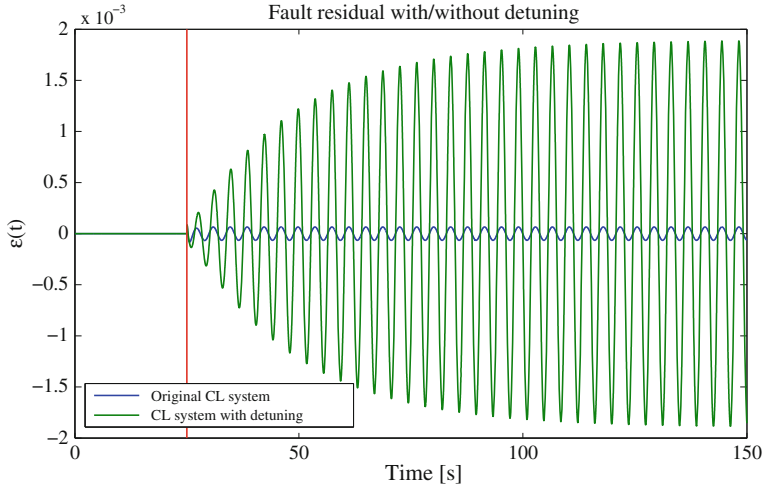


Fig. 4 Simulation showing the response of the fault residual to a parametric fault occurring at $t = 25$ s. It can be seen that the detuning sensitizes the fault residual with approximately a factor of 30 compared to the situation without Q

Figure 4 shows that the introduction of Q increased the sensitivity for the active fault diagnosis signal significantly.

Including Q in the controller, the sensitivity was improved with a factor of approximately 30. This means that the probability of detection given a certain signal-to-noise ratio will improve significantly. Alternatively, the amplitude of the active fault diagnosis signal can be reduced correspondingly, so as to avoid adverse spill-over effects on the output.

5 Fault Isolation

In the fault isolation case, it is possible to apply both passive and active methods. The two different methods will shortly be described in the following.

To be able to isolate faults, the faults in the system need to give unique signatures in the residual vectors. This is significantly easier using the active approach than using the passive approach. In the passive approach, the input to the system is not well defined with the result that the signatures in the residual vector are not well defined. However, using a decoupling approach for the design of Q, it is still possible to apply the passive approach for fault isolation as described in the following. In the active approach, where the auxiliary input is known, the signatures in the residual vector will also be known with respect to the possible faults in the system.

5.1 Passive Based Isolation

The passive based method shortly described here is described in details in [16].

This passive method is based on decoupling. In the passive detection case, Q need to be designed such that it will change the residual vector when it is included in the feedback controller in the faulty case. Let us assume now that a fault has been detected in the system and it needs to be isolated. Isolation based on a unique signature or a unique signature change in the residual vector is achieved by designing a dedicated Q_i for each specific parametric fault θ_i. Q_i is designed such that it will change the residual vector when it is included in the feedback controller for all faults apart from when fault θ_i had occurred in the system. Together with the detection of faults in the system, we get a unique indication in the residual vector when θ_i has occurred in the system, the residual vector is unchanged when Q_i is applied.

The design of Q_i needs to satisfy the following conditions:

$$\begin{aligned} &S(\theta_i)Q_i = 0,\ \theta_i \neq 0 \\ &S(\theta_j)Q_i \neq 0,\ \theta_j \neq 0,\ i \neq j \end{aligned} \tag{32}$$

If Q_i satisfy (32), the closed-loop residual vector ε_Q will satisfy:

$$\begin{aligned} &\varepsilon_Q(\theta_i) = \varepsilon(\theta_i) \text{ for } Q = Q_i,\ \theta_i \neq 0 \\ &\varepsilon_Q(\theta_j) \neq \varepsilon(\theta_j) \text{ for } Q = Q_i,\ \theta_j \neq 0,\ i \neq j \end{aligned} \tag{33}$$

i.e. the residual vector is unchanged if fault θ_i has occurred in the system or else it will be changed by including Q_i in the feedback loop.

For satisfying the first condition in (32), Q_i must be selected such that it is in the nullspace of $S(\theta_i)$, i.e. $Q_i \in \mathcal{N}(S(\theta_i))$. The second condition is satisfied if Q_i is not in the nullspace of $S(\theta_j)$. This requires that $\mathcal{N}(S(\theta_j)) \neq \mathcal{N}(S(\theta_i))$, $j \neq i$. These design conditions can only be satisfied for MIMO systems. Further, the normal rank of $S(\theta_i)$ is one, so in general, there exists a non-zero Q_i satisfying the above conditions. The condition can be formulated [16] in the following way:

$$normal\ rank \begin{pmatrix} G_{zu,i:} \\ G_{zu,j:} \end{pmatrix} = 2,\ i \neq j \tag{34}$$

where $G_{zu,i:}$ is the $i'th$ row of G_{zu}. If this condition is satisfied, then there exists a non-zero Q_i that will satisfy the conditions (32) and isolation of fault θ_i from θ_j is possible.

The design of Q_i is not so critical with respect to performance degradation in connection with fault isolation as in connection with fault detection. If faults occurred in the system, it will in general be more important to get a fast isolation of the faults than having focus performance degradation.

More details for design of Q_i can be found in [16].

5.2 Active Based Isolation

In the active approach, we have again the freedom to design both Q and the auxiliary input η_Q. In the active based isolation case, two different approaches can be applied. It is possible to apply the same decoupling approach as described for the passive approach or an approach where the isolation is obtained by maximizing the effect from a specific fault.

Using the decoupling concept from the passive approach, the design of Q_i needs to satisfy the same conditions as given in (32). The major difference between the passive approach and the active approach based on decoupling, is the detection of a change in the residual vector. In the active approach, we just need to detect if the signature from η_Q is changed or unchanged when Q_i is included in the feedback controller. In the passive case, it will in general be more complicated to detect a change.

The other approach for active fault isolation is based on maximizing the effect in the residual vector when a certain fault had occurred in the system. The active fault isolation method described in [14] is based on design of the auxiliary input direction

and the residual output direction for optimizing the effects in ϵ from a specified fault in the system. By including a controller modification in this approach, it will be possible to optimize the isolation step further by getting more significant signatures in the residual vector with respect to the different faults in the system. This will give a more reliable isolation than applying the same auxiliary inputs.

For SISO systems, the isolation is done by an investigation of the amplitude and phase shift through $S(\theta)$ with a specified auxiliary input, see [18]. Again, using Q_i, it will be possible to maximize the separation between amplitude and phase in the residual signal with faster and better isolation as the result.

More details about controller based fault diagnosis based on the active approach can be found in [12].

6 Discussion

The concept of using controller modification in connection with fault diagnosis has been considered in this paper. It is shown that using the YJBK controller architecture, it is possible to detune the feedback controller with respect to optimizing detection and isolation of parametric faults without significant performance reduction. The concept is in general connected with active fault diagnosis. However, the fault isolation problem can also be formulated in the passive version, where the isolation is done by decoupling in the detuned controller.

As compared with standard active based fault diagnosis methods, the controller-based methods gives an extra design parameter for the design of the residual generator. The methods combine the advantages from both active fault diagnosis and the pure controller based fault detection method as described in [20] and at the same time minimizing disadvantages with these methods. In the active fault diagnosis methods described in [10, 13, 14], the auxiliary input is designed with respect to optimizing the fault diagnosis. This might cause an unnecessary performance reduction. The controller-based fault detection method from [20] can make the closed-loop system unstable for a fault that results in a significant loss in performance. A combination of these two concepts means that the controller can be detuned in a certain frequency range such that it will not destroy the closed-loop performance. It is then possible to apply an auxiliary input signal in this frequency range. Further, the detuning of the controller will also allow reducing the amplitude of the auxiliary input while still obtaining fault detection and isolation in reasonable time.

Acknowledgments The author would like to thank Dr. Niels Kjølstad Poulsen, Technical University of Denmark, and Professor Jakob Stoustrup, Pacific Northwest National Laboratory, USA, for the joint work on the area of controller based fault diagnosis.

References

1. Ashari, A., Nikoukhah, R., Campbell, S.: Auxiliary signal design for robust active fault detection of linear discrete-time systems. Automatica **47**(9), 1887–1895 (2011)
2. Ashari, A., Nikoukhah, R., Campbell, S.: Effects of feedback on active fault detection. Automatica **48**(5), 866–872 (2012)
3. Blanke, M., Kinnart, M., Lunze, J., Staroswiecki, M.: Diagnosis and Fault-tolerant Control. Springer, Berlin (2006)
4. Campbell, S., Nikoukhah, R.: Auxiliary Signal Design for Failure Detection. Princeton University Press, Princeton (2004)
5. Doraiswami, R., Diduch, C., Tang, J.: A new diagnostic model for identifying parametric faults. IEEE Trans. Control Syst. Technol. **18**(3), 533–544 (2010)
6. Frank, P.: Analytical and qualitative model-based fault diagnosis—a survey and some new results. Eur. J. Control **2**, 6–28 (1996)
7. Gertler, J.: Fault detection and isolation using parity relations. Control Eng. Pract. **5**(5), 653–661 (1997)
8. Keresteciogl u, F.: Change Detection and Input Design in Dynamic Systems. Research Studies Press, Baldock, Hertfordshire (1993)
9. Niemann, H.: Dual youla parameterization. IEE Proc. Control Theory Appl. **150**(5), 493–497 (2003)
10. Niemann, H.: A setup for active fault diagnosis. IEEE Trans. Autom. Control **51**(9), 1572–1578 (2006)
11. Niemann, H.: A model-based approach for fault-tolerant control. Int. J. Appl. Math. Comput. Sci. **22**(1), 67–86 (2012)
12. Niemann, H., Poulsen, N., Stoustrup, J.: Controller Modification Applied for Active Fault Diagnosis (2015), under preparation
13. Niemann, H., Poulsen, N.: Active fault detection in MIMO systems. In: Proceedings of the American Control Conference, Portland, Oregon, pp. 1975–1980 (June 2014)
14. Niemann, H., Poulsen, N.: Active Fault Isolation in MIMO Systems. In: Proceedings of the 19th Ifac World Congress. pp. 8012–18017. Cape Town, South Africa (August 2014)
15. Niemann, H., Stoustrup, J., Poulsen, N.: Controller modification applied for active fault detection. In: Proceedings of the American Control Conference, Portlan, Oregon, June 2014, pp. 1963–1968 (June 2014)
16. Niemann, H., Stoustrup, J., Poulsen, N.: Fault Isolation in MIMO System Based on Decoupling (2015), submitted for publication
17. Palma, L., Coito, F., Neves-Silva, R.: Diagnosis of parametric faults based on identification and statistical methods. In: Proceedings of the 44th IEEE Conference on Decision and Control and 2005 European Control Conference, Seville, Spain, pp. 3838–3843 (2005)
18. Poulsen, N., Niemann, H.: Active fault diagnosis based on stochastic tests. Int. J. Appl. Math. Comput. Sci. **18**(4), 487–496 (2008)
19. Simandl, M., Puncochar, I.: Active fault detection and control: unified formulation and optimal design. Automatica **45**(9), 2052–2059 (2009)
20. Stoustrup, J., Niemann, H.: Active fault diagnosis by controller modification. Int. J. Syst. Sci. **41**(4), 925–936 (2010)
21. Tay, T., Mareels, I., Moore, J.: High Performance Control. Birkhäuser, New York (1997)
22. Zhang, X.: Auxiliary Signal Design in Fault Detection and Diagnosis. Springer, Heidelberg (1989)
23. Zhou, K., Doyle, J., Glover, K.: Robust and Optimal Control. Prentice Hall, Boston (1995)

One Approach to Design the Fuzzy Fault Detection Filters for Takagi-Sugeno Models

Dušan Krokavec and Anna Filasová

Abstract The paper relates a principle for designing the fuzzy fault detection filters devoted to a class of continuous-time nonlinear systems represented by Takagi-Sugeno models. The extension of the fuzzy reference model principle and the incremental quadratic constraints are proposed to obtain an approximation of H_∞/H_- criterion in the residual weight matrix parameter design for TS fuzzy fault detection filters. The design conditions are outlined in terms of linear matrix inequalities to posses a stable design framework.

Keywords Fault detection · Quadratic performance · Lyapunov function · Fuzzy models · Matrix formulation

1 Introduction

The fault detection filters (FDF) are mostly used to generate fault residual signals in active fault tolerant control systems. Because it is generally not possible to decouple fault effects from the perturbation influence in residuals [5, 7], the H_∞/H_- approach is used to tackle this conflict [10, 11]. Since faults are usually detected by setting a threshold on the residual signals, determination of the actual threshold is formulated as an adaptive threshold task [8]. Other approaches reduce FDF design to H_∞ problem to discriminate fault and disturbance effects in FDF signals by using the reference residual models (RRM) [3, 4].

By providing the possibility of weighting linear state-space representations of the class of nonlinear systems, the Takagi-Sugeno (TS) fuzzy approach [17], which

D. Krokavec (✉) · A. Filasová
Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice,
Letná 9, 042 00 Kosice, Slovakia
e-mail: dusan.krokavec@tuke.sk
url: http://web.tuke.sk/kkui

A. Filasová
e-mail: anna.filasova@tuke.sk

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_2

avails mainly sector system dynamics approximation, have attracted noticeable penetration in fault detection [12, 15, 16] and estimation [9], usually utilizing the linear matrix inequality (LMI) design condition formulation. Building upon the theory of the systems whose some nonlinear time-varying terms satisfy the incremental quadratic constraints (IQC) [1, 2], in the paper the TS fuzzy models with local nonlinear terms are folded to design TS FDFs. In conjunction with RRM, it is demonstrated that IQC, parameterized by a multiplier matrix, can be reflected in LMI design conditions. The principle consists in generating the residual signals vector and estimating the subset of unmeasurable premise variables.

Throughout the paper $\boldsymbol{x}^T$, $\boldsymbol{X}^T$ denotes the transpose of the vector $\boldsymbol{x}$ and matrix $\boldsymbol{X}$, respectively, $\boldsymbol{X} = \boldsymbol{X}^T > 0$ means that $\boldsymbol{X}$ is a symmetric positive definite matrix, the symbol $\boldsymbol{I}_n$ indicates the nth order identity matrix, $\mathbb{R}$ denotes the set of real numbers, $\mathbb{R}^{n\times r}$ refers to the set of all $n \times r$ real matrices and $L_2\langle 0, +\infty)$ entails the space of square-integrable vector functions over $\langle 0, +\infty)$.

2 Takagi-Sugeno Fuzzy Fault Detection Filter

The systems under consideration belong to the class of MIMO nonlinear dynamic continuous-time systems, described by using TS approach as follows

$$\dot{\boldsymbol{q}}(t) = \sum_{i=1}^{s} \mathrm{h}_i(\boldsymbol{\theta}(t)) \left(\boldsymbol{A}_i \boldsymbol{q}(t) + \boldsymbol{B}_i \boldsymbol{u}(t) + \boldsymbol{E}_i \mathbf{p}(t) + \mathbf{B}_{fi} \boldsymbol{f}(t) + \mathbf{B}_{di} \boldsymbol{d}(t) \right) \tag{1}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{q}(t) \tag{2}$$

where $\boldsymbol{q}(t) \in \mathbb{R}^n$, $\boldsymbol{u}(t) \in \mathbb{R}^r$, $\boldsymbol{y}(t) \in \mathbb{R}^m$ are vectors of the state, input, and output variables, respectively, $\boldsymbol{C} \in \mathbb{R}^{m\times n}, \boldsymbol{A}_i \in \mathbb{R}^{n\times n}, \boldsymbol{B}_i \in \mathbb{R}^{n\times r}, \boldsymbol{E}_i \in \mathbb{R}^{n\times r_p}, \boldsymbol{B}_{fi} \in \mathbb{R}^{n\times r_f}$, $\boldsymbol{B}_{di} \in \mathbb{R}^{n\times r_d}$, $i = 1, 2, \ldots s$, are constant matrices and $\boldsymbol{d}(t) \in \mathbb{R}^{r_d}$ is the disturbance input that belongs to $L_2\langle 0, +\infty)$. Here s, v are the numbers of sub-models and premise variables, the vector of premise variables is

$$\boldsymbol{\theta}(t) = \left[\theta_1(t)\ \theta_2(t)\ \cdots\ \theta_v(t) \right] \tag{3}$$

and $\mathrm{h}_i(\theta(t))$ is the i-th membership function satisfying the following properties

$$0 \le \mathrm{h}_i(\boldsymbol{\theta}(t)) \le 1, \quad \sum_{i=1}^{s} \mathrm{h}_i(\boldsymbol{\theta}(t)) = 1 \text{ for all } i \in \langle 1, \ldots, s\rangle \tag{4}$$

The nonlinear function $\mathbf{p}(t) \in \mathbb{R}^{r_p}$ is a bounded function of $\boldsymbol{q}(t)$, given as [2]

$$\mathbf{p}(t) = \varphi\left(\boldsymbol{U}\boldsymbol{q}(t) + \boldsymbol{W}\mathbf{p}(t)\right) \tag{5}$$

where $\boldsymbol{U} \in I\!R^{m_p \times n}$, $\boldsymbol{W} \in I\!R^{m_p \times r_p}$ are constant matrices. Note, if $\mathbf{p}(t)$ does not depends on the derivative of a system state variable then $\boldsymbol{W}$ is zero matrix.

It is considered that a fault $\boldsymbol{f}(t)$ may occur at an uncertain time, the size of the fault is unknown but bounded and all pairs $(\boldsymbol{A}_i, \boldsymbol{C})$, $i = 1, 2, \ldots s$, are observable (more details can be found, e.g., in [12, 13]).

Considering (1) and (2) TS FDF is defined as

$$\dot{\boldsymbol{q}}_e(t) = \sum_{i=1}^{s} \mathrm{h}_i(\boldsymbol{\theta}(t)) \left(\boldsymbol{A}_i \boldsymbol{q}_e(t) + \boldsymbol{B}_i \boldsymbol{u}(t) + \boldsymbol{E}_i \mathbf{p}_e(t) + \boldsymbol{J}_i \left(\boldsymbol{y}(t) - \boldsymbol{y}_e(t) \right) \right) \tag{6}$$

$$\mathbf{p}_e(t) = \varphi \left(\boldsymbol{U} \boldsymbol{q}_e(t) + \boldsymbol{W} \mathbf{p}_e(t) + \boldsymbol{J} (\boldsymbol{y}(t) - \boldsymbol{y}_e(t)) \right) \tag{7}$$

$$\boldsymbol{r}(t) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) \boldsymbol{V}_i \left(\boldsymbol{y}(t) - \boldsymbol{y}_e(t) \right), \quad \boldsymbol{y}_e(t) = \boldsymbol{C} \boldsymbol{q}_e(t) \tag{8}$$

where $\boldsymbol{q}_e(t) \in I\!R^n$ is estimate of $\boldsymbol{q}(t)$, $\boldsymbol{y}_e(t) \in I\!R^m$ is the observed output vector, $\mathbf{r}(t) \in I\!R^{m_r}$ is the residual signal, $\mathbf{p}_e(t) \in I\!R^{r_p}$ is estimate of $\mathbf{p}(t)$, and $\boldsymbol{J} \in I\!R^{m_p \times m}$, $\boldsymbol{J}_i \in I\!R^{n \times m}$, $\boldsymbol{V}_i \in I\!R^{m \times m}$, $i = 1, 2, \ldots, s$, is the set of gains.

Defining the notations

$$\boldsymbol{e}(t) = \boldsymbol{q}(t) - \boldsymbol{q}_e(t), \quad \delta \mathbf{p}(t) = \mathbf{p}(t) - \mathbf{p}_e(t), \quad \boldsymbol{A}_{ei} = \boldsymbol{A}_i - \boldsymbol{J}_i \boldsymbol{C} \tag{9}$$

the deviation form of TS FDF is

$$\dot{\boldsymbol{e}}(t) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) \left(\boldsymbol{A}_{ei} \boldsymbol{e}(t) + \boldsymbol{B}_{fi} \boldsymbol{f}(t) + \boldsymbol{B}_{di} \boldsymbol{d}(t) + \boldsymbol{E}_i \delta \mathbf{p}(t) \right) \tag{10}$$

$$\mathbf{p}_e(t) = \varphi \left(\boldsymbol{U} \boldsymbol{q}_e(t) + \boldsymbol{W} \mathbf{p}_e(t) + \boldsymbol{J} \boldsymbol{C} \boldsymbol{e}(t) \right) \tag{11}$$

$$\boldsymbol{r}(t) = \sum_{i=1}^{s} h_i(\theta(t)) \boldsymbol{V}_i \boldsymbol{C} \boldsymbol{e}(t) \tag{12}$$

In the following it is assumed that the premise variables associated with $\mathbf{p}(t)$ are unmeasurable while all others premise variables are measurable.

To explain basic relationships, the following lemma is presented.

Lemma 1 *If a matrix $\boldsymbol{M} \in \mathcal{M}$, where $\mathcal{M}$ is the set of real incremental multiplier matrices of dimension $(m_p + r_p) \times (m_p + r_p)$, then for the given matrices $\boldsymbol{U} \in I\!R^{m_p \times n}$, $\boldsymbol{W} \in I\!R^{m_p \times r_p}$, $\boldsymbol{J} \in I\!R^{m_p \times m}$ and $\boldsymbol{C} \in I\!R^{m \times n}$ IQC is*

$$\left[\boldsymbol{e}^T(t) \; \delta \mathbf{p}^T(t) \right] \boldsymbol{N} \begin{bmatrix} \boldsymbol{e}(t) \\ \delta \mathbf{p}(t) \end{bmatrix} \geq 0 \tag{13}$$

$$\boldsymbol{N} = \begin{bmatrix} (\boldsymbol{U} - \boldsymbol{JC})^T & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \boldsymbol{Q}^T \boldsymbol{M} \boldsymbol{Q} \begin{bmatrix} \boldsymbol{U} - \boldsymbol{JC} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_{r_p} \end{bmatrix}, \quad \boldsymbol{Q} = \begin{bmatrix} \boldsymbol{I}_{m_p} & \boldsymbol{W} \\ \boldsymbol{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \tag{14}$$

and $\boldsymbol{I}_{m_p} \in \mathbb{R}^{m_p \times m_p}$, $\boldsymbol{I}_{r_p} \in \mathbb{R}^{r_p \times r_p}$ *are identity matrices of given dimensions.*

Proof (compare [1, 14]) Writing (11) as follows

$$\begin{aligned} \mathbf{p}_e(t) &= \varphi\left(\boldsymbol{U}(\boldsymbol{q}(t) - \boldsymbol{e}(t)) + \boldsymbol{W}\mathbf{p}_e(t) + \boldsymbol{J}(\boldsymbol{Cq}(t) - \boldsymbol{C}(\boldsymbol{q}(t) - \boldsymbol{e}(t))\right) \\ &= \varphi\left(\boldsymbol{Uq}(t) + \boldsymbol{W}\mathbf{p}_e(t) - (\boldsymbol{U} - \boldsymbol{JC})\boldsymbol{e}(t)\right) \end{aligned} \tag{15}$$

and introducing the variables

$$z_1(t) = \boldsymbol{Uq}(t) + \boldsymbol{W}\mathbf{p}(t)\,, \quad z_2(t) = \boldsymbol{Uq}(t) + \boldsymbol{W}\mathbf{p}_e(t) - (\boldsymbol{U} - \boldsymbol{JC})\boldsymbol{e}(t) \tag{16}$$

$$\delta z(t) = z_1(t) - z_2(t) = (\boldsymbol{U} - \boldsymbol{JC})\boldsymbol{e}(t) + \boldsymbol{W}\delta\mathbf{p}(t) \tag{17}$$

then (15) and (16) implies

$$\delta\mathbf{p}(t) = \mathbf{p}(t) - \mathbf{p}_e(t) = \varphi(z_1(t)) - \varphi(z_2(t)) = \delta\varphi(t) \tag{18}$$

Writing (17) and (18) compactly, it yields for a symmetric $\boldsymbol{M} \in \mathcal{M}$

$$\begin{bmatrix} \delta z(t) \\ \delta\varphi(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{U} - \boldsymbol{JC} & \boldsymbol{W} \\ \boldsymbol{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \begin{bmatrix} \boldsymbol{e}(t) \\ \delta\mathbf{p}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_{m_p} & \boldsymbol{W} \\ \boldsymbol{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \begin{bmatrix} \boldsymbol{U} - \boldsymbol{JC} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \begin{bmatrix} \boldsymbol{e}(t) \\ \delta\mathbf{p}(t) \end{bmatrix} \tag{19}$$

$$\begin{bmatrix} \delta z^T(t) & \delta\varphi^T(t) \end{bmatrix} \boldsymbol{M} \begin{bmatrix} \delta z(t) \\ \delta\varphi(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{e}^T(t) & \delta\mathbf{p}^T(t) \end{bmatrix} \boldsymbol{N} \begin{bmatrix} \boldsymbol{e}(t) \\ \delta\mathbf{p}(t) \end{bmatrix} \geq 0 \tag{20}$$

where $\boldsymbol{N}$ takes the form (14). This concludes the proof. ■

3 Reference Residual Model

The reference residual model in the proposed structure provides a pattern that partly separates from the observer data model the interactions represented by cross-bonds in $\boldsymbol{d}(t)$ and $\boldsymbol{f}(t)$ and also those given by residual weight matrices $\boldsymbol{V}_i$ in $\boldsymbol{r}(t)$. Thus, by formalizing RRM in an appropriate mathematical framework, these interaction properties are modified for RRM design while the other related functions are keep together in defined layer of complexity and extensibility.

Taking into account the above, then (12) is reduced to

$$\boldsymbol{r}(t) = \boldsymbol{Ce}(t) \tag{21}$$

and (10) can be rewritten for $r_g = r_f = r_d$ as

$$\dot{\boldsymbol{e}}(t) = \sum_{i=1}^{s} h_i(\theta(t)) \left(\boldsymbol{A}_{ei}\boldsymbol{e}(t) + \boldsymbol{E}_i \mathbf{p}_e(t) + \begin{bmatrix} \boldsymbol{B}_{di} & -\boldsymbol{B}_{fi} \end{bmatrix} \begin{bmatrix} \boldsymbol{d}(t) \\ -\boldsymbol{f}(t) \end{bmatrix} \right) \tag{22}$$

Inserting the same cross-bonds between $\boldsymbol{d}(t)$ and $\boldsymbol{f}(t)$ (22) can be redefined as

$$\dot{\boldsymbol{e}}^{\circ}(t) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) \left(\boldsymbol{A}^{\circ}_{ei}\boldsymbol{e}^{\circ}(t) + \boldsymbol{E}^{\circ}_i \mathbf{p}^{\circ}_e(t) + \begin{bmatrix} \boldsymbol{B}_{di} & -\boldsymbol{B}_{fi} \end{bmatrix} \boldsymbol{T}^{\circ} \begin{bmatrix} \boldsymbol{d}(t) \\ -\boldsymbol{f}(t) \end{bmatrix} \right) \tag{23}$$

$$\boldsymbol{A}^{\circ}_{ei} = \boldsymbol{A}_i - \boldsymbol{J}^{\circ}_i \boldsymbol{C}, \quad \boldsymbol{E}^{\circ}_i = \boldsymbol{E}_i \tag{24}$$

where the cross-bonds matrix $\boldsymbol{T}^{\circ}$ was selected as follows:

$$\boldsymbol{T}^{\circ} = \begin{bmatrix} \boldsymbol{I}_{r_g} & \boldsymbol{I}_{r_g} \\ \boldsymbol{I}_{r_g} & \boldsymbol{I}_{r_g} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_{r_g} \\ \boldsymbol{I}_{r_g} \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_{r_g} & \boldsymbol{I}_{r_g} \end{bmatrix} = \boldsymbol{H}^{\circ}\boldsymbol{H}^{\circ T}, \quad \boldsymbol{H}^{\circ T} = \begin{bmatrix} \boldsymbol{I}_{r_g} & \boldsymbol{I}_{r_g} \end{bmatrix} \tag{25}$$

Denoting

$$\boldsymbol{G}^{\circ}_i = \begin{bmatrix} \boldsymbol{B}_{di} & -\boldsymbol{B}_{fi} \end{bmatrix} \boldsymbol{H}^{\circ}, \quad \boldsymbol{g}^{\circ}(t) = \boldsymbol{H}^{\circ T} \begin{bmatrix} \boldsymbol{d}(t) \\ -\boldsymbol{f}(t) \end{bmatrix} = \boldsymbol{d}(t) - \boldsymbol{f}(t) \tag{26}$$

where $\boldsymbol{G}^{\circ}_i \in \mathbb{R}^{n \times r_d}$, $\boldsymbol{g}^{\circ}(t) \in \mathbb{R}^{r_d}$, RRM can be written as

$$\dot{\boldsymbol{e}}^{\circ}(t) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) \left(\boldsymbol{A}_{ei}\boldsymbol{e}^{\circ}(t) + \boldsymbol{E}_i \delta\mathbf{p}^{\circ}(t) + \boldsymbol{G}^{\circ}_i \boldsymbol{g}^{\circ}(t) \right) \tag{27}$$

$$\boldsymbol{r}^{\circ}(t) = \boldsymbol{C}\boldsymbol{e}^{\circ}(t), \quad \delta\mathbf{p}^{\circ}(t) = \mathbf{p}(t) - \mathbf{p}^{\circ}_e(t) \tag{28}$$

$$\mathbf{p}^{\circ}_e(t) = \varphi \left(\boldsymbol{U}\boldsymbol{q}^{\circ}_e(t) + \boldsymbol{W}\mathbf{p}^{\circ}_e(t) + \boldsymbol{J}^{\circ}\boldsymbol{C}\boldsymbol{e}^{\circ}(t) \right) \tag{29}$$

The observer parameters $\boldsymbol{J}^{\circ} \in \mathbb{R}^{m_p \times m}$, $\boldsymbol{J}^{\circ}_i \in \mathbb{R}^{n \times m}$ for $i = 1, 2, \ldots, s$ have to be designed in such a way that

$$\|\boldsymbol{r}^{\circ}(t)\|^2_{\infty} \leq \gamma^{\circ} \|\boldsymbol{g}^{\circ}(t)\|^2_{\infty} \tag{30}$$

for the square of H_{∞} norm γ° is as small as possible. Since, with respect to (26),

$$\|\boldsymbol{\Gamma}^{\circ}\|_{\infty} = \|\boldsymbol{\Gamma}^{\circ}_d - \boldsymbol{\Gamma}^{\circ}_f\|_{\infty} \geq \|\boldsymbol{\Gamma}^{\circ}_d\|_{\infty} - \|\boldsymbol{\Gamma}^{\circ}_f\|_{\infty} \tag{31}$$

where $\|\boldsymbol{\Gamma}^\circ\|_\infty$ is the H_∞ norm of the residual transfer function matrix with respect to $\boldsymbol{g}^\circ$, then minimizing γ° means maximizing $\|\boldsymbol{\Gamma}_f^\circ\|_\infty$ while minimizing $\|\boldsymbol{\Gamma}_d^\circ\|_\infty$. This minimizes impact of disturbance on the residual signal amplitude.

Remark 1 In order to use the above forms when $r_f \neq r_d$, the corresponding degenerative input matrix is extended by $|r_f - r_d|$ zero columns and the corresponding vector by $|r_f - r_d|$ zero elements to the common dimension $r_g = \max(r_f, r_d)$.

Theorem 1 *The reference residual model (28) and (29) is stable with the quadratic performance γ° if there exist symmetric positive definite matrices $\boldsymbol{P}^\circ \in \mathbb{R}^{n\times n}$, $\boldsymbol{X}^\circ \in \mathbb{R}^{m_p\times m_p}$, $\boldsymbol{Y}^\circ \in \mathbb{R}^{r_p\times r_p}$, matrices $\boldsymbol{Z}^\circ \in \mathbb{R}^{m_p\times m}$, $\boldsymbol{Z}_i^\circ \in \mathbb{R}^{n\times m}$, $i = 1, 2, \ldots, s$, and a positive scalar $\gamma^\circ \in \mathbb{R}$ such that for all i*

$$\boldsymbol{P}^\circ = \boldsymbol{P}^{\circ T} > 0,\ \boldsymbol{X}^\circ = \boldsymbol{X}^{\circ T} > 0,\ \boldsymbol{Y}^\circ = \boldsymbol{Y}^{\circ T} > 0,\ \gamma^\circ > 0 \tag{32}$$

$$\begin{bmatrix} \boldsymbol{P}^\circ\boldsymbol{A}_i + \boldsymbol{A}_i^T\boldsymbol{P}^\circ - \boldsymbol{Z}_i^\circ\boldsymbol{C} - \boldsymbol{C}^T\boldsymbol{Z}_i^{\circ T} & * & * & * & * \\ \boldsymbol{E}_i^{\circ T}\boldsymbol{P}^\circ & -\boldsymbol{Y}^\circ & * & * & * \\ \boldsymbol{G}_i^{\circ T}\boldsymbol{P}^\circ & \boldsymbol{0} & -\gamma^\circ\boldsymbol{I}_{r_g} & * & * \\ \boldsymbol{C} & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{I}_m & * \\ \boldsymbol{X}^\circ\boldsymbol{U} - \boldsymbol{Z}^\circ\boldsymbol{C} & \boldsymbol{X}^\circ\boldsymbol{W} & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{X}^\circ \end{bmatrix} < 0 \tag{33}$$

When the above conditions hold, the reference model gain matrices are given as

$$\boldsymbol{J}_i^\circ = (\boldsymbol{P}^\circ)^{-1}\boldsymbol{Z}_i^\circ, \quad \boldsymbol{J}^\circ = (\boldsymbol{X}^\circ)^{-1}\boldsymbol{Z}^\circ \tag{34}$$

Hereafter, $*$ denotes the symmetric item in a symmetric matrix.

Proof Defining the Lyapunov function candidate

$$v(\boldsymbol{e}^\circ(t)) = \boldsymbol{e}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{e}^\circ(t) + \int_0^t \left(\boldsymbol{r}^{\circ T}(x)\boldsymbol{r}^\circ(x) - \gamma^\circ\,\boldsymbol{g}^{\circ T}(x)\boldsymbol{g}^\circ(x)\right) \mathrm{d}x \tag{35}$$

then, after evaluation the derivative of (35), it is obtained

$$\dot v(\boldsymbol{e}^\circ(t)) = \dot{\boldsymbol{e}}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{e}^\circ(t) + \boldsymbol{e}^{\circ T}(t)\boldsymbol{P}^\circ\dot{\boldsymbol{e}}^\circ(t) + \boldsymbol{r}^{\circ T}(t)\boldsymbol{r}^\circ(t) - \gamma^\circ\,\boldsymbol{g}^{\circ T}(t)\boldsymbol{g}^\circ(t) < 0 \tag{36}$$

Substitution of (27) and (28) into (36) gives

$$\begin{aligned} \dot v(\boldsymbol{e}^\circ(t)) = \sum_{i=1}^s h_i(\boldsymbol{\theta}(t))(\boldsymbol{e}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{E}_i^\circ\delta\mathbf{p}^\circ(t) + \delta\mathbf{p}^{\circ T}(t)\boldsymbol{E}_i^{\circ T}\boldsymbol{P}^\circ\boldsymbol{e}^\circ(t)) - \\ -\gamma^\circ\boldsymbol{g}^{\circ T}(t)\boldsymbol{g}^\circ(t)) + \sum_{i=1}^s h_i(\boldsymbol{\theta}(t))\boldsymbol{e}^{\circ T}(t)(\boldsymbol{A}_{ei}^{\circ T}\boldsymbol{P}^\circ + \boldsymbol{P}^\circ\boldsymbol{A}_{ei}^\circ + \boldsymbol{C}^T\boldsymbol{C})\boldsymbol{e}^\circ(t) + \\ + \sum_{i=1}^s h_i(\boldsymbol{\theta}(t))(\boldsymbol{e}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{G}_i^\circ\boldsymbol{g}^\circ(t) + \boldsymbol{g}^{\circ T}(t)\boldsymbol{G}_i^{\circ T}\boldsymbol{P}^\circ\boldsymbol{e}^\circ(t)) < 0 \end{aligned} \tag{37}$$

Thus, defining the composed vector

$$\boldsymbol{e}_c^{\circ T}(t) = \left[\boldsymbol{e}^{\circ T}(t)\ \delta\mathbf{p}^{\circ T}(t)\ \boldsymbol{g}^{\circ T}(t)\right] \tag{38}$$

(37) can be defined as follows

$$\dot{v}(\boldsymbol{e}^{\circ}(t)) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t))\boldsymbol{e}_c^{\circ T}(t)\boldsymbol{P}_{ci}^{\circ}\boldsymbol{e}_c^{\circ}(t) \leq -\boldsymbol{e}_c^{\circ T}(t)\boldsymbol{N}^{\diamond}\boldsymbol{e}_c^{\circ}(t) < 0 \tag{39}$$

$$\boldsymbol{P}_{ci}^{\circ} = \begin{bmatrix} \boldsymbol{P}^{\circ}\boldsymbol{A}_{ei}^{\circ} + \boldsymbol{A}_{ei}^{\circ T}\boldsymbol{P}^{\circ} + \boldsymbol{C}^T\boldsymbol{C} & * & * \\ \boldsymbol{E}_i^{\circ T}\boldsymbol{P}^{\circ} & \mathbf{0} & * \\ \boldsymbol{G}_i^{\circ T}\boldsymbol{P}^{\circ} & \mathbf{0} & -\gamma^{\circ}\boldsymbol{I}_{r_g} \end{bmatrix}, \quad \boldsymbol{N}^{\diamond} = \operatorname{diag}\left[\boldsymbol{N}^{\circ}\ \mathbf{0}\right] \tag{40}$$

which gives

$$\dot{v}(\boldsymbol{e}^{\circ}(t)) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t))\boldsymbol{e}_c^{\circ T}(t)(\boldsymbol{P}_{ci}^{\circ} + \boldsymbol{N}^{\diamond})\boldsymbol{e}_c^{\circ}(t) < 0 \tag{41}$$

Introducing $\boldsymbol{U}_e^{\circ}$ and defining the incremental multiplier matrix as follows

$$\boldsymbol{U}_e^{\circ} = \boldsymbol{U} - \boldsymbol{J}^{\circ}\boldsymbol{C}, \quad \boldsymbol{M}^{\circ} = \operatorname{diag}\left[\boldsymbol{X}^{\circ}\ -\boldsymbol{Y}^{\circ}\right] \tag{42}$$

where $\boldsymbol{X}^{\circ} \in \mathbb{R}^{m_p \times m_p}$, $\boldsymbol{Y}^{\circ} \in \mathbb{R}^{r_p \times r_p}$ are symmetric positive definite matrices, then (14) and (19) implies

$$\begin{aligned} \boldsymbol{N}^{\circ} &= \begin{bmatrix} \boldsymbol{U}_e^{\circ T} & \mathbf{0} \\ \boldsymbol{W}^T & \boldsymbol{I}_{r_p} \end{bmatrix} \begin{bmatrix} \boldsymbol{X}^{\circ} & \mathbf{0} \\ \mathbf{0} & -\boldsymbol{Y}^{\circ} \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_e^{\circ} & \boldsymbol{W} \\ \mathbf{0} & \boldsymbol{I}_{r_p} \end{bmatrix} = \\ &= \begin{bmatrix} \boldsymbol{U}_e^{\circ T}\boldsymbol{X}^{\circ} \\ \boldsymbol{W}^T\boldsymbol{X}^{\circ} \end{bmatrix} (\boldsymbol{X}^{\circ})^{-1} \left[\boldsymbol{X}^{\circ}\boldsymbol{U}_e^{\circ}\ \boldsymbol{X}^{\circ}\boldsymbol{W}\right] - \begin{bmatrix} \mathbf{0} \\ \boldsymbol{I}_{r_p} \end{bmatrix} \boldsymbol{Y}^{\circ} \left[\mathbf{0}\ \boldsymbol{I}_{r_p}\right] \end{aligned} \tag{43}$$

Thus, (41) is negative if $(\boldsymbol{P}_{ci}^{\circ} + \boldsymbol{N}^{\diamond})$ is negative definite, that is the matrix sum

$$\begin{bmatrix} \boldsymbol{U}_e^T\boldsymbol{X}^{\circ} \\ \boldsymbol{W}^T\boldsymbol{X}^{\circ} \\ \mathbf{0} \end{bmatrix} (\boldsymbol{X}^{\circ})^{-1} \left[\boldsymbol{X}^{\circ}\boldsymbol{U}_e\ \boldsymbol{X}^{\circ}\boldsymbol{W}\ \mathbf{0}\right] + \begin{bmatrix} \boldsymbol{P}^{\circ}\boldsymbol{A}_{ei}^{\circ} + \boldsymbol{A}_{ei}^{\circ T}\boldsymbol{P}^{\circ} + \boldsymbol{C}^T\boldsymbol{C} & * & * \\ \boldsymbol{E}_i^{\circ T}\boldsymbol{P}^{\circ} & -\boldsymbol{Y}^{\circ} & * \\ \boldsymbol{G}_i^{\circ T}\boldsymbol{P}^{\circ} & \mathbf{0} & -\gamma^{\circ}\boldsymbol{I}_{r_g} \end{bmatrix} \tag{44}$$

has to be negative definite. Applying twice the Schur complement property and using, with respect to (24) and (42), the notations

$$\boldsymbol{Z}_i^{\circ} = \boldsymbol{P}^{\circ}\boldsymbol{J}_i^{\circ}, \quad \boldsymbol{Z}^{\circ} = \boldsymbol{X}^{\circ}\boldsymbol{J}^{\circ} \tag{45}$$

then (44) implies (33). This concludes the proof. ■

4 FDF Design Condition

To rebind $\boldsymbol{r}(t)$ with RRM, the overall FDF model, incorporating (27) and (28) with $\boldsymbol{T}^{\circ} = \boldsymbol{I}_{2r_g}$, can be expressed as

$$\dot{\boldsymbol{e}}^{\bullet}(t) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) \left(\boldsymbol{A}_{ci}^{\bullet} \boldsymbol{e}^{\bullet}(t) + \boldsymbol{G}_i^{\bullet} \boldsymbol{g}^{\bullet}(t) + \boldsymbol{E}_i^{\bullet} \delta \mathbf{p}^{\bullet}(t) \right) \tag{46}$$

$$\boldsymbol{r}^{\bullet}(t) = \boldsymbol{r}(t) - \boldsymbol{r}^{\circ}(t) = \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) \boldsymbol{V}_i^{\star} \boldsymbol{C}^{\star} \boldsymbol{e}^{\bullet}(t) \tag{47}$$

$$\boldsymbol{A}_{ei}^{\bullet} = \boldsymbol{A}_i^{\bullet} - \boldsymbol{J}_i^{\bullet} \boldsymbol{C}_i^{\bullet}, \; \boldsymbol{E}_i^{\bullet} = \operatorname{diag} \left[\boldsymbol{E}_i \;\; \boldsymbol{E}_i \right], \quad \boldsymbol{V}_i^{\star} = \left[\boldsymbol{V}_i \; -\boldsymbol{I}_m \right] \tag{48}$$

$$\boldsymbol{A}_i^{\bullet} = \operatorname{diag} \left[\boldsymbol{A}_i \;\; \boldsymbol{A}_i \right], \quad \boldsymbol{C}_i^{\bullet} = \operatorname{diag} \left[\boldsymbol{C} \; \boldsymbol{J}_i^{\circ} \boldsymbol{C} \right], \quad \boldsymbol{J}_i^{\bullet} = \operatorname{diag} \left[\boldsymbol{J}_i \; \boldsymbol{I}_n \right] \tag{49}$$

$$\boldsymbol{C}^{\star} = \operatorname{diag} \left[\boldsymbol{C} \; \boldsymbol{C} \right], \quad \boldsymbol{U}_e^{\star} = \boldsymbol{U} - \boldsymbol{J}\boldsymbol{C}, \quad \boldsymbol{U}_e^{\circ} = \boldsymbol{U} - \boldsymbol{J}^{\circ}\boldsymbol{C} \tag{50}$$

$$\boldsymbol{e}^{\bullet}(t) = \begin{bmatrix} \boldsymbol{e}(t) \\ \boldsymbol{e}^{\circ}(t) \end{bmatrix}, \; \delta \mathbf{p}^{\bullet}(t) = \begin{bmatrix} \delta \mathbf{p}(t) \\ \delta \mathbf{p}^{\circ}(t) \end{bmatrix}, \; \boldsymbol{g}^{\bullet}(t) = \begin{bmatrix} \boldsymbol{f}(t) \\ \boldsymbol{d}(t) \end{bmatrix}, \; \boldsymbol{G}_i^{\bullet} = \begin{bmatrix} \boldsymbol{B}_{fi} \; \boldsymbol{B}_{di} \\ \boldsymbol{B}_{fi} \; \boldsymbol{B}_{di} \end{bmatrix} \tag{51}$$

and $\boldsymbol{e}^{\bullet}(t) \in \mathbb{R}^{2n}$, $\boldsymbol{g}^{\bullet}(t) \in \mathbb{R}^{2r_d}$, $\boldsymbol{G}_i^{\bullet} \in \mathbb{R}^{2n \times 2r_d}$, $\boldsymbol{A}_i^{\bullet} \in \mathbb{R}^{2n \times 2n}$, $\boldsymbol{V}_i^{\star} \in \mathbb{R}^{m \times 2m}$, $\boldsymbol{J}_i^{\bullet} \in \mathbb{R}^{2n \times (m+n)}$, $\boldsymbol{E}_i^{\bullet} \in \mathbb{R}^{2n \times 2r_p}$, $\boldsymbol{C}_i^{\bullet} \in \mathbb{R}^{(m+n) \times 2n}$, $\boldsymbol{U}_e^{\star}, \boldsymbol{U}_e^{\circ} \in \mathbb{R}^{m_p \times n}$, $\boldsymbol{C}^{\star} \in \mathbb{R}^{m \times 2n}$. It should be noted that these matrix structures must be defined for the existence of structured LMI matrix variables.

Remark 2 According to (47), formulation of the optimization criterion means that the double summation through membership functions occurs in the product $\boldsymbol{r}^{\bullet T}(t)\boldsymbol{r}^{\bullet}(t)$. Since $\sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) = 1$, for solving this optimization problem the following approximation is applied (proof see, e.g., in [13])

$$\begin{aligned} \boldsymbol{r}^{\bullet T}(t)\boldsymbol{r}^{\bullet}(t) = \boldsymbol{e}^{\bullet T}(t) \sum_{i=1}^{s} \sum_{j=1}^{s} h_i(\boldsymbol{\theta}(t)) h_j(\boldsymbol{\theta}(t)) \boldsymbol{C}^{\star T} \boldsymbol{V}_j^{\star T} \boldsymbol{V}_i^{\star} \boldsymbol{C}^{\circ} \boldsymbol{e}^{\bullet}(t) \leq \\ \leq \boldsymbol{e}^{\bullet T}(t) \sum_{i=1}^{s} h_i(\boldsymbol{\theta}(t)) \boldsymbol{C}^{\star T} \boldsymbol{V}_i^{\star T} \boldsymbol{V}_i^{\star} \boldsymbol{C}^{\star} \boldsymbol{e}^{\bullet}(t) \end{aligned} \tag{52}$$

Theorem 2 *The fault detection filter (46) and (47) associated with RRM (28) and (29) is stable with the quadratic performance $\gamma^{\bullet}$ if there exist symmetric positive definite matrices $\boldsymbol{P}_1^{\star}, \boldsymbol{P}_2^{\star} \in \mathbb{R}^{n \times n}$, $\boldsymbol{X}^{\star} \in \mathbb{R}^{m_p \times m_p}$, $\boldsymbol{Y}^{\star} \in \mathbb{R}^{r_p \times r_p}$, matrices $\boldsymbol{Z}^{\star} \in \mathbb{R}^{m_p \times m}$, $\boldsymbol{Z}_i^{\star} \in \mathbb{R}^{n \times m}$, $\boldsymbol{V}_i \in \mathbb{R}^{m \times n}$, $i = 1, 2, \ldots, s$ and a positive scalars $\gamma^{\bullet} \in \mathbb{R}$ such that for all i*

$$\boldsymbol{P}_1^{\star} = \boldsymbol{P}_1^{\star T} > 0, \; \boldsymbol{P}_2^{\star} = \boldsymbol{P}_2^{\star T} > 0, \; \boldsymbol{X}^{\star} = \boldsymbol{X}^{\star T} > 0, \; \boldsymbol{Y}^{\star} = \boldsymbol{Y}^{\star T} > 0, \; \gamma^{\bullet} > 0 \tag{53}$$

$$\begin{bmatrix} \boldsymbol{P}^{\bullet}\boldsymbol{A}_i^{\bullet} + \boldsymbol{A}_i^{\bullet T}\boldsymbol{P}^{\bullet} - \boldsymbol{Z}_i^{\bullet}\boldsymbol{C}_i^{\bullet} - \boldsymbol{C}_i^{\bullet T}\boldsymbol{Z}_i^{\bullet T} & * & * & * & * \\ \boldsymbol{E}_i^{\bullet T}\boldsymbol{P}^{\bullet} & -\boldsymbol{I}^{\bullet}\boldsymbol{Y}^{\bullet}\boldsymbol{I}^{\bullet} & * & * & * \\ \boldsymbol{G}_i^{\bullet T}\boldsymbol{P}^{\bullet} & \boldsymbol{0} & -\gamma^{\bullet}\boldsymbol{I}_{2r_g} & * & * \\ \boldsymbol{V}_i^{\star}\boldsymbol{C}^{\star} & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{I}_m & * \\ \boldsymbol{X}^{\bullet}\boldsymbol{U}^{\bullet} - \boldsymbol{Z}^{\bullet}\boldsymbol{C}^{\bullet} & \boldsymbol{X}^{\bullet}\boldsymbol{W}^{\bullet} & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{X}^{\bullet} \end{bmatrix} < 0 \tag{54}$$

$$\boldsymbol{P}^{\bullet} = diag\left[\boldsymbol{P}_1^{\star}\ \boldsymbol{P}_2^{\star}\right], \quad \boldsymbol{Z}_i^{\bullet} = diag\left[\boldsymbol{Z}_i^{\star}\ \boldsymbol{P}_2^{\star}\right], \quad \boldsymbol{Z}^{\bullet} = diag\left[\boldsymbol{Z}^{\star}\ \boldsymbol{I}_{m_p}\right] \tag{55}$$

$$\boldsymbol{X}^{\bullet} = diag\left[\boldsymbol{X}^{\star}\ \boldsymbol{I}_{m_p}\right], \quad \boldsymbol{Y}^{\bullet} = diag\left[\boldsymbol{Y}^{\star}\ \boldsymbol{I}_{r_p}\right], \quad \boldsymbol{V}_i^{\star} = \left[\boldsymbol{V}_i\ -\boldsymbol{I}_m\right] \tag{56}$$

$$\boldsymbol{U}^{\bullet} = diag\left[\boldsymbol{U}\ \sqrt{\boldsymbol{X}^{\circ}}\boldsymbol{U}_e^{\circ}\right], \quad \boldsymbol{C}^{\bullet} = diag\left[\boldsymbol{C}\ \sqrt{\boldsymbol{X}^{\circ}}\boldsymbol{U}_e^{\circ}\right] \tag{57}$$

$$\boldsymbol{W}^{\bullet} = diag\left[\boldsymbol{W}\ \sqrt{\boldsymbol{X}^{\circ}}\boldsymbol{W}\right], \quad \boldsymbol{I}^{\bullet} = diag\left[\boldsymbol{I}_{r_p}\ \sqrt{\boldsymbol{Y}^{\circ}}\right] \tag{58}$$

where $\boldsymbol{P}^{\bullet} \in \mathbb{R}^{2n\times 2n}$, $\boldsymbol{Z}_i^{\bullet} \in \mathbb{R}^{2n\times(m+n)}$, $\boldsymbol{Y}^{\bullet} \in \mathbb{R}^{2r_p\times 2r_p}$, $\boldsymbol{V}_i^{\star} \in \mathbb{R}^{m_r\times(m+m_r)}$, $\boldsymbol{X}^{\bullet} \in \mathbb{R}^{m_p\times 2m_p}$, $\boldsymbol{Z}^{\bullet} \in \mathbb{R}^{m_p\times 2m_p}$ *are structured matrix variables and all remaining matrix parameters are defined in (48)–(51).*

When the above conditions are satisfied, then

$$\boldsymbol{J}_i = (\boldsymbol{P}_1^{\star})^{-1}\boldsymbol{Z}_i^{\star}, \quad \boldsymbol{J} = (\boldsymbol{X}^{\star})^{-1}\boldsymbol{Z}^{\star}, \quad \boldsymbol{V}_i = \boldsymbol{V}_i^{\star}\left[\boldsymbol{I}_m\ \boldsymbol{0}\right]^T \tag{59}$$

Proof Now the Lyapunov function candidate is defined as

$$v(\boldsymbol{e}^{\bullet}(t)) = \boldsymbol{e}^{\bullet T}(t)\boldsymbol{P}^{\bullet}\boldsymbol{e}^{\bullet}(t) + \int_0^t \left(\boldsymbol{r}^{\bullet T}(x)\boldsymbol{r}^{\bullet}(x) - \gamma^{\bullet}\boldsymbol{g}^{\bullet T}(x)\boldsymbol{g}^{\bullet}(x)\right) \mathrm{d}x \tag{60}$$

and its time derivative is

$$\begin{aligned} &\dot{v}(\boldsymbol{e}^{\bullet}(t)) \\ &= \dot{\boldsymbol{e}}^{\bullet T}(t)\boldsymbol{P}^{\bullet}\boldsymbol{e}^{\bullet}(t) + \boldsymbol{e}^{\bullet T}(t)\boldsymbol{P}^{\bullet}\dot{\boldsymbol{e}}^{\bullet}(t) + \boldsymbol{r}^{\bullet T}(t)\boldsymbol{r}^{\bullet}(t) - \gamma^{\bullet}\boldsymbol{g}^{\bullet T}(t)\boldsymbol{g}^{\bullet}(t) < 0 \end{aligned} \tag{61}$$

where the structure of $\boldsymbol{A}_i^{\bullet}$, $\boldsymbol{C}_i^{\bullet}$ implies the structure of $\boldsymbol{P}^{\bullet}$ in (58).

Considering the property (52) and substituting (46) and (47) into (61) results in

$$\begin{aligned} \dot{v}(\boldsymbol{e}^{\bullet}(t)) \le\ & -\gamma^{\bullet}\boldsymbol{g}^{\bullet T}(t)\boldsymbol{g}^{\bullet}(t) + \sum_{i=1}^{s} \mathrm{h}_i(\boldsymbol{\theta}(t))\boldsymbol{e}^{\bullet T}(t)\boldsymbol{C}^{\star T}\boldsymbol{V}_i^{\star T}\boldsymbol{V}_i^{\star}\boldsymbol{C}^{\star}\boldsymbol{e}^{\bullet}(t) \\ &+ \sum_{i=1}^{s} \mathrm{h}_i(\boldsymbol{\theta}(t))(\boldsymbol{A}_{ei}^{\bullet}\boldsymbol{e}^{\bullet}(t) + \boldsymbol{G}_i^{\bullet}\boldsymbol{g}^{\bullet}(t) + \boldsymbol{E}_i^{\bullet}\delta\mathbf{p}^{\bullet}(t))^T\boldsymbol{P}^{\bullet}\boldsymbol{e}^{\bullet}(t) \\ &+ \sum_{i=1}^{s} \mathrm{h}_i(\boldsymbol{\theta}(t))\boldsymbol{e}^{\bullet T}(t)\boldsymbol{P}^{\bullet}(\boldsymbol{A}_{ei}^{\bullet}\boldsymbol{e}^{\bullet}(t) + \boldsymbol{G}_i^{\bullet}\boldsymbol{g}^{\bullet}(t) + \boldsymbol{E}_i^{\bullet}\delta\mathbf{p}^{\bullet}(t)) < 0 \end{aligned} \tag{62}$$

and with the notation

$$\boldsymbol{e}_c^{\bullet T}(t) = \left[\boldsymbol{e}^{\bullet T}(t)\ \delta\mathbf{p}^{\bullet T}(t)\ \ \boldsymbol{g}^{\bullet T}(t)\right] \tag{63}$$

the time derivative of $v(\boldsymbol{e}^{\bullet}(t))$ can be prescribe (in analogy with (41)) as

$$\dot{v}(\boldsymbol{e}^{\bullet}(t)) \le \sum_{i=1}^{s} \mathrm{h}_i(\boldsymbol{\theta}(t))\boldsymbol{e}_c^{\bullet T}(t)(\boldsymbol{P}_{ci}^{\bullet} + \boldsymbol{N}^{\triangleright})\boldsymbol{e}_c^{\bullet}(t) < 0 \tag{64}$$

where $\boldsymbol{N}^{\triangleright}$ is a positive definite matrix, $(\boldsymbol{P}_{ci}^{\bullet} + \boldsymbol{N}^{\triangleright})$ is negative definite and

$$\boldsymbol{P}_{ci}^{\bullet} = \begin{bmatrix} \boldsymbol{P}^{\bullet}\boldsymbol{A}_{ei}^{\bullet} + \boldsymbol{A}_{ei}^{\bullet T}\boldsymbol{P}^{\bullet} + \boldsymbol{C}^{\star T}\boldsymbol{V}_i^{\star T}\boldsymbol{V}_i^{\star}\boldsymbol{C}^{\circ} & * & * \\ \boldsymbol{E}_i^{\bullet T}\boldsymbol{P}^{\bullet} & \mathbf{0} & * \\ \boldsymbol{G}_i^{\bullet T}\boldsymbol{P}^{\bullet} & \mathbf{0} & -\gamma^{\bullet}\boldsymbol{I}_{2r_g} \end{bmatrix} \tag{65}$$

Defining the incremental multiplier matrix with respect to the structure of (63) as follows

$$\boldsymbol{M}^{\bullet} = \mathrm{diag}\left[\boldsymbol{X}^{\star}\ \boldsymbol{X}^{\circ}\ -\boldsymbol{Y}^{\star}\ -\boldsymbol{Y}^{\circ}\right] \tag{66}$$

where $\boldsymbol{X}^{\star} \in \mathbb{R}^{m_p \times m_p}$, $\boldsymbol{Y}^{\star} \in \mathbb{R}^{r_p \times r_p}$ are symmetric positive definite matrices and $\boldsymbol{X}^{\circ}$, $\boldsymbol{Y}^{\circ}$ are the constant matrices satisfying (32) and (33), then it is

$$\boldsymbol{N}^{\bullet} = \begin{bmatrix} \boldsymbol{U}_e^{\star T} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}_e^{\circ T} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{W}^T & \mathbf{0} & \boldsymbol{I}_{r_p} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{W}^T & \mathbf{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \begin{bmatrix} \boldsymbol{X}^{\star} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{X}^{\circ} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\boldsymbol{Y}^{\star} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\boldsymbol{Y}^{\circ} \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_e^{\star} & \mathbf{0} & \boldsymbol{W} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}_e^{\circ} & \mathbf{0} & \boldsymbol{W} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{I}_{r_p} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \tag{67}$$

and (67) can be separated in the two components

$$\boldsymbol{N}_1^{\bullet} = \begin{bmatrix} \boldsymbol{U}_e^{\star T}\boldsymbol{X}^{\star} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}_e^{\circ T}\sqrt{\boldsymbol{X}^{\circ}} \\ \boldsymbol{W}^T\boldsymbol{X}^{\star} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{W}^T\sqrt{\boldsymbol{X}^{\circ}} \end{bmatrix} \begin{bmatrix} \boldsymbol{X}^{\circ} \\ \mathbf{0}\ \ \boldsymbol{I}_{m_p} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{X}^{\star}\boldsymbol{U}_e^{\star} & \mathbf{0} & \boldsymbol{X}^{\star}\boldsymbol{W} & \mathbf{0} \\ \mathbf{0} & \sqrt{\boldsymbol{X}^{\circ}}\boldsymbol{U}_e^{\circ} & \mathbf{0} & \sqrt{\boldsymbol{X}^{\circ}}\boldsymbol{W} \end{bmatrix} \tag{68}$$

$$\boldsymbol{N}_2^{\bullet} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \boldsymbol{I}_{r_p} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \begin{bmatrix} -\boldsymbol{Y}^{\star} & \mathbf{0} \\ \mathbf{0} & -\boldsymbol{Y}^{\circ} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{0} & \boldsymbol{I}_{r_p} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I}_{r_p} \end{bmatrix} \tag{69}$$

To obtain relationships that allow to use structured matrix variables, it can be written with $\boldsymbol{U}_e^{\star}$, $\boldsymbol{U}_e^{\circ}$ given in (50)

$$\begin{bmatrix} X^{\star}U_e^{\star} & \mathbf{0} \\ \mathbf{0} & \sqrt{X^{\circ}}U_e^{\circ} \end{bmatrix} = \begin{bmatrix} X^{\star} & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} \begin{bmatrix} U & \mathbf{0} \\ \mathbf{0} & \sqrt{X^{\circ}}U_e^{\circ} \end{bmatrix} - \begin{bmatrix} X^{\star}J & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} \begin{bmatrix} C & \mathbf{0} \\ \mathbf{0} & \sqrt{X^{\circ}}U_e^{\circ} \end{bmatrix} \tag{70}$$

$$\begin{bmatrix} X^{\star}W & \mathbf{0} \\ \mathbf{0} & \sqrt{X^{\circ}}W \end{bmatrix} = \begin{bmatrix} X^{\star} & \mathbf{0} \\ \mathbf{0} & I_{m_p} \end{bmatrix} \begin{bmatrix} W & \mathbf{0} \\ \mathbf{0} & \sqrt{X^{\circ}}W \end{bmatrix} \tag{71}$$

and with the notations (56)–(58), where $Z^{\star} = X^{\star}J$, (68) can be written as

$$N_1^{\bullet} = \begin{bmatrix} (X^{\bullet}U^{\bullet} - Z^{\bullet}C^{\bullet})^T \\ W^{\bullet T}X^{\bullet} \end{bmatrix} (X^{\bullet})^{-1} \begin{bmatrix} X^{\bullet}U^{\bullet} - Z^{\bullet}C^{\bullet} & X^{\bullet}W^{\bullet} \end{bmatrix} \tag{72}$$

Since the connecting matrix element in (69) can be factorized as

$$-\begin{bmatrix} Y^{\star} & \mathbf{0} \\ \mathbf{0} & Y^{\circ} \end{bmatrix} = -\begin{bmatrix} I_{r_p} & \mathbf{0} \\ \mathbf{0} & \sqrt{Y^{\circ}} \end{bmatrix} \begin{bmatrix} Y^{\star} & \mathbf{0} \\ \mathbf{0} & I_{r_p} \end{bmatrix} \begin{bmatrix} I_{r_p} & \mathbf{0} \\ \mathbf{0} & \sqrt{Y^{\circ}} \end{bmatrix} = -I^{\bullet}Y^{\bullet}I^{\bullet} \tag{73}$$

then, using the same procedure as in redaction of the matrix (44), from (65), (72) and (69), (73), the following can be obtained

$$\begin{bmatrix} P^{\bullet}A_{ei}^{\bullet} + A_{ei}^{\bullet T}P^{\bullet} & * & * & * & * \\ E_i^{\bullet T}P^{\bullet} & -I^{\bullet}Y^{\bullet}I^{\bullet} & * & * & * \\ G_i^{\bullet T}P^{\bullet} & \mathbf{0} & -\gamma I_{2r_g} & * & * \\ V_i^{\star}C^{\star} & \mathbf{0} & \mathbf{0} & -I_m & * \\ X^{\bullet}U^{\bullet} - Z^{\bullet}C^{\bullet} & X^{\bullet}W^{\bullet} & \mathbf{0} & \mathbf{0} & -X^{\bullet} \end{bmatrix} < 0 \tag{74}$$

Using the block diagonal matrix $P^{\bullet}$ and $A_{ei}^{\bullet}$ given in (48), (49) and (55), where $Z_i^{\star} = P_1^{\star}J_i$, then it can be written

$$P^{\bullet}A_{ei}^{\bullet} = \begin{bmatrix} P_1^{\star} & \mathbf{0} \\ \mathbf{0} & P_2^{\star} \end{bmatrix} \begin{bmatrix} A_i & \mathbf{0} \\ \mathbf{0} & A_i \end{bmatrix} - \begin{bmatrix} P_1^{\star}J_i & \mathbf{0} \\ \mathbf{0} & P_2^{\star} \end{bmatrix} \begin{bmatrix} C & \mathbf{0} \\ \mathbf{0} & J_i^{\circ}C \end{bmatrix} = P^{\bullet}A_i^{\bullet} - Z_i^{\bullet}C^{\bullet} \tag{75}$$

Thus (74) with (75) implies (54). This concludes the proof.

5 Illustrative Example

As a simple illustrative model, the nonlinear dynamics of the ball-and-beam system, represented by the nonlinear state-space model, was taken from [6], where

$$\begin{aligned}
\dot{q}_1(t) &= q_2(t)\,, & z(t) &= q_1(t)\\
\dot{q}_2(t) &= a\left(q_1(t)q_4^2(t) - g\sin(q_3(t))\right), & y_1(t) &= q_1(t)\\
\dot{q}_3(t) &= q_4(t)\,, & y_2(t) &= q_3(t)\\
\dot{q}_4(t) &= b\left(-q_1(t) + gu(t)\right)
\end{aligned}$$

while the input variable $u(t)$ is the angular acceleration of the beam [rad/s^2], the output variable $z(t)$ is equal $q_1(t)$ and the measured variables are $q_1(t)$ and $q_3(t)$, while $q_1(t)$ is the position of the ball [m], $q_2(t)$ is the velocity of the ball [m/s], $q_3(t)$ is the angle of the beam [rad] and $q_4(t)$ is the angular velocity of the beam [rad/s]. The model parameters are

$$a = \frac{m}{m + \frac{J}{r^2}} = 0.7143, \quad b = \frac{m}{J + J_b} = 1.1$$

J— the ball inertia moment $1.76\,10^{-5}$ kg m^2, m—the mass of the ball 0.11 kg
J_b—the beam inertia moment 0.1 kg m^2, r—the radius of the ball 0.02 m
g—the gravitational constant 9.81 m/s^2.

Introducing the premise variable $\theta(t) = q_1(t)q_4(t)$ which is bounded in the sector $q_1(t)q_4(t) \in \langle -d, d\rangle = \langle -5, 5\rangle$, the associated membership functions are

$$h_2(\theta(t)) = \begin{cases} 1, & \theta(t) \geq d \\ \frac{1}{d}\theta(t), & 0 < \theta(t) < d, \\ 0, & \theta(t) \leq 0 \end{cases} \quad h_3(\theta(t)) = \begin{cases} 0, & \theta(t) \geq d \\ -\frac{1}{d}\theta(t), & 0 > \theta(t) > -d \\ 1, & \theta(t) \leq -d \end{cases}$$

$$h_1(\theta(t)) = 1 - h_2(\theta(t)) - h_3(\theta(t))$$

It is supposed that FDF is designed to support the fault detection in the structure with unmeasurable $q_4(t)$ and the nonlinear function $\mathbf{p}(t)$ is therefore given as

$$\mathbf{p}(t) = \sin(q_4(t)) = \sin\left(\begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}\boldsymbol{q}(t)\right) = \sin(\boldsymbol{U}\boldsymbol{q}(t) + \boldsymbol{W}\mathbf{p}(t))\,,$$

$$\boldsymbol{q}^T(t) = \begin{bmatrix} q_1(t) & q_2(t) & q_3(t) & q_4(t) \end{bmatrix}, \quad \boldsymbol{U} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{W} = 0$$

Consequently, the representation in the TS fuzzy system model gives for all i

$$\boldsymbol{A}_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -b & 0 & 0 & 0 \end{bmatrix}, \boldsymbol{A}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & ad \\ 0 & 0 & 0 & 1 \\ -b & 0 & 0 & 0 \end{bmatrix}, \boldsymbol{A}_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -ad \\ 0 & 0 & 0 & 1 \\ -b & 0 & 0 & 0 \end{bmatrix}, \boldsymbol{C}^T = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix},$$

$$\boldsymbol{B}_i = \boldsymbol{B}_{fi} = \boldsymbol{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ bg \end{bmatrix}, \quad \boldsymbol{B}_{di} = \boldsymbol{B}_d = \begin{bmatrix} 0.1 \\ 0.2 \\ 0.5 \\ 0.4 \end{bmatrix}, \quad \boldsymbol{E}_i = \boldsymbol{E} = \begin{bmatrix} 0 \\ -ag \\ 0 \\ 0 \end{bmatrix}$$

Within the above parameters and Theorem 1 and by solving LMIs (32) and (33) with SeDuMi packet, the RRM gains were found as

$$\gamma^\circ = 4.2758\,, \quad \boldsymbol{J}^\circ = \begin{bmatrix} -0.0071 & 0.0916 \end{bmatrix}$$

$$\boldsymbol{J}_1^\circ = \begin{bmatrix} 12.0282 & 0.3613 \\ 120.9501 & 2.9155 \\ 1.0973 & 17.4208 \\ 0.6048 & 108.0026 \end{bmatrix}, \boldsymbol{J}_2^\circ = \begin{bmatrix} 11.7658 & 0.4128 \\ 117.9370 & 3.4767 \\ -5.8114 & 17.2653 \\ -46.7773 & 106.9042 \end{bmatrix}$$

$$\boldsymbol{J}_3^\circ = \begin{bmatrix} 12.2317 & 0.3081 \\ 123.3139 & 2.3357 \\ 7.9978 & 17.5818 \\ 47.9626 & 109.1395 \end{bmatrix}$$

Subsequently, by Theorem 2, the parameters of the stable FDF were computed as follows

$$\gamma^\bullet = 1.7725, \quad \boldsymbol{J} = \begin{bmatrix} -0.0064 & 0.1231 \end{bmatrix}$$

$$\boldsymbol{J}_1 = \begin{bmatrix} 15.3748 & 0.3219 \\ 107.4360 & 1.7553 \\ 0.0468 & 17.0285 \\ -2.3693 & 138.1231 \end{bmatrix}, \boldsymbol{J}_2 = \begin{bmatrix} 15.2233 & 0.5052 \\ 106.2246 & 3.1328 \\ -4.9619 & 16.9679 \\ -45.6834 & 137.4746 \end{bmatrix}$$

$$\boldsymbol{J}_3 = \begin{bmatrix} 15.4152 & 0.1386 \\ 107.8083 & 0.3748 \\ 5.0541 & 17.0252 \\ 40.9506 & 138.0520 \end{bmatrix}$$

$$\boldsymbol{V}_1 = \begin{bmatrix} 0.0005 & -0.0006 \\ -0.0003 & 0.1467 \end{bmatrix}, \boldsymbol{V}_2 = \begin{bmatrix} 0.0006 & 0.0241 \\ 0.0008 & 0.1473 \end{bmatrix}, \boldsymbol{V}_3 = \begin{bmatrix} 0.0008 & -0.0263 \\ -0.0014 & 0.1459 \end{bmatrix}$$

It should be noted that this example at first explains the inclusion of RRM and incremental quadratic constraints in FDF design.

6 Concluding Remarks

The introduced nonlinear fuzzy FDF design method based on RRM is presented in the paper. This is achieved by application of Lyapunov function and incremental quadratic constraints parameterized by a symmetric multiplier matrix. In the presented version, the sensitivity of the reference residual model and FDF stability problem is solved, considering premise variables determined from the subsets of measurable and unmeasurable state variables. It is obvious that the adaptation methodology proposed to TS fuzzy state observer is imminent.

It should be pointed out that the proposed technique using TS fuzzy models with nonlinear terms might give more conservative results than the existing approaches in some cases, but the advantage of them lies in designing a problem oriented fuzzy FDF with fewer rules and less computational burden. It is clear that in specific cases it is necessary to have a compromise between the complexity of the implemented method and the number of LMIs to be solved.

Acknowledgments The work presented in this paper was supported by VEGA, the Grant Agency of the Ministry of Education and the Academy of Science of Slovak Republic under Grant No. 1/0348/14. This support is very gratefully acknowledged.

References

1. Acikmese, A.B., Corless, M.: Observers for systems with nonlinearities satisfying an incremental quadratic inequality. In: Proceedings of 2005 American Control Conference, pp. 3622–3629. Portland, OR, USA (2005)
2. Acikmese, A.B., Corless, M.: Observers for systems with nonlinearities satisfying incremental quadratic constraints. Automatica **47**(7), 1339–1348 (2011)
3. Bai, L., Tian, Z., Shi, S.: Design of H_∞ robust fault detection filter for linear uncertaintime-delay systems. ISA Trans. **45**(4), 491–502 (2006)
4. Bai, L., Tian, Z., Shi, S.: Robust fault detection for a class of nonlinear time-delay systems. J. Frankl. Inst. **344**(6), 873–888 (2007)
5. Castaldi, P., Mimmo, N., Simani, S.: Differential geometry based active fault tolerant control for aircraft. Control engineering practice **32**, 227–235 (2014)
6. Chang, Y.H., Chan, W.S., Chang, C.W., Tao, C.W.: Adaptive fuzzy dynamic surface control for ball and beam system. Int. J. Fuzzy Syst. **13**(1), 1–7 (2011)
7. De Persis, C., Isidori, A.: A geometric approach to nonlinear fault detection and isolation. IEEE Trans. Autom. Control **45**(6), 853–865 (2001)
8. Ding, S.X.: Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools. Springer, Berlin (2013)
9. Gao, Z., Shi, X., Ding, S.X.: Fuzzy state/disturbance observer design for T-S fuzzy systems with application to sensor fault estimation. IEEE Trans. Syst. Man Cybern. Part B Cybern. **38**(3), 875–880 (2008)
10. Guo, J., Huang, X., Cui, Y.: Design and analysis of robust fault detection filter using LMI tools. Comput. Math. Appl. **57**(11–12), 1743–1747 (2009)
11. Hou, M., Patton, R.J.: An LMI approach to H_-/H_∞ fault detection observers. In: Proceedings of UKACC International Conference on Control (CONTROL'96), pp. 305–310. Exeter, UK (1996)

12. Ichalal, D., Marx, B., Ragot, J., Maquin, D.: Fault detection, isolation and estimation for Takagi-Sugeno nonlinear systems. J. Frankl. Inst. **351**(7), 3651–3676 (2014)
13. Krokavec, D., Filasová, A.: Optimal fuzzy control for a class of nonlinear systems. Math. Probl. Eng. **481942**, 1–29 (2012)
14. Krokavec, D., Filasová, A.: LMI based fuzzy observer design for Takagi-Sugeno models containing vestigial nonlinear terms. Arch. Control Sci. **24**(1), 39–52 (2014)
15. Nguang, S.K., Shi, P., Ding, S.: Fault detection filter for uncertain fuzzy systems: an LMI approach. In: Proceedings of 16th IFAC World Congress, pp. 1839–1839. Prag, Czech Republic (2005)
16. Nguang, S.K., Shi, P., Ding, S.: Fault detection for uncertain fuzzy systems: an LMI approach. IEEE Trans. Fuzzy Syst. **15**(6), 1251–1262 (2007)
17. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and Control. IEEE Trans. Syste. Man Cybern. **15**(1), 116–132 (1985)

Robust UIO Design for an Actuator Fault Identification

Piotr Witczak and Marcin Mrugalski

Abstract In this paper an actuator robust fault identification scheme is developed, which is based on an observer within $\mathcal{H}_\infty$ framework for a class of non-linear systems. The proposed approach is designed in such a way that a prescribed disturbance attenuation level is achieved with respect to the actuator fault estimation error while guaranteeing the convergence of the observer. The effectiveness of the proposed approach is verified with the laboratory multi-tank system.

Keywords Fault estimation · Fault identification · Robust estimation · Non-linear systems · Observers · LPV systems

1 Introduction

During last three decades several efficient Fault Detection and Isolation (FDI) methods for non-linear dynamic systems were developed [6, 10–12]. Such methods allows to reduce the economical losses resulting from the industrial systems malfunction. However, in the last decade the expectations for the industrial systems and fault diagnosis started to change. It was expected that the systems can be operated efficiently despite of existing faults. Such an assumption caused that the scientists focuses on developing the Fault Tolerant Control (FTC) strategies [2, 9, 13, 14, 17, 18].

To achieve this goal the efficient fault estimation methods of the actuators faults which could be used during control process should be elaborated. Such methods allows for the application of the active FTC strategies enabling compensation of the faulty actuator by increasing performance of the other actuator existing in the control system. The problem of the actuators fault estimation can be perceived as the task

P. Witczak (✉) · M. Mrugalski
Institute of Control and Computation Engineering, University of Zielona Góra,
ul. Podgórna 50, 65–246 Zielona Góra, Poland
e-mail: P.Witczak@issi.uz.zgora.pl

M. Mrugalski
e-mail: M.Mrugalski@issi.uz.zgora.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_3

of estimation of the system unknown inputs and can be solved by the application of the Unknown Input Observer (UIO) [3, 7, 14–16]. Such techniques enable for the state and unknown inputs reconstruction on the basis of mathematical model of the system and measurements from the system inputs and outputs.

In this paper a novel observer synthesis procedure, which is based on the concept of the UIO for the actuators fault detection and estimation, is proposed. The developed approach is a combination of the linear-system strategies [4] for a class of non-linear systems [20]. The UIO is designed in such a way that a prescribed disturbance attenuation level is achieved with respect to the actuator fault estimation error while guaranteeing the convergence of the observer. The resulting design procedures boil down to solving a set of linear matrix inequalities.

The paper is organized as follows. Section 2 describes the design procedure of the robust UIO using $\mathcal{H}_\infty$ framework for the actuator fault identification. Section 3 provides a introduction into the structure and parameters of a multi-tank system and contains an illustrative example, which shows the performance of the proposed approach for an actuators fault detection and estimation. The final part of the paper is devoted to conclusions.

2 Methodology of the Robust UIO Design for the Actuators Faults Estimation

The main objective of this section is to provide a detailed design procedure of the robust observer, which can be used for the robust actuator fault diagnosis. As a result the estimate of the actuator fault is obtained. In order to achieve this goal the observer should be designed in such a way that a prescribed disturbance attenuation level is achieved with respect to the actuator fault estimation error while guaranteeing the convergence of the observer.

A dynamic, non-linear system can be represent by the LPV model in a relatively simple way. To design such a model, it is necessary to linearize a non-linear system around a number of operating points. The number of points determines the accuracy of the LPV model. The local system behavior around the operating point is represented by each of these linear models. Let us consider the following discrete-time non-linear model:

$$\boldsymbol{x}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k) \tag{1}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k \tag{2}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{y} \in \mathbb{R}^p$ is the output, $\boldsymbol{u} \in \mathbb{R}^m$ is the input vector and $\boldsymbol{h}(\cdot)$ is a non-linear function. Such model can be represented in the form of a discrete-time polytopic LPV model:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}(h_k)\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \tag{3}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k \tag{4}$$

where $\boldsymbol{A}(h_k)$, $\boldsymbol{B}$, $\boldsymbol{C}$ are state-space matrices and $h_k \in \mathbb{R}^l$ is a time-varying parameter vector which ranges over a fixed polytope. The dependence of $\boldsymbol{A}$ on h_k represents a general discrete-time quasi-LPV model. The model (3)–(4) can be written in the following alternative form of the state-space model:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{L}_a\boldsymbol{f}_{a,k} + \boldsymbol{W}_1\boldsymbol{w}_k \tag{5}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{C}\boldsymbol{x}_{k+1} + \boldsymbol{W}_2\boldsymbol{w}_{k+1} \tag{6}$$

where $\boldsymbol{x}_k \in \mathbb{X} \subset \mathbb{R}^n$ is the state vector, $\boldsymbol{u}_k \in \mathbb{R}^r$ stands for the input, $\boldsymbol{y}_k \in \mathbb{R}^m$ denotes the output, $\boldsymbol{f}_{a,k} \in \mathbb{R}^r$ stands for the actuator and $\boldsymbol{L}_a$ is its distribution matrix. Moreover, $\boldsymbol{w}_k \in l_2$ is an exogenous disturbance vector with $\boldsymbol{W}_1 \in \mathbb{R}^{n\times n}$, $\boldsymbol{W}_2 \in \mathbb{R}^{m\times n}$ being its distribution matrices while:

$$l_2 = \left\{ \mathbf{w} \in \mathbb{R}^n |\ \|\mathbf{w}\|_{l_2} < +\infty \right\}, \quad \|\mathbf{w}\|_{l_2} = \left(\sum_{k=0}^{\infty} \|\boldsymbol{w}_k\|^2 \right)^{\frac{1}{2}} \tag{7}$$

As the description of the LPV model is delivered then the robust UIO design procedure can be developed. Following [4], let us assume that the system is observable and the following rank condition is satisfied:

$$\operatorname{rank}(\boldsymbol{C}\boldsymbol{L}_a) = \operatorname{rank}(\boldsymbol{L}_a) = s \tag{8}$$

Under the assumption (8) it is possible to obtain:

$$\boldsymbol{H} = (\boldsymbol{C}\boldsymbol{L}_a)^+ = \left[(\boldsymbol{C}\boldsymbol{L}_a)^T\boldsymbol{C}\boldsymbol{L}_a\right]^{-1}(\boldsymbol{C}\boldsymbol{L}_a)^T \tag{9}$$

It should be underlined, that the proposed approach is designed for actuator faults estimation only, and hence, the setting $\boldsymbol{f}_{s,k} = \boldsymbol{0}$ is employed in the sequel. Substituting $\boldsymbol{f}_{s,k} = \boldsymbol{0}$ into (6) and multiplying it by matrix $\boldsymbol{H}$, and then substituting (5), it can be shown that:

$$\boldsymbol{f}_{a,k} = \boldsymbol{H}(\boldsymbol{y}_{k+1} - \boldsymbol{C}\boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}_k - \boldsymbol{C}\boldsymbol{g}\left(\boldsymbol{x}_k\right) - \boldsymbol{C}\boldsymbol{W}_1\boldsymbol{w}_k - \boldsymbol{W}_2\boldsymbol{w}_{k+1}) \tag{10}$$

Finally, by substituting (10) into (5) it can be shown that:

$$\boldsymbol{x}_{k+1} = \bar{\boldsymbol{A}}\boldsymbol{x}_k + \bar{\boldsymbol{B}}\boldsymbol{u}_k + \boldsymbol{G}\boldsymbol{g}\left(\boldsymbol{x}_k\right) + \bar{\boldsymbol{L}}\boldsymbol{y}_{k+1} + \boldsymbol{G}\boldsymbol{W}_1\boldsymbol{w}_k - \bar{\boldsymbol{L}}\boldsymbol{W}_2\boldsymbol{w}_{k+1} \tag{11}$$

where $\boldsymbol{G} = (\boldsymbol{I}_n - \boldsymbol{L}_a\boldsymbol{H}\boldsymbol{C})$, $\bar{\boldsymbol{A}} = \boldsymbol{G}\boldsymbol{A}$, $\bar{\boldsymbol{B}} = \boldsymbol{G}\boldsymbol{B}$, $\bar{\boldsymbol{L}} = \boldsymbol{L}_a\boldsymbol{H}$.

The estimation of the system state $\hat{\boldsymbol{x}}_k$ with the corresponding observer:

$$\hat{\boldsymbol{x}}_{k+1} = \bar{\boldsymbol{A}}\hat{\boldsymbol{x}}_k + \bar{\boldsymbol{B}}\boldsymbol{u}_k + \boldsymbol{G}\boldsymbol{g}\left(\hat{\boldsymbol{x}}_k\right) + \bar{\boldsymbol{L}}\boldsymbol{y}_{k+1} + \boldsymbol{K}_a(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k) \tag{12}$$

allows to obtain the estimate of the actuators faults:

$$\hat{\boldsymbol{f}}_{a,k} = \boldsymbol{H}(\boldsymbol{y}_{k+1} - \boldsymbol{CA}\hat{\boldsymbol{x}}_k - \boldsymbol{CBu}_k - \boldsymbol{Cg}\left(\hat{\boldsymbol{x}}_k\right)) \tag{13}$$

It should be underlined that the state estimation error is defined as:

$$\begin{aligned}\boldsymbol{e}_{k+1} &= \left(\bar{\boldsymbol{A}} - \boldsymbol{K}_a\boldsymbol{C}\right)\boldsymbol{e}_k + \boldsymbol{Gs}_k + (\boldsymbol{GW}_1 - \boldsymbol{K}_a\boldsymbol{W}_2)\boldsymbol{w}_k - \bar{\boldsymbol{L}}\boldsymbol{W}_2\boldsymbol{w}_{k+1} = \\ &= \boldsymbol{A}_1\boldsymbol{e}_k + \boldsymbol{Gs}_k + \bar{\boldsymbol{W}}_1\boldsymbol{w}_k + \bar{\boldsymbol{W}}_2\boldsymbol{w}_{k+1}\end{aligned} \tag{14}$$

where

$$\boldsymbol{s}_k = \boldsymbol{g}\left(\boldsymbol{x}_k\right) - \boldsymbol{g}\left(\hat{\boldsymbol{x}}_k\right) \tag{15}$$

Similarly, the fault estimation error $\boldsymbol{\varepsilon}_{f_a,k}$ can be calculated according to the following equation:

$$\boldsymbol{\varepsilon}_{f_a,k} = \boldsymbol{f}_{a,k} - \hat{\boldsymbol{f}}_{a,k} = -\boldsymbol{HC}\left(\boldsymbol{Ae}_k + \boldsymbol{s}_k + \boldsymbol{W}_1\boldsymbol{w}_k\right) - \boldsymbol{HW}_2\boldsymbol{w}_{k+1} \tag{16}$$

Note that both $\boldsymbol{e}_k$ and $\boldsymbol{\varepsilon}_{f_a,k}$ are non-linear with respect to $\boldsymbol{e}_k$. To solve this problem, the undermentioned solution is proposed. Using the Differential Mean Value Theorem (DMVT) [19], it can be shown that:

$$\boldsymbol{g}\left(\boldsymbol{a}\right) - \boldsymbol{g}\left(\boldsymbol{b}\right) = \boldsymbol{M}_x(\boldsymbol{a} - \boldsymbol{b}) \tag{17}$$

with

$$\boldsymbol{M}_x = \begin{bmatrix} \dfrac{\partial g_1}{\partial x}(\boldsymbol{c_1}) \\ \vdots \\ \dfrac{\partial g_n}{\partial x}(\boldsymbol{c_n}) \end{bmatrix} \tag{18}$$

where $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n \in \mathrm{Co}(\boldsymbol{a}, \boldsymbol{b})$, $\boldsymbol{c}_i \neq \boldsymbol{a}$ and $\boldsymbol{c}_i \neq \boldsymbol{b}$, $i = 1, \ldots, n$. Assumed that:

$$\bar{g}_{i,j} \geq \frac{\partial g_i}{\partial x_j} \geq \underline{g}_{i,j}, \quad i = 1, \ldots, n, \quad j = 1, \ldots, n \tag{19}$$

a gradient can be calculated as follows:

$$\frac{\partial g_i(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k} = \left[\frac{\partial g_i(x)}{\partial x_1}, \ldots, \frac{\partial g_i(x)}{\partial x_n}\right]^T \tag{20}$$

Now, it can be shown that:

$$\mathbb{M}_x = \left\{\boldsymbol{M} \in \mathbb{R}^{n\times n} | \bar{g}_{i,j} \geq m_{x,i,j} \geq \underline{g}_{i,j},\ i,j = 1, \ldots, n\right\} \tag{21}$$

Using (17) and assuming that $\boldsymbol{M}_{x,k} \in \mathbb{M}_x$ the term $\boldsymbol{A}_1\boldsymbol{e}_k + \boldsymbol{G}\boldsymbol{s}_k$ in (14) can be written as:

$$\boldsymbol{A}_1\boldsymbol{e}_k + \boldsymbol{G}\boldsymbol{s}_k = (\bar{\boldsymbol{A}} + \boldsymbol{G}\boldsymbol{M}_{x,k} - \boldsymbol{K}_a\boldsymbol{C})\boldsymbol{e}_k \tag{22}$$

From (22), it can be deduced that the $\boldsymbol{e}_{k+1}$ described by (14) can be converted into following form:

$$\begin{aligned} \boldsymbol{e}_{k+1} &= \boldsymbol{A}_2(h_k)\boldsymbol{e}_k + \bar{\boldsymbol{W}}_1\boldsymbol{w}_k + \bar{\boldsymbol{W}}_2\boldsymbol{w}_{k+1} \\ \boldsymbol{A}_2(h_k) &= \tilde{\boldsymbol{A}}(h_k) - \boldsymbol{K}_a\boldsymbol{C} \end{aligned} \tag{23}$$

defining an LPV polytopic system [1] with:

$$\tilde{\mathbb{A}} = \left\{ \tilde{\boldsymbol{A}}(h_k) : \quad \tilde{\boldsymbol{A}}(h_k) = \sum_{i=1}^{N} h_{ki}\tilde{\boldsymbol{A}}_i, \ \sum_{i=1}^{N} h_{ki} = 1, \ h_{ki} \geq 0 \right\} \tag{24}$$

where $N = 2^{n^2}$. Note that this is a general description, which does not take into account that some elements of $\boldsymbol{M}_{x,k}$ may be constant. In such cases, N is given by $N = 2^{(n-c)^2}$ where c stands for the number of constant elements of $\boldsymbol{M}_{x,k}$. Similarly, the fault estimation error $\boldsymbol{\varepsilon}_{f_a,k}$ can be converted into:

$$\boldsymbol{\varepsilon}_{f_a,k} = -\boldsymbol{H}\boldsymbol{C}\left(\boldsymbol{A}_3(h_k)\boldsymbol{e}_k + \boldsymbol{W}_1\boldsymbol{w}_k\right) - \boldsymbol{H}\boldsymbol{W}_2\boldsymbol{w}_{k+1} \tag{25}$$

$$\mathbb{A}_3 = \left\{ \boldsymbol{A}_3(h_k) : \quad \boldsymbol{A}_3(h_k) = \sum_{i=1}^{N} h_{ki}\boldsymbol{A}_{3,i}, \ \sum_{i=1}^{N} h_{ki} = 1, \ h_{ki} \geq 0 \right\} \tag{26}$$

The objective of further deliberations is to design the observer (12) in such a way that the state estimation error $\boldsymbol{e}_k$ is asymptotically convergent and the following upper bound is guaranteed:

$$\|\boldsymbol{\varepsilon}_f\|_{l_2} \leq \omega\|\mathbf{w}\|_{l_2} \tag{27}$$

where $\omega > 0$ is a prescribed disturbance attenuation level. Thus, on the contrary to the approaches presented in the literature, ω should be achieved with respect to the fault estimation error but not the state estimation error.

The problem of $\mathcal{H}_\infty$ observer design [8, 20] is to determine the gain matrix $\boldsymbol{K}_a$ such that:

$$\lim_{k\to\infty} \boldsymbol{e}_k = \mathbf{0} \quad \text{for } \boldsymbol{w}_k = \mathbf{0} \tag{28}$$

$$\|\boldsymbol{\varepsilon}_f\|_{l_2} \leq \omega\|\mathbf{w}\|_{l_2} \quad \text{for } \boldsymbol{w}_k \neq \mathbf{0}, \ \boldsymbol{e}_0 = \mathbf{0} \tag{29}$$

To solve the above problem it is necessary to find a Lyapunov function V_k:

$$\Delta V_k + \boldsymbol{\varepsilon}_{f_a,k}^T \boldsymbol{\varepsilon}_{f_a,k} - \mu^2 \boldsymbol{w}_k^T \boldsymbol{w}_k - \mu^2 \boldsymbol{w}_{k+1}^T \boldsymbol{w}_{k+1} < 0,\ k = 0, \dots \infty \tag{30}$$

where $\Delta V_k = V_{k+1} - V_k$, $\mu > 0$. Note that the structure of (30) is uncommon in the literature. Indeed, the novelty is that the term $-\mu^2 \boldsymbol{w}_{k+1}^T \boldsymbol{w}_{k+1}$ is introduced. This is caused the fault decoupling procedure (cf.(10)). Indeed, if $\boldsymbol{w}_k = \boldsymbol{0}$, $(k = 0, \dots, \infty)$ then (30) boils down to:

$$\Delta V_k + \boldsymbol{\varepsilon}_{f_a,k}^T \boldsymbol{\varepsilon}_{f_a,k} < 0,\ k = 0, \dots \infty \tag{31}$$

and hence $\Delta V_k < 0$, which leads to (28). If $\boldsymbol{w}_k \neq \boldsymbol{0}$ for $k = 0, \dots, \infty$ then the Lyapunov function (30) yields:

$$J = \sum_{k=0}^{\infty} \left(\Delta V_k + \boldsymbol{\varepsilon}_{f_a,k}^T \boldsymbol{\varepsilon}_{f_a,k} - \mu^2 \boldsymbol{w}_k^T \boldsymbol{w}_k - \mu^2 \boldsymbol{w}_{k+1}^T \boldsymbol{w}_{k+1} \right) < 0 \tag{32}$$

and can be written as:

$$J = -V_0 + \sum_{k=0}^{\infty} \boldsymbol{\varepsilon}_{f_a,k}^T \boldsymbol{\varepsilon}_{f_a,k} - \mu^2 \sum_{k=0}^{\infty} \boldsymbol{w}_k^T \boldsymbol{w}_k - \mu^2 \sum_{k=0}^{\infty} \boldsymbol{w}_{k+1}^T \boldsymbol{w}_{k+1} < 0 \tag{33}$$

Bearing in mind that:

$$\mu^2 \sum_{k=0}^{\infty} \boldsymbol{w}_{k+1}^T \boldsymbol{w}_{k+1} = \mu^2 \sum_{k=0}^{\infty} \boldsymbol{w}_k^T \boldsymbol{w}_k - \mu^2 \boldsymbol{w}_0^T \boldsymbol{w}_0 \tag{34}$$

inequality (33) can be written as:

$$J = -V_0 + \sum_{k=0}^{\infty} \boldsymbol{\varepsilon}_{f_a,k}^T \boldsymbol{\varepsilon}_{f_a,k} - 2\mu^2 \sum_{k=0}^{\infty} \boldsymbol{w}_k^T \boldsymbol{w}_k + \mu^2 \boldsymbol{w}_0^T \boldsymbol{w}_0 < 0 \tag{35}$$

Knowing that $V_0 = 0$ for $\boldsymbol{e}_0 = 0$, (35) leads to (29) with $\omega = \sqrt{2}\mu$.

As the general scheme for designing the robust observer is proposed, the following form of the Lyapunov function is assumed [19]:

$$V_k = \boldsymbol{e}_k^T \boldsymbol{P}(h_k) \boldsymbol{e}_k \tag{36}$$

where $\boldsymbol{P}(h_k) \succ \boldsymbol{0}$. On the contrary to the design approach presented in the literature (see, e.g. [20]) it is not assumed that $\boldsymbol{P}(h_k) = \boldsymbol{P}$ is constant. Indeed, $\boldsymbol{P}(h_k)$ can be perceived as a parameter-depended matrix [1] of the following form:

$$\boldsymbol{P}(h_k) = \sum_{i=1}^{N} h_{ki}\boldsymbol{P}_i \tag{37}$$

As a consequence

$$\begin{aligned}
&\Delta V_k + \boldsymbol{\varepsilon}_{f_a,k}^T\boldsymbol{\varepsilon}_{f_a,k} - \mu^2\boldsymbol{w}_k^T\boldsymbol{w}_k - \mu^2\boldsymbol{w}_{k+1}^T\boldsymbol{w}_{k+1} \\
&= \boldsymbol{e}_k^T\left(\boldsymbol{A}_2(h_k)^T\boldsymbol{P}(h_{k+1})\boldsymbol{A}_2(h_k) + \boldsymbol{A}_3(h_k)^T\boldsymbol{H}_1\boldsymbol{A}_3(h_k) - \boldsymbol{P}(h_k)\right)\boldsymbol{e}_k \\
&+ \boldsymbol{e}_k^T\left(\boldsymbol{A}_2(h_k)^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_1 + \boldsymbol{A}_3(h_k)^T\boldsymbol{H}_1\boldsymbol{W}_1\right)\boldsymbol{w}_k \\
&+ \boldsymbol{e}_k^T\left(\boldsymbol{A}_2(h_k)^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_2 + \boldsymbol{A}_3(h_k)^T\boldsymbol{H}_2\right)\boldsymbol{w}_{k+1} \\
&+ \boldsymbol{w}_k^T\left(\bar{\boldsymbol{W}}_1^T\boldsymbol{P}(h_{k+1})\boldsymbol{A}_2(h_k) + \boldsymbol{W}_1^T\boldsymbol{H}_1\boldsymbol{A}_3(h_k)\right)\boldsymbol{e}_k \\
&+ \boldsymbol{w}_k^T\left(\bar{\boldsymbol{W}}_1^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_1 + \boldsymbol{W}_1^T\boldsymbol{H}_1\boldsymbol{W}_1 - \mu^2\boldsymbol{I}\right)\boldsymbol{w}_k \\
&+ \boldsymbol{w}_k^T\left(\bar{\boldsymbol{W}}_1^T\boldsymbol{P}(h_{k+1})\boldsymbol{W}_2 + \boldsymbol{W}_1^T\boldsymbol{H}_2\right)\boldsymbol{w}_{k+1} \\
&+ \boldsymbol{w}_{k+1}^T\left(\bar{\boldsymbol{W}}_2^T\boldsymbol{P}(h_{k+1})\boldsymbol{A}_{2,k} + \boldsymbol{H}_2^T\boldsymbol{A}_3(h_k)\right)\boldsymbol{e}_k \\
&+ \boldsymbol{w}_{k+1}^T\left(\bar{\boldsymbol{W}}_2^T\boldsymbol{P}(h_{k+1})\boldsymbol{W}_1 + \boldsymbol{H}_2^T\boldsymbol{W}_1\right)\boldsymbol{w}_k \\
&+ \boldsymbol{w}_{k+1}^T\left(\bar{\boldsymbol{W}}_2^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_2 + \boldsymbol{W}_2^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{W}_2 - \mu^2\boldsymbol{I}\right)\boldsymbol{w}_{k+1} < 0
\end{aligned} \tag{38}$$

where $\Delta V_k = V_{k+1} - V_k$, $\boldsymbol{H}_1 = \boldsymbol{C}^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{C}$ and $\boldsymbol{H}_2 = \boldsymbol{C}^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{W}_2$. Defining $\boldsymbol{v}_k = \left[\boldsymbol{e}_k^T, \boldsymbol{w}_k^T, \boldsymbol{w}_{k+1}^T\right]^T$, inequality (38) takes the following form:

$$\Delta V_k + \boldsymbol{\varepsilon}_{f_a,k}^T\boldsymbol{\varepsilon}_{f_a,k} - \mu^2\boldsymbol{w}_k^T\boldsymbol{w}_k - \mu^2\boldsymbol{w}_{k+1}^T\boldsymbol{w}_{k+1} = \boldsymbol{v}_k^T\boldsymbol{M}_V\boldsymbol{v}_k < 0 \tag{39}$$

where $\boldsymbol{M}_V$ is given by:

$$\boldsymbol{M}_V = \begin{bmatrix}
\boldsymbol{A}_2(h_k)^T\boldsymbol{P}(h_{k+1})\boldsymbol{A}_2(h_k) + \boldsymbol{A}_3(h_k)^T\boldsymbol{H}_1\boldsymbol{A}_3(h_k) - \boldsymbol{P}(h_k) \\
\bar{\boldsymbol{W}}_1^T\boldsymbol{P}(h_{k+1})\boldsymbol{A}_2(h_k) + \boldsymbol{W}_1^T\boldsymbol{H}_1\boldsymbol{A}_3(h_k) \\
\bar{\boldsymbol{W}}_2^T\boldsymbol{P}(h_{k+1})\boldsymbol{A}_2(h_k) + \boldsymbol{H}_2^T\boldsymbol{A}_3(h_k) \\
\boldsymbol{A}_2(h_k)^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_1 + \boldsymbol{A}_3(h_k)^T\boldsymbol{H}_1\boldsymbol{W}_1 \\
\bar{\boldsymbol{W}}_1^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_1 + \boldsymbol{W}_1^T\boldsymbol{H}_1\boldsymbol{W}_1 - \mu^2\boldsymbol{I} \\
\bar{\boldsymbol{W}}_2^T\boldsymbol{P}(h_{k+1})\boldsymbol{W}_1 + \boldsymbol{H}_2^T\boldsymbol{W}_1 \\
\boldsymbol{A}_2(h_k)^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_2 + \boldsymbol{A}_3(h_k)^T\boldsymbol{H}_2 \\
\bar{\boldsymbol{W}}_1^T\boldsymbol{P}(h_{k+1})\boldsymbol{W}_2 + \boldsymbol{W}_1^T\boldsymbol{H}_2 \\
\bar{\boldsymbol{W}}_2^T\boldsymbol{P}(h_{k+1})\bar{\boldsymbol{W}}_2 + \boldsymbol{W}_2^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{W}_2 - \mu^2\boldsymbol{I}
\end{bmatrix} \tag{40}$$

The above results can be written in the form of theorem describing the developed framework of observer design:

Theorem 1 *For a prescribed disturbance attenuation level* $\mu > 0$ *for the fault estimation error* (16)*, the* $\mathcal{H}_\infty$ *observer design problem for the system* (5)–(6) *and the observer* (12) *is solvable if there exists matrices* $\boldsymbol{P}_i \succ \boldsymbol{0}$ $(i = 1, \ldots, N$ *and* $j = 1, \ldots, N)$*,* $\boldsymbol{U}$ *and* $\boldsymbol{N}$ *such that the following LMIs are satisfied:*

$$\begin{bmatrix} \boldsymbol{A}_{3,i}^T \boldsymbol{H}_1 \boldsymbol{A}_{3,j} - \boldsymbol{P}_i & \boldsymbol{A}_{3,i}^T \boldsymbol{H}_1 \boldsymbol{W}_1 & \boldsymbol{A}_{3,i}^T \boldsymbol{H}_3 & \boldsymbol{A}_{2,i} \boldsymbol{U}^T \\ \boldsymbol{W}_1^T \boldsymbol{H}_1 \boldsymbol{A}_{3,i} & \boldsymbol{W}_1^T \boldsymbol{H}_1 \boldsymbol{W}_1 - \mu^2 \boldsymbol{I} & \boldsymbol{W}_1^T \boldsymbol{H}_2 & \bar{\boldsymbol{W}}_1^T \boldsymbol{U}^T \\ \boldsymbol{H}_2^T \boldsymbol{A}_{3,i} & \boldsymbol{H}_2^T \boldsymbol{W}_1 & \boldsymbol{W}_2^T \boldsymbol{H}^T \boldsymbol{H} \boldsymbol{W}_2 - \mu^2 \boldsymbol{I} & \bar{\boldsymbol{W}}_2^T \boldsymbol{U}^T \\ \boldsymbol{U} \boldsymbol{A}_{2,i} & \boldsymbol{U} \bar{\boldsymbol{W}}_1 & \boldsymbol{U} \bar{\boldsymbol{W}}_2 & \boldsymbol{P}_j - \boldsymbol{U} - \boldsymbol{U}^T \end{bmatrix} \prec \boldsymbol{0} \tag{41}$$

for $i = 1, \ldots, N$ *and* $j = 1, \ldots, N$ *where:*

$$\boldsymbol{U} \boldsymbol{A}_{2,i} = \boldsymbol{U}(\tilde{\boldsymbol{A}}_i - \boldsymbol{K}_a \boldsymbol{C}) = \boldsymbol{U} \tilde{\boldsymbol{A}}_i - \boldsymbol{N} \boldsymbol{C}, \tag{42}$$

$$\boldsymbol{U} \bar{\boldsymbol{W}}_1 = \boldsymbol{U}(\boldsymbol{G} \boldsymbol{W}_1 - \boldsymbol{K}_a \boldsymbol{W}_2) = \boldsymbol{U} \boldsymbol{G} \boldsymbol{W}_1 - \boldsymbol{N} \boldsymbol{W}_2 \tag{43}$$

Proof The following two lemmas can be perceived as the generalization of those presented in [1].

Lemma 1 *The following statements are equivalent*

1. *There exists* $\boldsymbol{X} \succ 0$ *such that:*

$$\boldsymbol{V}^T \boldsymbol{X} \boldsymbol{V} - \boldsymbol{W} \prec 0 \tag{44}$$

2. *There exists* $\boldsymbol{X} \succ 0$ *such that:*

$$\begin{bmatrix} -\boldsymbol{W} & \boldsymbol{V}^T \boldsymbol{U}^T \\ \boldsymbol{U} \boldsymbol{V} & \boldsymbol{X} - \boldsymbol{U} - \boldsymbol{U}^T \end{bmatrix} \prec 0 \tag{45}$$

Subsequently, observing that the matrix (40) must be negative definite and writing it as:

$$\begin{bmatrix} \boldsymbol{A}_2(h_k)^T \\ \bar{\boldsymbol{W}}_1^T \\ \bar{\boldsymbol{W}}_2^T \end{bmatrix} \boldsymbol{P}(h_{k+1}) \begin{bmatrix} \boldsymbol{A}_2(h_k) & \bar{\boldsymbol{W}}_1 & \bar{\boldsymbol{W}}_2 \end{bmatrix} \tag{46}$$

$$+ \begin{bmatrix} \boldsymbol{A}_3(h_k)^T \boldsymbol{H}_1 \boldsymbol{A}_3(h_k) - \boldsymbol{P}(h_k) & \boldsymbol{A}_3(h_k)^T \boldsymbol{H}_1 \boldsymbol{W}_1 & \boldsymbol{A}_3(h_k)^T \boldsymbol{H}_3 \\ \boldsymbol{W}_1^T \boldsymbol{H}_1 \boldsymbol{A}_3(h_k) & \boldsymbol{W}_1^T \boldsymbol{H}_1 \boldsymbol{W}_1 - \mu^2 \boldsymbol{I} & \boldsymbol{W}_1^T \boldsymbol{H}_2 \\ \boldsymbol{H}_2^T \boldsymbol{A}_3(h_k) & \boldsymbol{H}_2^T \boldsymbol{W}_1 & \boldsymbol{W}_2^T \boldsymbol{H}^T \boldsymbol{H} \boldsymbol{W}_2 - \mu^2 \boldsymbol{I} \end{bmatrix} \prec 0 \tag{47}$$

and then applying Lemma 1 leads to (41), which completes the proof.

Finally, the design procedure boils down to solving LMIs (41) and then (cf. (42)–(43)) $\boldsymbol{K}_a = \boldsymbol{U}^{-1}\boldsymbol{N}$. It can be also observed that the noticed design problem can be treated as minimization task, i.e.

$$\mu^* = \min_{\mu>0,\boldsymbol{P}_1>\boldsymbol{0},\boldsymbol{U},\boldsymbol{N}} \mu \tag{48}$$

under (41).

3 Fault Identification of the Multi-tank System

The objective of this section is to provide the reliable experimental results for proposed approach in the actuator fault estimation simultaneously with the state estimation. The multi-tank system [5] presented in Fig. 1, designed to reflect the behaviour and dynamics of full scale multi-tank industrial systems (e.g. hydroelectric power-plants, reservoir, etc.) was used in the laboratory conditions to show proposed approach at work. The multi-tank system consists of three separate tanks placed in series one under another, equipped with drain valves and level sensors based on a hydraulic pressure measurement. Each of them has a different cross-section in order to reflect system nonlinearities. The bottom tank is a water reservoir for the system. A variable speed water pump is used to fill the top tank. Due to gravity, the water flows through valves and tanks. The considered multi-tank system has been designed to operate with an external, PC-based digital controller. The control computer communicates with the level sensors, valves and a pump by a dedicated I/O board with the power interface. The I/O interface is controlled by the real-time software, which can operate in Matlab/Simulink environment.

Fig. 1 Multi-tank system

The distribution matrices $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ should express the influence and magnitude of disturbances $\boldsymbol{w}_k$ onto the state and output equations (5)–(6), respectively. To obtain appropriate ratio between the elements of $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$, series of constant liquid level measurements was performed for the top tank. Next, the mean was removed, which represents the constant level of the liquid. Subsequently the disturbances where analyzed. The standard deviation of the disturbance is equal $1.75 \cdot 10^{-4}$ (obtained for 1000 measurements). Similar results were obtained for the sensors in the middle and bottom tanks. The term $\boldsymbol{W}_1\boldsymbol{w}_k$ (cf. (5)) will represent the inaccuracy of the pump with respect to a desired control action. After a similar experiments like for the sensors, it was derived that the maximum magnitude of $\boldsymbol{W}_1\boldsymbol{w}_k$ is approximately five (5) times larger than that of $\boldsymbol{W}_2\boldsymbol{w}_k$. Thus resulting in the following values of the distribution matrices:

$$\boldsymbol{W}_1 = \mathrm{diag}(0.05, 0, 0), \quad \boldsymbol{W}_2 = 0.01\boldsymbol{I}_m \tag{49}$$

Subsequently the UIO design procedure was performed. As a result of solving the problem (41), the following couple was obtained:

$$\mu = 0.55; \quad \boldsymbol{K}_a = \begin{bmatrix} 0.1089 & 0 \\ 0.0004 & 1.7107 \\ 0 & 0.9473 \end{bmatrix} \tag{50}$$

Next, to proceed to the experiment itself, the following initial conditions for the system $\boldsymbol{x}_0 = [0.001, 0.001, 0.001]^T$ and the observer $\hat{\boldsymbol{x}}_0 = [0.2, 0.1, 0.1]^T$ were assumed, while the input is $\boldsymbol{u}_k = 0.009$. Moreover, following actuator fault scenario was used:

$$\boldsymbol{f}_{a,k} = \begin{cases} -0.001, & \text{for } 30000 \leq k \leq 50000 \\ 0, & \text{otherwise} \end{cases} \tag{51}$$

Initially the case when $\hat{\boldsymbol{x}}_0 = \boldsymbol{x}_0$ ($\boldsymbol{e}_0 = \boldsymbol{0}$) was considered. Figure 2 clearly indicates that condition (29) is satisfied, which means that attenuation level $\mu = 0.55$ is achieved. Subsequently the case where $\boldsymbol{w}_k = \boldsymbol{0}$ and $\hat{\boldsymbol{x}}_0 \neq \boldsymbol{x}_0$ was taken into consideration. Figure 3 clearly shows that (28) is satisfied as well. Finally, Fig. 4 shows the fault and its estimate for the nominal case ($\hat{\boldsymbol{x}}_0 \neq \boldsymbol{x}_0$ and $\boldsymbol{w}_k \neq \boldsymbol{0}$). At the same time observer estimates unavailable state x_2. This appealing property is depicted in Fig. 5, it is easy to observe that estimate converges to (and then tracks) the state. Thus making example complete.

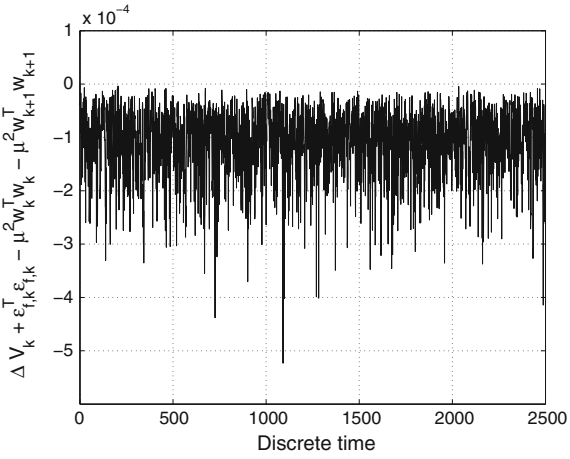


Fig. 2 Evolution of $\Delta V_k + \boldsymbol{\varepsilon}_{f_a,k}^T \boldsymbol{\varepsilon}_{f_a,k} - \mu^2 \boldsymbol{w}_k^T \boldsymbol{w}_k - \mu^2 \boldsymbol{w}_{k+1}^T \boldsymbol{w}_{k+1}$

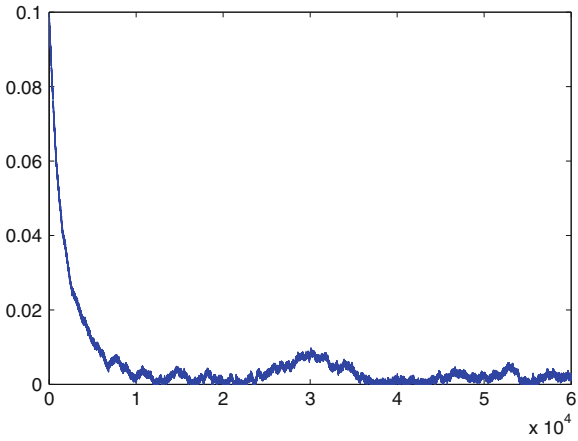


Fig. 3 Evolution of $\|e_k\|$

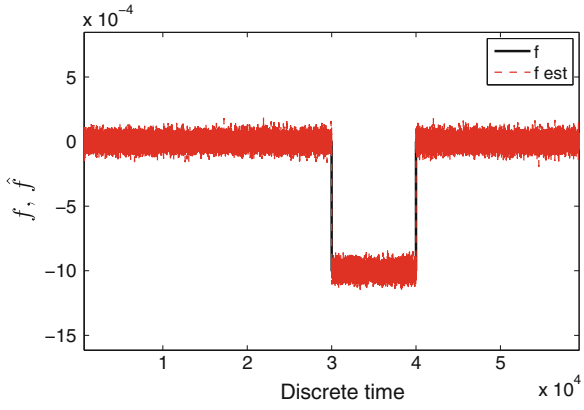


Fig. 4 Actuator fault and its estimate

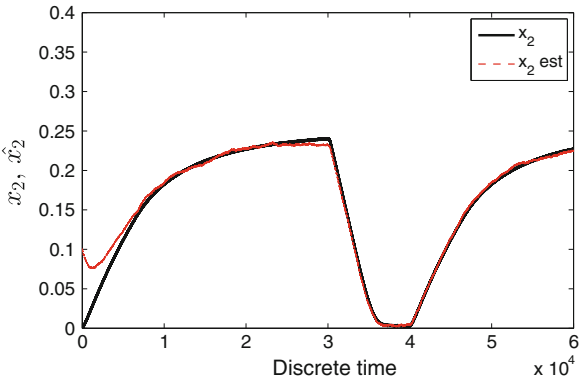


Fig. 5 State of the second tank and its estimate

4 Conclusions

In this paper a novel approach allowing for the robust actuators fault estimation on the basis of the polytopic LPV model and UIO is proposed. In particular, the complete methodology of observer design with the robust $\mathcal{H}_\infty$ approach is proposed to settle the problem of robust fault diagnosis. The proposed UIO is designed in such a way that prescribed disturbance attenuation level is achieved with respect to the actuator fault estimation error while guaranteeing the convergence of the observer. The final part of the paper is concerned with an exhaustive case study regarding the fault estimation of the multi-tank system with the application of the proposed approach. The achieved results clearly show the performance and quality of the proposed method, which confirms its practical usefulness.

Acknowledgments The work was supported by the National Science Center of Poland under grant: 2014–2017.

References

1. de Oliveira, M., Bernussou, J., Geromel, J.: A new discrete-time robust stability condition. Syst. Control Lett. **37**(4), 261–265 (1999)
2. Ducard, G.: Fault-tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles. Springer, Berlin (2009)
3. Frank, P.M., Marcu, T.: Diagnosis strategies and systems: principles, fuzzy and neural approaches. In: Teodorescu, H.N., Mlynek, D., Kandel, A., Zimmermann, H.J. (eds.) Intelligent Systems and Interfaces. Kluwer Academic Publishers, Boston (2000)
4. Gillijns, S., De Moor, B.: Unbiased minimum-variance input and state estimation for linear discrete-time systems. Automatica **43**, 111–116 (2007)
5. INTECO: Multitank System—User's Manual. http://www.inteco.com.pl (2013)
6. Iserman, R.: Fault Diagnosis Applications: Model Based Condition Monitoring, Actuators, Drives, Machinery, Plants, Sensors, and Fault-tolerant Systems. Springer, Berlin (2011)

7. Korbicz, J., Witczak, M., Puig, V.: Lmi-based strategies for designing observers and unknown input observers for non-linear, discrete-time systems. Bull. Polish Acad. Sci.-Tech. Sci. **55**(1), 31–42 (2007)
8. Li, H., Fu, M.: A linear matrix inequality approach to robust h_∞ filtering. IEEE Trans. Signal Proc. **45**(9), 2338–2350 (1997)
9. Mahmoud, M., Jiang, J., Zhang, Y.: Active Fault Tolerant Control Systems: Stochastic Analysis and Synthesis. Springer, Berlin (2003)
10. Mrugalska, B., Akielaszek-Witczak, A., Aubrun, C.: Towards product robust quality control with sequential d-optimum inputs design. Chem. Eng. Trans. **43**, 2137–2142 (2015)
11. Mrugalska, B., Akielaszek-Witczak, A., Stetter, R.: Robust quality control of products with experimental design. In: Popescu, D. (ed.) International Conference on Production Research - Regional Conference Africa, Europe and the Middle East and 3rd International Conference on Quality and Innovation in Engineering and Management, pp. 343–348. Technical University of Cluj-Napoca, Cluj-Napoca, Romania (2014)
12. Mrugalski, M.: Advanced Neural Network-based Computational Schemes for Robust Fault Diagnosis. Springer International Publishing, Heidelberg, Germany (2014)
13. Rotondo, D., Nejjari, P., Puig, V.: Robust quasi-lpv model reference FTC of a quadrotor UAV subject to actuator faults. Int. J. Appl. Math. Comput. Sci. **25**(1), 7–22 (2015)
14. Witczak, M.: Fault Diagnosis and Fault-tolerant Control Strategies for Non-linear Systems: Analytical and Soft Computing Approaches. Springer International Publishing, Heidelberg, Germany (2014)
15. Witczak, M., Mrugalski, M., Korbicz, J.: Towards robust neural-network-based sensor and actuator fault diagnosis: application to a tunnel furnace. Neural Proc. Lett. (2014)
16. Witczak, M., Pretki, P.: Design of an extended unknown input observer with stochastic robustness techniques and evolutionary algorithms. Int. J. Control **80**(5), 749–762 (2007)
17. Witczak, M., Witczak, P.: Efficient predictive fault-tolerant control for non-linear systems. In: Korbicz, J., Kowal, M. (eds.) Intelligent Systems in Technical and Medical Diagnostics. Advances in Intelligent Systems and Computing, vol. 230, pp. 65–76. Lagow Lubuski, Poland (2014)
18. Witczak, P., Luzar, M., Witczak, M., Korbicz, J.: A robust fault-tolerant model predictive control for linear parameter-varying systems. In: 19th International Conference on Methods and Models in Automation and Robotics. pp. 462–467. Miedzyzdroje, Poland (2014)
19. Zemouche, A., Boutayeb, M.: Observer design for Lipschitz non-linear systems: the discrete time case. IEEE Trans. Circ. Syst. II: Exp. Briefs **53**(8), 777–781 (2006)
20. Zemouche, A., Boutayeb, M., Iulia Bara, G.: Observer for a class of Lipschitz systems with extension to h_∞ performance analysis. Syst. Control Lett. **57**(1), 18–27 (2008)

Design of an Adaptive Sensor and Actuator Fault Estimation Scheme with a Quadratic Boundedness Approach

Marcin Witczak, Daniel Zegar and Marcin Pazera

Abstract This paper is concerned with the problem of designing adaptive sensor and actuator fault estimation scheme for linear discrete-time systems. In particular, a suitable system parametrization is introduced in order to transform the simultaneous sensor and actuator fault estimation problem into an actuator estimation one. The scheme is dedicated to the system influenced by disturbances and hence, a quadratic boundedness approach is employed to prove its convergence. The final part of the paper shows an illustrative example concerning fault estimation of a three-tank system.

Keywords Multi-tank system · Fault diagnosis · Actuator fault estimation · Sensor fault estimation

1 Introduction

Fault estimation issue has received a considerable attention during the last decade. This interest is clearly justified, since fault estimation strategies are used with high efficiency to Fault Diagnosis (FD) [2, 8–11, 15] and Fault-Tolerant Control (FTC) [5, 12, 13, 17]. These issues have primary importance in modern process automation and complex industrial systems. In general, faults can be divided into three categories: system dynamics [8, 15], actuator [3] and sensor [4] faults, respectively. This paper aims at designing an adaptive sensor and actuator estimation scheme with quadratic boundedness approaches [1]. This problem can be settled with various approaches. A particular attention deserve, e.g.: minimum variance filter [6], two-stage Kalman filter [7], a robust high-gain sliding mode observers [14], and an $\mathcal{H}_\infty$

M. Witczak (✉) · D. Zegar · M. Pazera
Institute of Control and Computation Engineering,
University of Zielona Góra, ul. Pogórna 50, 65-246 Zielona Gora, Poland
e-mail: m.witczak@issi.uz.zgora.pl

D. Zager
e-mail: d.zegar@issi.uz.zgora.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_4

approach [16, 17]. The above-mentioned procedures are successfully implemented in the FD and FTC. In particular, the fault estimates are used to compensate the fault effect. This clearly emphasizes the need for accurate and robust fault estimation.

To settle this problem within the framework of this paper, an efficient and robust fault estimation scheme is proposed. Moreover, based on the achieved estimates, an adaptive threshold which overbounds the real fault is proposed.

The paper is organized as follows: Sect. 2 presents the general structure of the system and associated fault estimator. Section 3 presents convergence analysis of the proposed approach and its design procedure. In Sect. 4, an adaptive threshold overbounding a real fault is proposed. Finally, an illustrative example is portrayed, which clearly demonstrates the performance of the proposed approach.

2 Fault Estimation Strategy

Let us consider the following linear discrete-time system of the form:

$$\bar{\boldsymbol{x}}_{k+1} = \bar{\boldsymbol{A}}\bar{\boldsymbol{x}}_k + \bar{\boldsymbol{B}}\boldsymbol{u}_k + \bar{\boldsymbol{B}}\boldsymbol{f}_{a,k} + \bar{\boldsymbol{W}}_1\boldsymbol{w}_k \tag{1}$$

$$\bar{\boldsymbol{y}}_k = \bar{\boldsymbol{C}}\bar{\boldsymbol{x}}_k + \bar{\boldsymbol{C}}_f\boldsymbol{f}_{s,k} + \bar{\boldsymbol{W}}_2\boldsymbol{w}_k \tag{2}$$

where $\bar{\boldsymbol{x}}_k \in \mathbb{R}^n$, $\boldsymbol{u}_k \in \mathbb{R}^r$, $\bar{\boldsymbol{y}}_k \in \mathbb{R}^m$, denote the state, input and output, respectively, $\boldsymbol{f}_{a,k}$ is the actuator fault and $\boldsymbol{f}_{s,k}$ is the sensor fault. The $\boldsymbol{w}_k \in \mathbb{R}^n$ is an exogenous disturbance vector and $\bar{\boldsymbol{W}}_1 \in \mathbb{R}^{n\times n}$, $\bar{\boldsymbol{W}}_2 \in \mathbb{R}^{m\times n}$ stand for its distribution matrices.

The objective of further deliberations is to design a scheme that will be able to simultaneously estimate the system state along with the sensor and actuator fault, respectively. To settle this problem, let us consider a filter of the form:

$$\boldsymbol{s}_{k+1} = \boldsymbol{D}(\bar{\boldsymbol{y}}_k - \boldsymbol{s}_k) \tag{3}$$

where $\boldsymbol{D} \in \mathbb{R}^{m\times m}$ is a matrix with eigenvalues placed within a unit circle. Substituting (2) into (3) gives:

$$\boldsymbol{s}_{k+1} = -\boldsymbol{D}\boldsymbol{s}_k + \boldsymbol{D}\bar{\boldsymbol{y}}_k = -\boldsymbol{D}\boldsymbol{s}_k + \boldsymbol{D}\bar{\boldsymbol{C}}\bar{\boldsymbol{x}}_k + \boldsymbol{D}\bar{\boldsymbol{C}}_f\boldsymbol{f}_{s,k} + \boldsymbol{D}\bar{\boldsymbol{W}}_2\boldsymbol{w}_k \tag{4}$$

Thus, an extended state vector can be defines as:

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} \bar{\boldsymbol{x}}_{k+1} \\ \boldsymbol{s}_{k+1} \end{bmatrix} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{L}\boldsymbol{f}_k + \boldsymbol{W}\boldsymbol{w}_k \tag{5}$$

where

$$\boldsymbol{f}_k = \begin{bmatrix} \boldsymbol{f}_{a,k} \\ \boldsymbol{f}_{s,k} \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} \bar{\boldsymbol{A}} & \boldsymbol{0} \\ \boldsymbol{D}\bar{\boldsymbol{C}} & -\boldsymbol{D} \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} \bar{\boldsymbol{B}} \\ \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{L} = \begin{bmatrix} \bar{\boldsymbol{B}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D}\bar{\boldsymbol{C}}_f \end{bmatrix}$$

$$\boldsymbol{W} = \begin{bmatrix} \bar{\boldsymbol{W}}_1 \\ \boldsymbol{D}\bar{\boldsymbol{W}}_2 \end{bmatrix}$$

and the corresponding output equation is

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k \tag{6}$$

with

$$\boldsymbol{C} = [\boldsymbol{0} \quad \boldsymbol{I}]$$

It can be observed that the problem of estimating $\bar{\boldsymbol{x}}_k, \boldsymbol{f}_{a,k}$ and $\boldsymbol{f}_{s,k}$ was reduced to estimating $\boldsymbol{x}_k$ and $\boldsymbol{f}_k$ described by (5) and (6), respectively. Thus, for a given system (5) and (6), an estimator is proposed, which takes the form:

$$\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}\hat{\boldsymbol{x}}_k + \boldsymbol{B}u_k + \boldsymbol{L}\hat{\boldsymbol{f}}_k + \boldsymbol{K}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k) \tag{7}$$

$$\hat{\boldsymbol{f}}_{k+1} = \hat{\boldsymbol{f}}_k + \boldsymbol{F}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k) \tag{8}$$

The objective of the subsequent section is to investigate the convergence of the scheme as well as to provide its design procedure.

Let us start with the state estimation error, which from (5), (6) and (7), (8) is governed by:

$$\boldsymbol{e}_{k+1} = \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} = [\boldsymbol{A} - \boldsymbol{K}\boldsymbol{C}]\boldsymbol{e}_k + \boldsymbol{L}\boldsymbol{e}_{f,k} + \boldsymbol{W}\boldsymbol{w}_k \tag{9}$$

with

$$\boldsymbol{e}_{f,k} = \boldsymbol{f}_k - \hat{\boldsymbol{f}}_k \tag{10}$$

being a fault estimation error, which obeys:

$$\boldsymbol{e}_{f,k+1} = \boldsymbol{f}_{k+1} - \hat{\boldsymbol{f}}_{k+1} = \boldsymbol{f}_{k+1} - \boldsymbol{f}_k + \boldsymbol{f}_k - \hat{\boldsymbol{f}}_k - \boldsymbol{F}\boldsymbol{C}\boldsymbol{e}_k = \boldsymbol{e}_{f,k} - \boldsymbol{F}\boldsymbol{C}\boldsymbol{e}_k + \boldsymbol{\varepsilon}_k \tag{11}$$

with

$$\boldsymbol{\varepsilon}_k = \boldsymbol{f}_{k+1} - \boldsymbol{f}_k \tag{12}$$

Thus, a simultaneous state and fault estimation error is given by:

$$\bar{\boldsymbol{e}}_{k+1} = \begin{bmatrix} \boldsymbol{e}_{k+1} \\ \boldsymbol{e}_{f,k+1} \end{bmatrix} = \boldsymbol{X}\bar{\boldsymbol{e}}_k + \boldsymbol{Z}\boldsymbol{v}_k \tag{13}$$

where

$$\boldsymbol{X} = \tilde{\boldsymbol{A}} - \tilde{\boldsymbol{K}}\tilde{\boldsymbol{C}}, \quad \tilde{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{L} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}, \quad \tilde{\boldsymbol{K}} = \begin{bmatrix} \boldsymbol{K} \\ \boldsymbol{F} \end{bmatrix}, \quad \tilde{\boldsymbol{C}} = \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{Z} = \begin{bmatrix} \boldsymbol{W} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$$

$$\boldsymbol{v}_k = \begin{bmatrix} \boldsymbol{w}_k \\ \boldsymbol{\varepsilon}_k \end{bmatrix} \tag{14}$$

For the purpose of further analysis, let us assume that $\boldsymbol{w}_k$ and $\boldsymbol{\varepsilon}_k$ belong to ellipsoidal sets $\mathbb{E}_w$ and $\mathbb{E}_\varepsilon$, which are given by:

$$\mathbb{E}_w = \{\boldsymbol{w} : \boldsymbol{w}^T\boldsymbol{Q}_w\boldsymbol{w} \leq 1\}, \quad \boldsymbol{Q}_w \succ \boldsymbol{0} \tag{15}$$

$$\mathbb{E}_\varepsilon = \{\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}_k^T\boldsymbol{Q}_\varepsilon\boldsymbol{\varepsilon} \leq 1\}, \quad \boldsymbol{Q}_\varepsilon \succ \boldsymbol{0} \tag{16}$$

It is thus evident that $\boldsymbol{v}_k \in \mathbb{E}_v$ with

$$\mathbb{E}_v = \{\boldsymbol{v} : \boldsymbol{v}^T\boldsymbol{Q}_v\boldsymbol{v} \leq 1\}, \quad \boldsymbol{Q}_v = \mathrm{diag}(\boldsymbol{Q}_w, \boldsymbol{Q}_\varepsilon) \tag{17}$$

For the sake of stability analysis, the following Lyapunow function can be used:

$$V_k = \bar{\boldsymbol{e}}_k^T\boldsymbol{P}\bar{\boldsymbol{e}}_k, \quad \boldsymbol{P} \succ \boldsymbol{0}. \tag{18}$$

Moreover, the following definitions are reminded.

Definition 1 System (13) is strictly quadratically bounded for all allowable $\boldsymbol{v}_k \in \mathbb{E}v$ if $\bar{\boldsymbol{e}}_k^T\boldsymbol{P}\bar{\boldsymbol{e}}_k > 1$ implies $\bar{\boldsymbol{e}}_{k+1}^T\boldsymbol{P}\bar{\boldsymbol{e}}_{k+1} < \bar{\boldsymbol{e}}_k^T\boldsymbol{P}\bar{\boldsymbol{e}}_k$ for any $\boldsymbol{v}_k \in \mathbb{E}_v$.

Note that strict quadratic boundedness of (13) guarantees that $V_{k+1} < V_k$ for any $\boldsymbol{v}_k \in \mathbb{E}_v$ when $V_k > 1$.

Definition 2 A set $\mathbb{E}$ is a positively invariant set for (13) for all $\boldsymbol{v}_k \in \mathbb{E}_v$ if $\bar{\boldsymbol{e}}_k \in \mathbb{E}$ implies $\bar{\boldsymbol{e}}_{k+1} \in \mathbb{E}$ for any $\boldsymbol{v}_k \in \mathbb{E}_v$.

Lemma 1 *The following statements are equivalent* [1]*:*

1. *The system* (13) *is strictly quadratically bounded for all* $\boldsymbol{v}_k \in \mathbb{E}_v$.

2. *The ellipsoid*

$$\mathbb{E} = \{\bar{\boldsymbol{e}}_k : \bar{\boldsymbol{e}}_k^T \boldsymbol{P} \bar{\boldsymbol{e}}_k \leq 1\} \tag{19}$$

is an invariant set for (13) *for any* $\boldsymbol{v}_k \in \mathbb{E}_v$.

3. *There exists a scalar* $\alpha \in (0, 1)$ *such that:*

$$\begin{bmatrix} \boldsymbol{X}^T \boldsymbol{P} \boldsymbol{X} - \boldsymbol{P} + \alpha \boldsymbol{P} & \boldsymbol{X}^T \boldsymbol{P} \boldsymbol{Z} \\ \boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{X} & \boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Z} - \alpha \boldsymbol{Q}_v \end{bmatrix} \preceq \boldsymbol{0} \tag{20}$$

Theorem 1 *The system* (1) *is strictly quadratically bounded for all* $\boldsymbol{v}_k \in \mathbb{E}_v$ *if there exist matrices* $\boldsymbol{P} \succ \boldsymbol{0}$, $\boldsymbol{U}$ *and a scalar* $\alpha \in (0, 1)$, *such that the following inequality is satisfied:*

$$\begin{bmatrix} -\boldsymbol{P} + \alpha \boldsymbol{P} & \boldsymbol{0} & \bar{\boldsymbol{A}}^T \boldsymbol{P} - \bar{\boldsymbol{C}}^T \boldsymbol{U}^T \\ \boldsymbol{0} & -\alpha \boldsymbol{Q}_v & \boldsymbol{Z}^T \boldsymbol{P} \\ \boldsymbol{P} \boldsymbol{A} - \boldsymbol{U} \bar{\boldsymbol{C}} & \boldsymbol{P} \boldsymbol{Z} & -\boldsymbol{P} \end{bmatrix} \preceq \boldsymbol{0} \tag{21}$$

Proof Inequality (20) can be written as

$$\begin{bmatrix} \boldsymbol{X}^T \\ \boldsymbol{Z}^T \end{bmatrix} \boldsymbol{P} \begin{bmatrix} \boldsymbol{X} & \boldsymbol{Z} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{P} + \alpha \boldsymbol{P} & \boldsymbol{0} \\ \boldsymbol{0} & -\alpha \boldsymbol{Q}_v \end{bmatrix} \preceq \boldsymbol{0} \tag{22}$$

which by Schur complement is given by:

$$\begin{bmatrix} -\boldsymbol{P} + \alpha \boldsymbol{P} & \boldsymbol{0} & \boldsymbol{X}^T \boldsymbol{P} \\ \boldsymbol{0} & -\alpha \boldsymbol{Q}_v & \boldsymbol{Z}^T \boldsymbol{P} \\ \boldsymbol{P} \boldsymbol{X} & \boldsymbol{P} \boldsymbol{Z} & -\boldsymbol{P} \end{bmatrix} \preceq \boldsymbol{0} \tag{23}$$

Substituting

$$\boldsymbol{P} \boldsymbol{X} = \boldsymbol{P}(\tilde{\boldsymbol{A}} - \tilde{\boldsymbol{K}} \tilde{\boldsymbol{C}}) = \boldsymbol{P} \tilde{\boldsymbol{A}} - \boldsymbol{P} \tilde{\boldsymbol{K}} \tilde{\boldsymbol{C}} = \boldsymbol{P} \tilde{\boldsymbol{A}} - \boldsymbol{U} \tilde{\boldsymbol{C}} \tag{24}$$

Finally, the design procedure is reduced to solving (21) and then calculating

$$\tilde{\boldsymbol{K}} = \begin{bmatrix} \boldsymbol{K} \\ \boldsymbol{F} \end{bmatrix} = \boldsymbol{P}^{-1} \boldsymbol{U} \tag{25}$$

3 Adaptive Threshold Design

The objective of this section is to provide a design procedure that can be used for calculating an adaptive threshold overbounding the unknown real state and fault. Let us start with the following Lemma:

Lemma 2 *If the system* (13) *is strictly quadratically bounded for all* $\boldsymbol{V}k \in \mathbb{E}_v$, *then there exists* $\alpha \in (0, 1)$ *such that:*

$$\boldsymbol{v}_k \leq \gamma_k(\alpha), \quad k = 0, 1, \ldots \tag{26}$$

where the sequence $\gamma_k(\alpha)$ *is defined as*

$$\gamma_k(\alpha) = (1-\alpha)^k \boldsymbol{V}_0 + 1 - (1-\alpha)^k, \quad k = 0, 1, \ldots \tag{27}$$

It can easily be shown that (27) can be written as

$$\gamma_k(\alpha) = (1-\alpha)^k (\boldsymbol{V}0 - 1) + 1, \quad k = 0, 1, \ldots \tag{28}$$

which for $\alpha \in (0, 1)$ converges exponentially to one. Moreover, the convergence ratio depends on α, i.e., the larger α, the faster convergence.

It is thus evident from (26), that for any $\boldsymbol{v}_k \in \mathbb{E}_v$, $\boldsymbol{e}_k$ falls into the ellipsoid

$$\bar{\boldsymbol{e}}_k^T \left[\gamma_k^{-1}(\alpha)\boldsymbol{P}\right] \bar{\boldsymbol{e}}_k \leq 1 \tag{29}$$

The principal semi-axes of the ellipsoid (29) are given by:

$$z_{i,k} = \gamma_k(\alpha)^{\frac{1}{2}} \lambda_i(\boldsymbol{P})^{-\frac{1}{2}}, \quad i = 1, \ldots, n+r \tag{30}$$

where $\lambda_i(\boldsymbol{P})$ is the ith eigenvalue of $\boldsymbol{P}$.

This clearly indicates that:

$$-z_{i,k} \leq \bar{\boldsymbol{e}}_{i,k} \leq z_{i,k}, \quad i = 1, \ldots, n+r \tag{31}$$

Thus, from (31) and (9) as well as (11), it is evident that:

$$\hat{\boldsymbol{x}}_{i,k} - z_{i,k} \leq \boldsymbol{x}_{i,k} \leq \hat{\boldsymbol{x}}_{i,k} + z_{i,k}, \quad i = 1, \ldots, n \tag{32}$$

$$\hat{\boldsymbol{f}}_{j,k} - z_{i,k} \leq \boldsymbol{f}_{j,k} \leq \hat{\boldsymbol{f}}_{j,k} + z_{i,k}, \quad j = 1, \ldots, r \quad i = n+1 \ldots, n+r \tag{33}$$

Note that (32) and (33) can be perceived as confidence intervals for $\boldsymbol{x}_k$ and $\boldsymbol{f}_k$, respectively. As it was already mentioned, γ_k converges exponentially to one, while the convergence depends on α. Thus, the steady-state length of (32) and (33) depends solely on the eigenvalues of $\boldsymbol{P}$, which describe the size of the ellipsoid. This means that in order to minimize the size of the ellipsoid it is necessary to maximize the underlying cost function given by one of the following criteria:

- D—optimality criterion

$$\phi(\boldsymbol{P}) = \det(\boldsymbol{P}) \tag{34}$$

- E—optimality criterion

$$\phi(\boldsymbol{P}) = \lambda_{\max}(\boldsymbol{P}) \tag{35}$$

- A—optimality criterion

$$\phi(\boldsymbol{P}) = \mathrm{trace}(\boldsymbol{P}) \tag{36}$$

Thus, (34) can be used to minimize the volume of (29), while (35) minimizes the length of its largest axis. Finally, A-optimality criterion suppresses the average axis length of the ellipsoid. This means that the final design procedure is

Step 1 *Select* $\alpha > 0$

Step 2 *Solve*

$$\boldsymbol{P}^* = \arg\max_{\boldsymbol{P}>\boldsymbol{0}} \phi(\boldsymbol{P})$$

under the constraint (21).

4 Case Study

In order to verify the suggested approach, it is exercised for the multi-tank system. The multi-tank system (Fig. 1) is arranged for simulating the real industrial multi-tank system in the laboratory conditions. It can be regularly used to practically certify both, linear and non-linear control, identification and diagnostics methods. The examined system consists of three separate tanks. Those tanks are placed one above another and armed with drain valves and level sensors based on hydraulic pressure measurements. All together has a different cross-section, because of system nonlinearities. The lower bottom tank is a water container for the system. A variable which describes speed of the water pump is used to fill the lofty tank. The force of gravity influences the water outflows from the tanks. The presented multi-tank system has been designed, to act with an external, PC-based digital controller. The control computer exchanges data with the level sensors, also communicates with valves and a pump via a dedicated I/O board and the power interface. Real time software is controlled by the I/O board, which is maintained in the Matlab/Simulink environment. The scheme of the multi-tank system is portrayed in Fig. 2.

Fig. 1 Multi-tank system

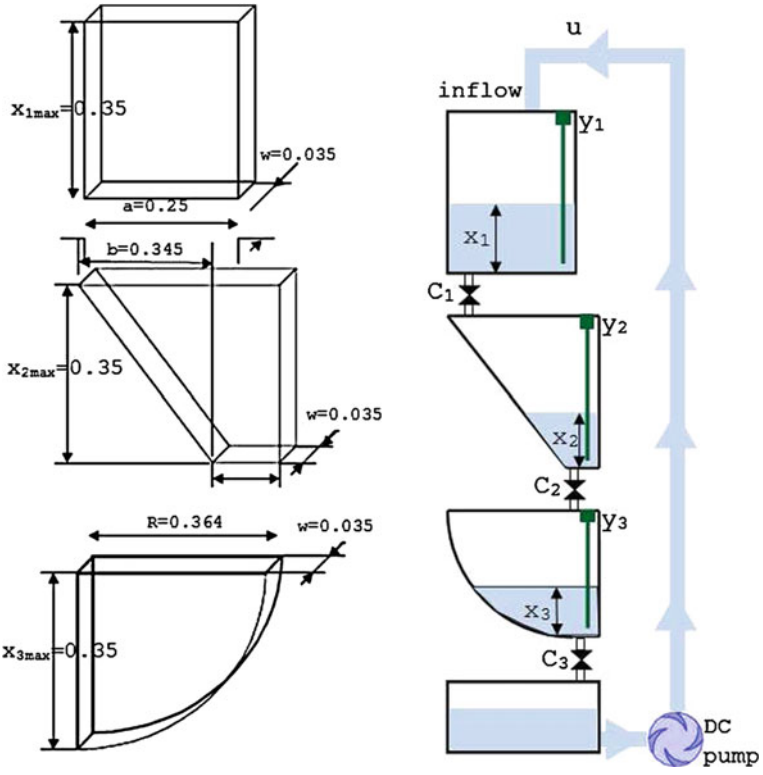


Fig. 2 Geometrical parameters of the tanks

The sampling time ($T_s = 0.1$ s) yield the discrete-time non-linear system (1) and (2), where:

$$\bar{\boldsymbol{A}} = \begin{bmatrix} 0.9972 & 0 & 0 \\ 0.0692 & 0.9138 & 0 \\ 0 & 0.0007 & 0.9995 \end{bmatrix}, \quad \bar{\boldsymbol{B}} = \begin{bmatrix} 11.4286 \\ 0 \\ 0 \end{bmatrix}$$

$$\bar{\boldsymbol{C}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \bar{\boldsymbol{C}}_f = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Now, let us consider the following two faults scenarios:

- Actuator fault scenario:

$$\boldsymbol{f}_{a,k} = \begin{cases} -0.1\boldsymbol{u}_k, & \text{for } 200 \leq k \leq 500 \\ 0, & \text{otherwise} \end{cases} \tag{37}$$

- Sensor fault scenario (first sensor):

$$\boldsymbol{f}_{s,k} = \begin{cases} -0.05, & \text{for } 550 \leq k \leq 700 \\ 0, & \text{otherwise} \end{cases} \tag{38}$$

Figure 3 shows actuator fault and its estimate, while Fig. 4 portrays sensor fault along with its estimate. In Fig. 5, the third system state and its estimate is presented. It is clear that the state observer designed in Sect. 2 estimates the third system state with a satisfactory accuracy. The obtained results confirm high performance of the proposed methodology and recommend it for the FTC applications.

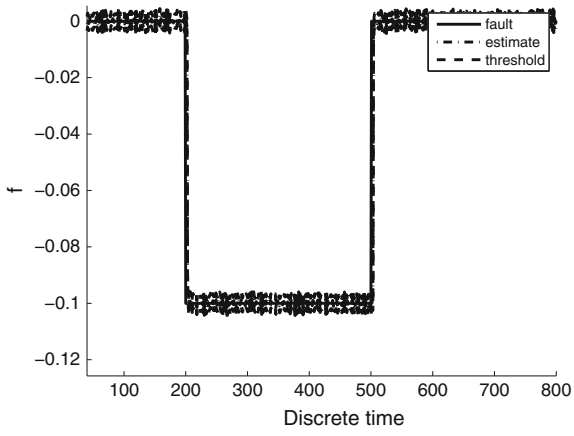


Fig. 3 Actuator fault and its estimate

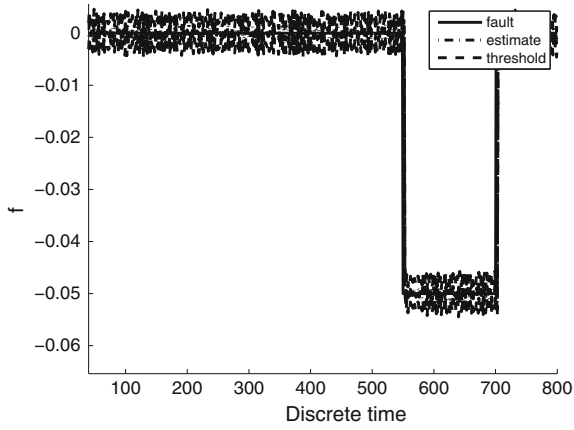


Fig. 4 Sensor fault and its estimate

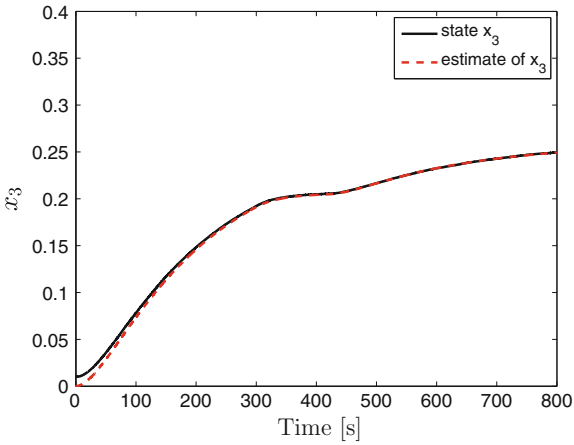


Fig. 5 Third state and its estimate

5 Conclusions

The main objective of this paper was to propose a novel structure and design a procedure of fault identification for a class of linear discrete-time systems. First, a procedure for transforming the model with sensor and actuator fault is transformed into an extended one with actuator fault only. Subsequently, an actuator fault estimation scheme is developed with the quadratic boundedness approach. This strategy allows to obtain an invariant set of the extended estimation error that is further used for designing an adaptive threshold overbounding the true state and fault. The preliminary results show the performance of the proposed approach, which was obtained based on the laboratory three-tank system.

Acknowledgments The work was supported by the National Science Centre of Poland under grant: 2013/11/B/ST7/01110.

References

1. Alessandri, A., Baglietto, M., Battistelli, G.: Design of state estimators for uncertain linear systems using quadratic boundedness. Automatica **42**(3), 497–502 (2006)
2. Boulkroune, B., Djemili, I., Aitouche, A., Cocquempot, V., et al.: Robust nonlinear observer design for actuator fault detection in diesel engines. Int. J. Appl. Math. Comput. Sci. **23**(3), 1–7 (2013)
3. Buciakowski, M., de Rozprza-Faygel, M., Ochalek, J., Witczak, M.: Actuator fault diagnosis and fault-tolerant control: application to the quadruple-tank process. J. Phys. Conf. Ser. **570**, 10 (2014)
4. Buciakowski, M., Witczak, M., Luzar, M.: Robust fault-tolerant control for a multi-tank system. In: 11th International Conference on Diagnostics of Processes and Systems—DPS 2013. p. 12. Łagów Lubuski, Polska (2013)
5. Ducard, G.: Fault-tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles. Springer, Berlin (2009)
6. Gillijns, S., De Moor, B.: Unbiased minimum-variance input and state estimation for linear discrete-time systems. Automatica **43**, 111–116 (2007)
7. Keller, J.Y., Darouach, M.: Two-stage Kalman estimator with unknown exogenous inputs. Automatica **35**(2), 339–342 (1999)
8. Korbicz, J., Kościelny, J., Kowalczuk, Z., Cholewa, W.E.: Fault Diagnosis. Models, Artificial Intelligence Applications. Springer, Berlin (2004)
9. Mrugalski, M.: An unscented Kalman filter in designing dynamic GMDH neural networks for robust fault detection. Int. J. Appl. Math. Comput. Sci. **23**(1), 157–169 (2013)
10. Mrugalski, M., Witczak, M.: State-space GMDH neural networks for actuator robust fault diagnosis. Adv. Electr. Comput. Eng. **12**(3), 65–72 (2012)
11. Mrugalski, M.: Advanced Neural Network-based Computational Schemes for Robust Fault Diagnosis, Studies in Computational Intelligence, vol. 510. Springer, Heidelberg (2014)
12. Noura, H., Theilliol, D., Ponsart, J., Chamseddine, A.: Fault-tolerant Control Systems: Design and Practical Applications. Springer, Berlin (2003)
13. Puig, V.: Fault diagnosis and fault tolerant control using set-membership approaches: application to real case studies. Int. J. Appl. Math. Comput. Sci. **20**(4), 619–635 (2010)
14. Veluvolu, K., Kim, M., Lee, D.: Nonlinear sliding mode high-gain observers for fault estimation. Int. J. Syst. Sci. **42**(7), 1065–1074 (2011)
15. Witczak, M.: Modelling and Estimation Strategies for Fault Diagnosis of Non-linear Systems. Springer, Berlin (2007)
16. Witczak, M., Buciakowski, M., Mrugalski, M.: An H_∞ approach to fault estimation of non-linear systems: application to one-link manipulator. In: 19th International Conference of Methods and Models in Automation and Robotics, pp. 456–461. West Pomeranian University of Technology, Szczecin, Międzyzdroje, Polska (2014)
17. Witczak, M.: Fault Diagnosis and Fault-tolerant Control Strategies for Non-linear Systems. Lecture Notes in Electrical Engineering, vol. 226. Springer, Berlin (2014)

Single Fault Isolability Metrics of the Binary Isolating Structures

Michał Bartyś

Abstract First, the critical analysis of isolability features of three chosen metrics is presented. It is proved that all studied and analysed metrics are not sufficiently well designed with respect to requirements of optimization methods referring to the binary fault isolation structures. Next, a set of the novel relatively simple definitions of single fault isolability metrics based on the binary valued structures of residual sets are introduced. The basic features of these metrics are formulated and proved. They allow for quantized fault isolability analysis within well defined normalized spans as well as are indicative of the degree of fault isolability. Finally, an illustrative example of determination of the isolability metrics of the electro-pneumatic actuator assembly is provided.

Keywords Fault isolation · Isolability metrics · Binary isolating structure of residual sets · Binary diagnostic matrix · Fault distinguishability

1 Introduction

This paper brings a new sight on single fault isolability metrics of binary isolation structures of residual sets. The necessity of redefinition of isolability metrics is motivated among others by the growing number of applications of optimization techniques applied e.g. for constrained sensor placement problems [10, 11]. The main contribution of this paper is the proposition of the set of definitions of the new isolability metrics that are useful for automatized evaluation of the isolability features of diagnostic systems based on binary structures of residual sets. The necessity for an extension of the set of fault isolability metrics arose in result of the critical

M. Bartyś (✉)
Institute of Automatic Control and Robotics, Warsaw University of Technology,
św. A. Boboli 8, 02-525 Warsaw, Poland
e-mail: bartys@mchtr.pw.edu.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_5

discussion from the perspective of practical implementations [7] of optimization approaches applied for development of diagnostic systems based on binary structures of residual sets.

The structure of the paper is following. The introductory part is devoted to an in-depth discussion of the three chosen isolability metrics. Section 2 brings the definitions of the four new isolability metrics and discusses most important features of those metrics as well as provides an illustrative and practical example for a hand driven optimization of the isolability structure based on electro-pneumatic actuator assembly. Section 3 summarizes achieved results.

1.1 Theoretical Diagnosis Accuracy

Probably one of the simplest definitions of the fault isolability is the diagnosis accuracy index [2]. Suppose that diagnosis is a set of N elementary blocks and each elementary block consists of n_i indistinguishable faults. Diagnosis accuracy index referred to each elementary block is defined as reciprocal of the number of indistinguishable faults indicated in this block:

$$d_i = \frac{1}{n_i} \tag{1}$$

Hence, the less the number of faults in the elementary diagnose, the greater is the accuracy index of the elementary diagnosis. Since $d_i \in [1/n_i \ \ldots \ 1]$, the accuracy index is dependent hyperbolically on the cardinality of the set of indistinguishable faults. This progressively prefers higher fault isolability. Unfortunately, this one-dimensional index does not directly address all important factors having influence on the quality of the fault isolation. For example, the diagnosis accuracy index (1) does not reflect directly the degree of bidirectional isolability [5]. On the other hand, the bidirectional isolability degree is very useful for the assessment of the quality of structural isolability particularly in the development phase of the diagnostic systems. Also, the theoretical mean diagnosis accuracy (2) defined in [2] does not bring us towards solving this problem:

$$d = \frac{1}{N} \sum_{i=1}^{N} d_i \tag{2}$$

Regardless of drawbacks stated above, the theoretical diagnosis index seems to be very practical. It applies almost for all model based fault isolation approaches.

1.2 Conditional and Unconditional Fault Distinguishability

Extension of the theoretical diagnosis accuracy index was introduced in [8] in order to make allowance for assessment of the quality of fault isolation in case of multi-valued Fault Information Systems (FIS). In [8], Kościelny et al. introduced definitions of conditional and unconditional fault distinguishability as well as definition of unconditional fault indistinguishability. These definitions also apply to binary structures of residual sets, further referred to as Binary Diagnostic Matrix (BDM). In fact, each BDM is a specific bi-valued FIS system.

In case of FIS, the two faults are conditionally distinguishable, if at least one pair of their alternative signatures is distinguishable [1]. In the BDM, the number of alternative signatures of any fault equals 1. Therefore, the conditional fault distinguishablity is identical with unconditional fault distinguishability and is referred briefly to as the fault distinguishability.

In case of BDM, the two faults are unconditionally indistinguishable or briefly indistinguishable in an isolating structure, if their signatures are identical and non-zero. Therefore, if the faults are unconditionally indistinguishable in a BDM, the BDM is neither weakly nor unidirectional nor bidirectional isolating in the sense of Gertler's definitions [5]. On the other hand, all faults in a BDM can be either mutually distinguishable or not. If all faults in the BDM structure are distinguishable, then a BDM is at least weakly isolating. Therefore, in case of BDM, the unconditional indistinguishability is the same as fault indistinguishability.

It is worth to mention that definitions of unconditional fault distinguishability do not allow directly infer of unidirectional or bidirectional strong isolability of the isolating structure. Moreover, neither the definition of unconditional fault distinguishability [8] nor definition of weakly isolating structure [5] refer to the case of multiple faults.

A simple measure of the fault distinguishability which is similar to (2) is introduced in [8] in the form:

$$\Gamma = \frac{1}{N}\sum_{i=1}^{N}\gamma_i; \quad \Gamma \in (0,1] \tag{3}$$

where $\gamma_i = \left|E_m\right|^{-1}$ is the reciprocal of the cardinality of the set of unconditionally distinguishable faults within the block E_m.

Since the unconditionally distinguishable faults are those that are distinguishable in the BDM structure, the fault distinguishabilty measure (2) can be used as a simple metric of the fault distinguishablity. If all faults in the BDM are indistinguishable, then $\Gamma = 1/n$ (where n is the total number of faults). If all faults in the BDM are distinguishable, then $\Gamma = 1$. Unfortunately, this metric does not reflect directly the degree (strength) of isolability of the structure what makes this factor less usable in case of searching for best isolating structure.

1.3 The Measure of Fault Isolability

The Rostek's measure [11] of the fault isolability of binary diagnostic matrix is

$$m(V) = \frac{1}{(n+1)n} \sum_{i=1}^{n} d_i \tag{4}$$

where: n is the total number of faults; d_i is the number of faults that are excluded by the signature of fault f_i increased by 1 in case if a detectable fault f_i.

Let us now discuss the proposed metric. From description given in [11], it may be concluded that signature of the fault f_i excludes fault f_k if for any pair of signature entries $\langle v_{j,i}, v_{j,k} \rangle$ assigned respectively for both faults holds:

$$\exists v_{j,i} \wedge \overline{v}_{j,k} = 1;\ \forall i,k \in [1 \ldots n];\ \forall i \neq k;\ \forall j \in [1 \ldots m] \tag{5}$$

where $v_{j,i}$, $v_{j,k}$ are the jth binary entries of the signatures V_i and V_k of faults f_i and f_k; n is the number of the signatures; m is the number of rows of BDM.

Now we will show that condition (5) holds for all signatures of any binary weakly isolating structure.

Definition 1 The binary signature V_i of any single fault f_i excludes fault f_k unilaterally right if for any $i,k \in [1 \ldots n]$ and $i < k$ holds:

$$(V_i \vee V_k \neq V_k) \tag{6}$$

Definition 2 The binary signature V_k of any single fault f_k excludes fault f_i unilaterally left if for any $i,k \in [1 \ldots n]$ and $i < k$ holds:

$$(V_i \vee V_k \neq V_i) \tag{7}$$

Definition 3 Any pair of binary signatures $\langle V_i, V_k \rangle$ of single faults $\langle f_i, f_k \rangle$ is excluding both faults bilaterally if for any $i,k \in [1 \ldots n]$ and $i < k$ holds:

$$(V_i \vee V_k \neq V_k) \wedge (V_i \vee V_k \neq V_i) \tag{8}$$

Remark 1 Equation (8) is a conjunction of (6) and (7). Hence, the bilaterally excluding signatures of any pair of faults are simultaneously right and left excluding.

The d_i value in the formula (4) of the fault isolability measure totalize all left, right and bilateral exclusions. This assumption is disputable, because the necessary condition of fault isolability is the only difference of the fault signatures independently if they are left, right or bilaterally excluding. The quality of isolability depends rather on how the signatures are differentiated (see Sect. 2).

Lemma 1 *The all signatures of single faults f_i in a weakly isolating structure may be at least excluding unilaterally.*

Proof Weakly isolating structure of a binary residual sets is consisting of exclusively different and non-zero single fault signatures [5]. For all pairs of all fault signatures in this structure holds $(V_i \neq V_k)$. Then, obviously if $(V_i \vee V_k = V_k)$ then $(V_i \vee V_k \neq V_i)$ and if $(V_i \vee V_k = V_i)$ then $(V_i \vee V_k \neq V_k)$. Hence, all signatures in a weakly isolating structure are at least excluding unilaterally. □

Remark 2 The minimal value of the fault isolability measure (4) equals $1/(n+1)$ of any binary isolating structure for which $\nexists i : (V_i = 0); i \in [1, \ldots, n]$.

Proof The measure (4) has minimal value when the total number of right, left and bilateral exclusions equals 0. This may happens if any pair of fault signatures is neither unilateral nor bilateral isolating. In this case, any of conditions (6)–(8) does not hold. Therefore, holds $(V_i \vee V_k = V_k) \vee (V_i \vee V_k = V_i)$. This means that all signatures of all faults in the structure are equal, i.e. $(V_i = V_k)$ and faults are indistinguishable but still detectable. Therefore, $d_i = 1;\ \forall i \in [1 \ldots n]$. In this case the fault isolability measure (4) equals:

$$m(V) = \frac{1}{(n+1)n} \sum_{i=1}^{n} 1 = \frac{1}{(n+1)} \tag{9}$$

□

Remark 3 The minimal value of the metrics (4) equals 1/2 for any weakly isolating binary diagnostic matrix for which $n > 1$.

Proof Assume that none of signatures is excluding bilaterally. Then, according to Lemma 1, the signature of any single fault f_i in a weakly isolating structure must be excluding unilaterally. Therefore, the signature of any fault f_i excludes right or left-laterally exactly $(n - i)$ remained faults. Hence, the minimal total number of excluded faults in the isolating structure equals: $n(n-1)/2$. After substitution to formula (4) and remembering that all faults in a weakly isolating structure are detectable, we obtain:

$$m(V) = \frac{1}{(n+1)n} \sum_{i=1}^{n} d_i = \frac{1}{(n+1)n} \left[\frac{n(n-1)}{2} + n \right] = \frac{1}{2} \tag{10}$$

□

Remark 4 The maximal value of the metrics (4) of the fault isolability for a strongly isolating binary diagnostic matrix equals $n/(n+1)$.

Proof Assume case in which all signatures of all faults are excluding bilaterally. Then each fault f_i excludes unilaterally $2 * (n - i)$ remained faults. Hence, the maximal count of excluded faults in the isolating structure equals: $n(n-1)$. After substitution of this result to formula (4) and remembering that all faults in a weakly isolating structure are detectable, we obtain:

$$m(V) = \frac{1}{(n+1)n} \sum_{i=1}^{n} d_i = \frac{1}{(n+1)n} [n(n-1)+n] = \frac{n}{(n+1)} \tag{11}$$

□

Remark 5 The measure (4) does not purely extract and average the fault isolability because the number of signatures of the isolating structure affects the value of this metrics too. Fortunately, this influence is a monotonic function of n. In addition, this measure is not normalized, what can not be considered in the categories of the formal excellence.

Example 1 In accordance with (10), the minimal value of the single fault isolability measure (4) equals 0.5 for a weakly isolating structure. But it does not mean that values of $m(V)$ less then 0.5 are indicative for non weakly isolating structures. Moreover $m(V)$ can be greater then 0.5 even in case, in which some faults are indistinguishable.

For example, the measure $m(V) = 0.767$ for the left hand BDM matrix shown in Table 1, despite the fact that faults f_4 and f_5 are indistinguishable. On the other hand, the measure $m(V)$ is also equal to 0.767 for the right hand matrix in Table 1, despite the fact, that all faults in this matrix are distinguishable.

Therefore, this measure might not be considered as indicative for the fault indistinguishability. If the fault indistinguishability is of concern, the metric (4) should be evaluated carefully. Additionally, this metric as well as these defined in (1)–(3) do not explicitly refer to the degree of strong isolability [5]. This should be assumed as their disadvantages, particularly when searching for optimal solutions under specified constrains [10, 11]. In Sect. 2 we introduce definitions of the isolability metrics that do not possesses the above mentioned drawbacks.

Table 1 An example of the binary diagnostic matrices

(a)

S/F	f_1	f_2	f_3	f_4	f_5
s_1	1	0	0	0	0
s_2	0	1	0	0	0
s_3	0	0	1	0	0
s_4	0	0	0	1	1

(b)

S/F	f_1	f_2	f_3	f_4	f_5
s_1	1	0	0	0	1
s_2	0	1	0	0	0
s_3	0	0	1	0	0
s_4	0	0	0	1	1

2 Redefinition of Single Fault Isolability Metrics

The necessary condition of fault isolability is the diversification of fault signatures independently, whether they are left, right or bilaterally excluding. The measure (4) totalize all left and right bilateral exclusions. Therefore, in general, its value should be assumed as overscored. Please note, that d_i value might be increased either by 1 or 2 in case, in which signatures are different. Therefore, the measure (4), besides its other advantages, cannot be considered in terms of precise measure of isolability. In this section we will try to define the metric that explicitly exposes this feature. Firstly, we define the isolability degree.

2.1 The Isolability Degree

Let us define a single fault diversity matrix M.

Definition 4 Given single fault diversity binary matrix $M[m : c]$:

$$M = [M_{1,2}, M_{1,3}, \ldots, M_{1,n}, M_{2,3}, M_{2,4} \ldots, M_{(n-1),n}] \tag{12}$$

where $M_{i,k} = V_i \otimes V_k$; $i \in [1 \ldots (n-1)]$; $k \in [(i+1) \ldots n]$; $c = \binom{n}{2} = \frac{n(n-1)}{2}$.
Each column of the single fault diversity matrix M contains information regarding distinguisability of each pair of single faults regardless of their unilateral or bilateral characterization. Obviously, the signatures of any faults $\langle f_i, f_k \rangle$ are different if there exists at least one non-zero entry of $M_{i,k}$. Let the measure of diversity of signatures be the algebraic sum of all non-zero entries in each column of the matrix M. Let us denote this number as $\mathfrak{d}_{i,k}$

$$\mathfrak{d}_{i,k} = \sum_{j=1}^{m} m_{j,i,k} \tag{13}$$

where: $m_{j,i,k} = v_{j,i} \otimes v_{j,k}$ is the entry of M. Let us create a diversity vector $\mathfrak{D}$:

$$\mathfrak{D}[1 : c] = [\mathfrak{d}_{1,2}, \mathfrak{d}_{1,3}, \ldots, \mathfrak{d}_{1,n}, \mathfrak{d}_{2,3}, \mathfrak{d}_{2,4} \ldots, \mathfrak{d}_{(n-1),n}] \tag{14}$$

and its two derivatives: isolability vector $\mathfrak{I}$ and isolability degree vector $\mathfrak{E}$. In order to simplify notation let us firstly totalize the number of differences between signatures of any pair of faults $\langle f_i, f_k \rangle$ for which $k > i$.

$$\mathfrak{i}_i = \sum_{k=i+1}^{n-1} \mathfrak{d}_{i,k} \tag{15}$$

Definition 5 Given isolability vector $\mathfrak{F}[1 : (n-1)]$:

$$\mathfrak{F} = [\mathfrak{i}_1, \mathfrak{i}_2, \ldots, \mathfrak{i}_{n-1}] \tag{16}$$

Let us determine the minimal number of differences between signatures of each pair of faults $\langle f_i, f_k \rangle$ for which $k > i$. In order to do it, let us create $(n-1)$ sets $\mathfrak{d}_i$ referred to as sets of distinctiveness of the ith fault which are associated with each pair of faults $\langle f_i, f_k \rangle$ under condition: $k > i$.

$$\mathfrak{d}_i = \{\mathfrak{d}_{i,(i+1)}, \ldots, \mathfrak{d}_{i,n}\};\ i = [1 \ldots (n-1)] \tag{17}$$

Now, we determine the set $\mathfrak{E} = \{\mathfrak{e}_1, \ldots \mathfrak{e}_{n-1}\}$ of infima of the sets $\mathfrak{d}_i$. This set will be further referred to as an isolability degree set of each pair of signatures of single faults $\langle f_i, f_k \rangle$:

$$\mathfrak{e}_i = \bigwedge \mathfrak{d}_i;\ \ \forall i \in [1 \ldots (n-1)] \tag{18}$$

Definition 6 Infimum $\mathfrak{e}$ of all infima of a set of distinctiveness of each pair of signatures of single faults $\langle f_i, f_k \rangle$ is referred to as isolability degree of the binary isolating structure.

$$\mathfrak{e} = \bigwedge_{i=1}^{n-1} \mathfrak{e}_i;\ \ \mathfrak{e} = \bigwedge_{i=1}^{n-1} \bigwedge_{k=1}^{n-1} \mathfrak{d}_k \tag{19}$$

Remark 6 Please note that $\mathfrak{e} \in \mathbb{N}^0$.

Lemma 2 *The isolability degree of the binary isolating structure equals 0 for any isolating structure possessing at least one pair of indistinguishable signatures.*

Proof Indeed, if there exists any pair of indistinguishable signatures then both signatures are indistinctive and therefore there exist at least one infimum of the distinctiveness set equals 0 and obviously the infimum of distinctiveness infima sets also equals 0.

$$\exists(V_i = V_k) \Rightarrow (V_i \otimes V_k = 0) \Rightarrow \exists(\mathfrak{d}_{i,k} = 0) \Rightarrow (\mathfrak{e}_i = 0) \Rightarrow (\mathfrak{e} = 0) \tag{20}$$

□

Remark 7 The Lemma 2 can be reversed. Therefore, if the isolation structure contains at least one pair of indistinguishable signatures then the isolability degree of this structure equals 0. This feature might be used for easy checking if the fault or faults are distinguishable within the structure.

Lemma 3 *The isolability degree of the weakly binary isolating structure at least equals 1.*

Proof According to [5], all columns are different and non-zero in any weakly isolating structure. Therefore, all pairs of signatures are distinguishable and therefore all infima of the distinctiveness set are non-zero and obviously the infimum of distinctiveness infima sets is also non-zero.

$$\nexists(V_i = V_k) \Rightarrow (V_i \otimes V_k \neq 0) \Rightarrow \nexists(\mathfrak{d}_{i,k} = 0) \Rightarrow \nexists(\mathfrak{e}_i = 0) \Rightarrow (\mathfrak{e} > 0) \tag{21}$$

□

Remark 8 The isolability degree $\mathfrak{e}$ is a natural number. Its value equals 0 for any isolating structure containing indistinguishable faults. Any value of $\mathfrak{e} \geqslant 1$ indicates weakly isolating structure. The higher the value of isolability degree, the higher is the quality of fault isolation.

Lemma 4 *The isolability degree of any diagonal isolating structure equals 2.*

Proof The only locations of a diagonal binary isolating structure that are different from 0 are those laying on diagonal for which $v_{i,i} = 1$. This means that each fault signature from the pair $\langle f_i, f_k \rangle$ will contain only one "1" value but in different positions for any $i \neq k$. Hence, any pair of fault signatures will differ on two positions. Therefore, all pairs of signatures are distinguishable and all infima of the distinctiveness set are equal to 2 and obviously the infimum of distinctiveness infima sets is also equal to 2.

$$\nexists(v_{i,k} = 1) \wedge (v_{i,i} = 0) \Rightarrow \nexists(\mathfrak{d}_{i,k} \neq 2) \Rightarrow \nexists(\mathfrak{e}_i \neq 2) \Rightarrow (\mathfrak{e} = 2) \tag{22}$$

Remark 9 The quality of isolation assessed in terms of isolability degree of all diagonal structures is equal. Clearly, the diagonal isolation structure is column canonical. Therefore, it is bidirectionally strongly isolating of degree 1.

Lemma 5 *The isolability degree* $\mathfrak{e}$ *is indicative for bidirectional strongly isolating structure of degree* $(\mathfrak{e} - 1)$.

Proof According to Gertler's definition, a structure is bidirectional strongly isolating of degree k if it is weakly isolating and if no column can be obtained from any other column by changing up to k elements. If for any isolation structure holds $\mathfrak{e} > 0$, then in accordance with Lemma 3, this structure is weakly isolating. In fact, the isolability degree $\mathfrak{e}$ indicates minimal number of different positions in each pair of columns of the isolation structure (19). Bidirectional strong isolation of degree k requires that, for each pair of columns, there must be at least $(k+1)$ positions where both columns in this pair are different. Hence, $(k = \mathfrak{e} - 1)$. □

Remark 10 The isolability degree $\mathfrak{e} > 1$ indicates bidirectional strong isolating structure and simultaneously points out degree of isolation. Therefore the isolability degree can be used for the assessment of the quality of fault isolation.

2.2 Mean Isolability Metrics

Definition 7 Given the mean value of isolability metric of single faults $\mathfrak{d}$:

$$\mathfrak{d} = \frac{1}{(n-1)} \sum_{i=1}^{n-1} \mathfrak{i}_i \tag{23}$$

where $(n-1)$ is the number of elements of the isolability vector $\mathfrak{F}$.
By substitution $\mathfrak{i}_i$ into (15) and $\mathfrak{d}_{i,k}$ into (13), we finally obtain:

$$\mathfrak{d} = \frac{1}{(n-1)} \sum_{i=1}^{n-1} \sum_{k=i+1}^{n-1} \sum_{j=1}^{m} m_{j,i,k} = \frac{1}{I} \sum_{i=1}^{I} \sum_{j=1}^{m} m_{j,i} \tag{24}$$

where $I = \binom{n}{2} = \frac{n(n-1)}{2}$.

Remark 11 The maximal mean value of isolability metric $\mathfrak{d}$ is equal to the number m of rows of the isolation structure.

Proof Since the maximal value of each ith sum is $\sum_{j=1}^{m} m_{j,i} = m$, then maximal value of $\mathfrak{d}$ equals:

$$max(\mathfrak{d}) = \frac{1}{I} \sum_{i=1}^{I} m = m \tag{25}$$

□

This makes allowance for normalization of the mean value of isolability metric.

Definition 8 Given normalized mean isolability of single faults metric $\mathfrak{d}^N$:

$$\mathfrak{d}^N = \frac{1}{m} \mathfrak{d} \tag{26}$$

Remark 12 The normalized mean isolability of single faults metric is related with the number of rows of isolation structure. Therefore, even by the same value of mean isolability, the normalized mean isolability may differ depending on the number of rows of isolation structure.

2.3 Normalized Drastic Isolability Metric

Definition 9 Given normalized drastic isolability of single faults metric $\mathfrak{d}^D$:

$$\mathfrak{d}^D = \begin{cases} 0 & \Rightarrow \ \mathfrak{e} = 1 \\ \mathfrak{d}^N & \Rightarrow \ \mathfrak{e} \geqslant 1 \end{cases} ; \ \mathfrak{e} \in \mathbb{N}^0 \tag{27}$$

Remark 13 The normalized drastic isolability metric of single faults is identical with normalized mean isolability metric for any weakly isolating structure. Otherwise it equals 0. The normalized drastic isolability metric may have practical meaning. For instance, it easily allows automatized seeking for strongly isolating structures of a predefined isolability degree.

Lemma 6 *The isolability degree and mean isolability metric are insensitive to the rows of isolation structure consisting of exclusively "1" values.*

Proof Because $v_{p,i} \otimes v_{p,k} = m_{p,j,i} = 0$ for any $i \neq j;\ i,j \in [1 \ldots n]$ for the isolation structure which pth row consists exclusively of "1" values then according to (13)

$$\mathfrak{d}_{i,k} = \sum_{j=1}^{m} m_{j,i,k} = 0 + \sum_{j=1}^{p-1} m_{p,i,k} + \sum_{j=p+1}^{m-1} m_{j,i,k} = \mathfrak{d}_{i,k} \tag{28}$$

Therefore, this pth row might be rejected without the loss of $\mathfrak{d}_{i,k}$ and all derivatives of $\mathfrak{d}_{i,k}$ including the isolability degree $\mathfrak{e}$ (19) and mean isolability metric $\mathfrak{d}$ (24). □

Remark 14 Clearly, it follows from Lemma 6 that redundant rows consisting exclusively of "1's" might be rejected from isolating structure without the loss of its isolability features. But the normalized mean isolability degree as well as drastic isolability metrics will rise in reverse manner with the number of rejected redundant rows. This feature of $\mathfrak{d}^N$ might be considered as practical because supports automatized seeking for "excellent" isolating structures.

2.4 The Practical Example

Example 2 Let us consider an electro-pneumatic actuator consisting of: electronic controller, electro-pneumatic converter, pneumatic servo-motor, control valve and electro-mechanical servo-motor's stem position feedback. Faults affect the final control element parts selectively. The actuator is equipped with three sensors measuring: supply air pressure P_z, servo-motor chamber's pressure P_s and servo-motor's stem position X. The causal graph of the final control element is depicted in Fig. 1. The list of 6 considered faults of the actuator assembly is presented in Table 2 and the list of diagnostic signals in Table 3.

The binary valued isolation structure of residual sets of electro-pneumatic actuator derived from Tables 2 and 3 is depicted in Table 4a. As can be seen from Table 4a, there is a set of 4 indistinguishable faults: $\{f_2 \wedge f_3 \wedge f_5 \wedge f_6\}$. Thus these faults gathered together form a block of indistinguishable faults $f_{2,3,5,6}$.

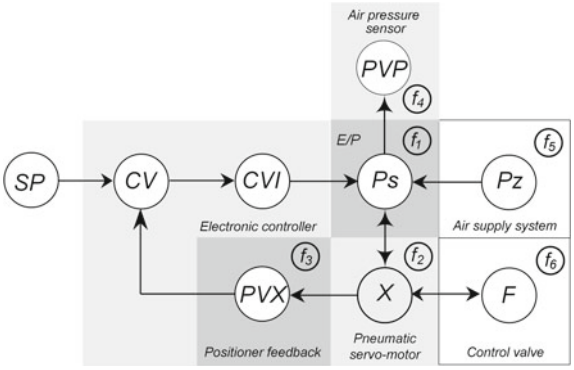


Fig. 1 Causal graph of the electro-pneumatic final control element. Notions: *SP*—set point, *CV*—control value, *CVI*—control value of the electro-pneumatic transducer, *CVI*—control value of the electro-pneumatic transducer, *PVP*—servomotor's chamber pressure measurement, *PVX*—stem displacement measurement, *F*—flow rate

Table 2 The list of considered faults

Fault	*Description*	*Type*
f_1	*E/P transducer*	*Internal*
f_2	*Pneumatic servo-motor*	*Internal*
f_3	*Position feedback*	*Internal*
f_4	*Pressure sensor fault*	*Internal*
f_5	*Supply air pressure*	*External*
f_6	*Control valve*	*External*

Table 3 List of diagnostic signals s_j and residuals r_j

Signal	*Residual*	*Affected by*
s_1	$r_1 = X - f(CV)$	f_1,f_2,f_3,f_5,f_6
s_2	$r_1 = X - f(X_{(t-20)}, X_{(t-30)}, X_{(t-30)}, CV)$	f_1,f_2,f_3,f_5,f_6
s_2	$r_2 = X - f(P_s)$	f_2,f_3,f_5,f_6
s_3	$r_3 = X - f(CVI)$	f_1,f_2,f_3,f_5,f_6
s_4	$r_4 = P_s - f(CVI)$	f_1,f_2,f_3,f_4,f_5,f_6
s_5	$r_5 = P_s - f(CV)$	f_1,f_2,f_3,f_4,f_5,f_6

Table 5 provides a list of values of fault isolation metrics calculated for isolation structures of Tables 1 and 4. The following can be derived from Table 5.

(a) according to (21), the isolating structures presented in Tables 1a and 4a,b,c, are neither weakly nor strongly isolating because degrees of isolation of all these matrices equal 0. In turn, the isolating structures presented in Table 1b and Table 4d are, because single fault isolation metrics $\mathbf{e} = 1$ and not bidirectional strongly isolating because degree of strong isolation is $k = 0$

Table 4 Binary isolation structures of the electro-pneumatic actuator: (a) structure derived immediately from Tables 2 and 3; (b) structure obtained from (a) by rejecting row s_5 of exclusively "1"s; (c) structure obtained from (b) by rejecting redundant row s_2; (d) the structure obtained from (c) by creating blocks of indistinguishable faults

(a)

S/F	f_1	f_2	f_3	f_4	f_5	f_6
s_1	1	1	1	0	1	1
s_2	1	1	1	0	1	1
s_3	0	1	1	0	1	1
s_4	1	1	1	1	1	1
s_5	1	1	1	1	1	1

(b)

S/F	f_1	f_2	f_3	f_4	f_5	f_6
s_1	1	1	1	0	1	1
s_2	1	1	1	0	1	1
s_3	0	1	1	0	1	1
s_4	1	1	1	1	1	1

(c)

S/F	f_1	f_2	f_3	f_4	f_5	f_6
s_1	1	1	1	0	1	1
s_3	0	1	1	0	1	1
s_4	1	1	1	1	1	1

(d)

S/F	f_1	f_4	$f_{2,3,5,6}$
s_1	1	0	1
s_3	0	0	1
s_4	1	1	1

Table 5 Analysis of fault isolation of the system with electro-pneumatic actuator

Metric	*Symbol*	*Ref.*	1*a*	1*b*	4*a*	4*b*	4*c*	4*d*
Mean diagnosis accuracy	d	[2]	0.88	1.00	0.75	0.75	0.75	1.00
Fault distinguishability	Γ	[8]	0.88	1.00	0.75	0.75	0.75	1.00
Fault isolability measure	$m(V)$	[11]	0.77	0.77	0.36	0.36	0.36	0.50
Degree of strong isolation	k	[5]	–	0	–	–	–	0
Degree of isolation metric	$\mathfrak{e}$	(19)	0	1	0	0	0	1
Mean isolability metric	$\mathfrak{d}$	(24)	1.80	2.00	1.20	1.20	0.87	1.33
Normalized mean isolability	$\mathfrak{d}^N$	(26)	0.45	0.50	0.24	0.30	0.29	0.44
Drastic isolability metric	$\mathfrak{d}^D$	(27)	0.00	0.50	0.00	0.00	0.00	0.44

(b) fault isolability measure $m(V)$ might be identical for the isolation structures of the same size despite the fact that only one of them is weakly isolating (see columns 1a and 1b in Table 5)
(c) the values of mean diagnosis accuracy d and fault distinguishability metric Γ may not be sensitive to the number of rows of isolation structure (see columns 4a, 4b and 4c in Table 5); this means that these metrics may not be indicative for alternative versions of isolation structures with less rows
(d) the mean isolability metric $\mathfrak{d}$ is insensitive to the number of rows only in case if there are rows consisting of exclusively "0"s or "1"s. These rows can easily be rejected in the early phase of isolation structure processing.

3 Summary

The collection of new metrics of the isolability of the single faults in the binary isolation structures have been introduced in this paper keeping in mind their applicability particularly for the diagnostic systems optimization tasks. The definitions of those metrics have been preceded by a critical analysis of their chosen literature counterparts. It has been proved that the introduced metrics better capture the meaning and sense of the fault isolability.

The introduced metrics seem to have advantages over their literature counterparts. They either quantize fault isolability within normalised span or directly indicate indistinguishability of faults as well as weak or strong bidirectional isolability or degree of isolability. The appropriate aggregate of these metrics in the form of a drastic mean isolability was also proposed. Finally, an illustrative example of determination of the isolability metrics for the analysis of isolability of the electro-pneumatic actuator was shown.

References

1. Bartyś, M.: Generalised reasoning about faults based on diagnostic matrix. Int. J. Appl. Math. Comput. Sci. **23**(2), 407–417 (2013)
2. Bartyś, M., Patton, R., Syfert, M., de las Heras, S., Quevedo, J.: Introduction to the DAMADICS actuator FDI benchmark study. Control Eng. Pract. **14**(6), 577–596. Pergamon-Elsevier (2006)
3. Basseville, M.: On fault detectability and isolability. Eur. J. Control **7**(6), 625–637 (2001)
4. Blanke, M., Staroswiecki, M.: Structural design of systems with safe behaviour under single and multiple faults. In: Proceedings of IFAC Symposium Safeprocess, Beijing, PR China, pp. 511–515 (2006)
5. Gertler, J.: Fault Detection and Diagnosis in Engineering Systems. Marcel Dekker Inc., New York (1998)
6. Isermann, R.: Fault diagnosis systems. An Introduction from Fault Detection to Fault Tolerance. Springer, New York (2006)
7. Korbicz, J., Kościelny, J.M., Kowalczuk, Z., Cholewa, W. (eds.): Fault diagnosis. Models, Artificial Intelligence, Applications. Springer, Berlin (2004)

8. Kościelny, J.M., Bartyś, M., Rzepiejewski, P., Sá da Costa, J.M.G.: Actuator fault distinguishability study of the DAMADICS benchmark problem. Control Eng. Pract. **14**(6), 645–652, Pergamon-Elsevier (2006)
9. Patton, R., Frank, P., Clark, R. (eds.): Issues of Fault Diagnosis for Dynamic Systems. Springer, Berlin (2000)
10. Rosich, A., Sarrate, R., Puig, V., Escobet, T.: Efficient optimal sensor placement for model-based FDI using an incremental algorithm. In: 46th IEEE Conference on Decision and Control, pp. 2590–2595 (2007)
11. Rostek, K.: Measure of fault isolability of diagnostic system. In: 25th International Workshop on Principles of Diagnosis, Gratz, Austria, 8–11 September 2014

Optimal Sensor Placement Under Budgetary Constraints

Kornel Rostek

Abstract In this paper a method for solving the optimal sensor placement problem is presented. The approach maximizes diagnosability and isolability, while not exceeding the budgetary constraint. The proposed strategy is based on a Binary Diagnostic Matrix. The proposed isolability measure distinguishes weak and strong isolability. It uses the branch-and-bound algorithm to find a solution. The method is then tested on a Fuel Cell Stack System.

Keywords Fault detection · Fault isolation · Binary diagnostic matrix · Optimal sensor placement · Integer programming · Fuel cell stack

1 Introduction

The performance of a fault diagnosis system for a given industrial process is strongly dependent on available measurements. Usually, already available sensors include only those needed for process control and safety. Those are often insufficient for diagnostic purposes and do not provide required fault detectability or isolability. Therefore, the FDI system designer often wants to propose new sensors to be installed. It is vital to achieve the best FDI system performance with minimal additional costs. The problem of optimal sensor selection can be understood as a combinatorial problem of selecting the optimal set of new measurements.

There are different approaches to solving the sensor placement problem for fault diagnosis. Proposed methods depend on the structure of diagnostic system, requirements for detectability and isolability effectiveness or a form of notation of a fault - symptom relation. Currently, there is no universal method of solving the optimal

This work was partially supported by the Warsaw University of Technology, Faculty of Mechatronics Deans Grant 504/01536.

K. Rostek (✉)
Warsaw University of Technology, Warsaw, Poland
e-mail: k.rostek@mchtr.pw.edu.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_6

sensor placement problem. Usually, the aim of optimization procedure is to minimize a cost or a number of sensors fulfilling predefined performance requirements. The other approach is to maximize the performance of a diagnostic system while not exceeding the available budget or the maximum number of new sensors. In this paper, systems based on Binary Diagnostic Matrices are taken under consideration. The analysed problem is to find a set of sensors, within budgetary constraint, providing best performance for a predefined set of faults.

The Binary Diagnostic Matrix (BDM, incidence matrix, fault occurrence matrix) is the most widely used form of notation of the faults-symptoms relation in FDI systems. It can be obtained by many methods, for example by modelling with fault influence, structural analysis or using expert knowledge. It was named in [1] as a structure matrix of a residual set. BDM is easily understood by industry experts and engineers which facilitate the cooperation and expert knowledge transfer between them and designers of the FDI system.

In recent years, numerous papers were published, devoted to different problems of the optimal sensor placement. The model-based Fault Detection and Isolation treats faults as deviations from normal values of process parameters or as unknown process inputs. Faults are detected when models behave differently than measured signals. This often takes form of a difference between a measured value and calculated one. It is called a residual. Residuals are often found using Analytical Redundancy Relations (ARRs). In [11] a method for finding the optimal sensor set based on ARRs is proposed. First all ARRs are found under the assumption that all sensor candidates are installed. Then, a sensor set is selected that minimizes the cost while satisfying detectability and isolability requirements. However, this solution is computationally expensive. A modified, incremental approach, using Minimal Structurally Overdetermined (MSO) sets, was proposed in [5]. In [9] the Binary Integer Programming is used to find the optimal sensor set using a set of all possible MSO sets. FDI requirements were ensured with non-linear constraints. The resulting problem is computationally difficult to solve. This method was further improved in [2] and [4]. The FDI requirements were specified as linear constraints. The cost function was also linear, so the problem was Binary Integer Linear Programming (BILP). It can be efficiently solved with branch-and-bound algorithm with standard Linear Programming (LP) solver. Those methods were thoroughly compared in [7]. In [12] a method of sensor placement, that makes it possible to satisfy diagnosability specifications, based on structural matrices was presented. This method does not require to design ARRs. Budgetary constraints were analysed in [8]. Branch-and-bound algorithm was used to obtain the optimal solution.

This paper presents a new method of solving the optimal sensor placement problem for FDI system based on BDM. The main contribution of this paper is solving performance maximization problem, measured with a method proposed in [6]. This approach allows to formulate a BILP problem with a budgetary constraint using a Binary Diagnostic Matrix. Then, the optimization problem can be efficiently solved.

To illustrate the proposed method, the model of a Fuel Cell Stack System is used. It was described in [7].

The paper is organized as follows. In Sect. 2, the theoretical background is given. Binary Diagnostic Matrix is defined. In Sect. 2.2 the measure of fault isolability is introduced. Section 3 describes a Binary Integer Linear Programming problem formulation. Section 4 presents a version of a branch-and-bound algorithm, which is used in this paper. Section 5 describes the application of the proposed method to a Fuel Cell Stack System. In Sect. 6 some remarks and conclusions are given.

2 Theoretical Background

2.1 Binary Diagnostic Matrix

Binary Diagnostic Matrix is a form of notation of a relationship specified by the Cartesian product of diagnostic signals set S and faults set $F = \{f_i = 1, 2, \ldots, n\}$. Each column $V_i = [v_{1,i}, v_{2,i}, v_{J,i}]^T$ of binary diagnostic matrix V can be associated with a fault f_i. Often column V_i is called signature of fault f_i. An example of binary diagnostic matrix is shown in Table 1.

Definition 1 Faults f_k,$f_m \in F$ are weakly isolable if their signatures are different.

In the example from Table 1 all faults with exception of a pair (f_2,f_3) are weakly isolable. A weak isolability in some applications is not sufficient. It is possible that due to different sensitivity of diagnostic tests or process dynamics some signals appear earlier and match a signature of a different, weakly isolated fault. In the above example, appearance of only the signal s_1 may be insufficient to reliably indicate the fault f_1. Later signals s_2 or s_3 may appear indicating faults f_2, f_3 or f_5. Therefore a stronger property is required.

Definition 2 Faults f_k,$f_m \in F$ are strongly isolable if their signatures are different and no signature can arise from other by turning "1"s into "0"s [1].

Weak isolability is a necessary condition for strong isolability.

In Table 1 faults f_5 and f_6 are strongly isolable. From Theorem 2 following statement can be extrapolated:

Definition 3 The signature V_i is excluding a fault f_k if V_i is different than V_k and V_i cannot be obtained from V_k by turning "1"s into "0"s.

Table 1 The Binary Diagnostic Matrix example. 1 in *j*th row and *i*th column means that f_i is detectable with signal s_j

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
s_1	1	1	1		1			
s_2		1	1	1	1			
s_3				1	1	1		1
s_4						1	1	1
s_5							1	1

Opposite does not have to be true. If faults are mutually excluding each other, then they are strongly isolable. In Table 1 signature V_2 is excluding f_1. Opposite is not true so they are not strongly isolable.

2.2 The Measure of Fault Isolability of Diagnostic System for Binary Diagnostic Matrix

The proposed algorithm for calculation of the measure of isolability for diagnostic system consists of two main steps:

- For each fault calculate the coefficient d_i where d_i is a number of faults excluded by signature of fault f_i. Then increase d_i by 1 if fault is detectable, because every detectable fault excludes the faultless mode. After this step: $d_i = 0$ if a fault is not detectable and $d_i \geq 1$ if a fault is detectable.
- Calculate the measure of fault isolability with following formula:

$$m(V) = \frac{1}{(n+1)n} \sum_{i=1}^{n} d_i \tag{1}$$

where n is the total number of faults.

For example, when analyzing diagnostic system described by Table 1, following values are obtained: $d_1 = 5$, (as V_1 excludes f_4, f_6, f_7, f_8 and +1 for detectability), $d_2 = d_3 = 6$; $d_4 = d_6 = d_7 = 7$; $d_5 = d_8 = 8$. After applying formula (6) we obtain $m(V) = 0.75$.

This algorithm for calculating isolability can be used without modifications for any bi-valued form of notation of a diagnostic relation, such as directions in residual space or a sequence of symptoms. To be able to use this isolability measure in any system it is enough to be able to determine if signature corresponding to given fault excludes other faults.

The value of this measure has valid physical interpretation. It is an average fraction of possible single fault diagnoses that are excluded with fault signature. There are n+1 possible diagnoses: faultless state and n single faults.

Maximum possible value for proposed isolability measure is 1. For a fully weakly isolating system V_{wi}:

$$m(V_{wi}) = \frac{1}{2} \tag{2}$$

As the number of strongly isolable pairs increases, the value of measure approaches 1. With every unisolable pair the value decreases. This is a very important property of this measure. It allows to compare performance of any FDI system to weakly isolating one. If $m(V_{wi}) < \frac{1}{2}$ then the FDI system is worse than fully weakly isolating, If $m(V_{wi}) > \frac{1}{2}$ then it is better.

3 Problem Formulation

Working under budgetary constraints is a typical situation for FDI system designer. Let S be the set of all possible new diagnostic signals. Let $s_k \in S$ and $s_k = 1$ if signal s_k is available in FDI system. Using (1) it is possible to construct an optimization problem for finding the set of sensors offering the best detectability and isolability:

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & \frac{1}{(n+1)n} \sum_{i=1}^{n} d_i \\ \text{s.t.} \quad & c^T x \leq b \\ & x_i \in \{0, 1\} \end{aligned} \tag{3}$$

where: x is a vector of decision variables, $x_i = 1$ when ith sensor is chosen, c is cost vector and b is available budget.

The value of d_i can be calculated in the following way:

$$d_i = \sum_{j=0}^{n} \underset{s}{max}(S_{i,j}) \tag{4}$$

$S_{i,j} \subset S$ is a set of diagnostic signals sensitive to a fault f_i and not sensitive to a fault f_j.

By introducing the faultless mode f_0 we can easily include fault detectability. $S_{i,0}$ is a set of signals allowing to detect a fault f_i.

Similarly s_k can be calculated as:

$$s_k = \underset{x}{min}(X_{s_k}) \tag{5}$$

where X_{s_k} is a set of new measurements necessary for a signal s_k.

Let us analyze a simple example (Table 2).

Following equations can be constructed:

$$\begin{aligned} s_1 &= min\{x_1, x_2\} \\ s_2 &= min\{x_2, x_3\} \\ d_1 &= max\{s_1\} = s_1 \\ d_2 &= max\{s_1, s_2\} + max\{s_2\} = max\{s_1, s_2\} + s_2 \end{aligned} \tag{6}$$

Table 2 Simple BDM with sensor requirements for diagnostic signals

	f_0	f_1	f_2
$s_1(x_1, x_2)$		1	1
$s_2(x_2, x_3)$			1

The maximization of the objective function $\frac{1}{6}(d_1 + d_2)$, substituted with (6) is a very difficult, non-linear optimisation problem. There are techniques that will allow us to solve this problem easily.

Lemma 1
Maximize $\max_x \{x_1, \ldots, x_k\}$ *problem has the same optimal solution as linear, constrained problem:*

$$\begin{array}{ll} \underset{x}{maximize} & x_{k+1} \\ s.t. & x_{k+1} \leq x_1 \\ & \vdots \\ & x_{k+1} \leq x_k \end{array} \tag{7}$$

Proof The solution of (7) is the biggest lower bound (infimum) of a set $\{x_1, \ldots, x_k\}$. For finite sets it is always equal to minimum, which is the solution of the original problem. □

Lemma 2
BILP problem $\max_x \{x_1, \ldots, x_k\}$ *s.t.* $x_i \in \{0, 1\}$ *has the same optimal solution as:*

$$\begin{array}{ll} \underset{x}{maximize} & min\{x_1 + x_2 + \cdots + x_k, 1\} \\ s.t. & x_i \in \{0, 1\} \end{array} \tag{8}$$

Proof When x_i is binary i.e. $x_i \in \{0, 1\}$, $\max\{x_1, \ldots, x_k\} = 0$ iff $x_1 = x_2 = \cdots = x_k = 0$. In such a case $x_1 + x_2 + \cdots + x_k = 0$. □

Using Lemma 1 and 2 it is possible to construct a higher dimensional, linear equivalent of (3) by substituting d_i and s_k with constrained new control variables.

Let us continue the example from Table 2. Using Lemma 1 and 2 we introduce new control variables $x_{s_1}, x_{s_2}, x_{d_2}$. Then following BILP problem can be obtained:

$$\begin{array}{ll} \underset{x}{\text{maximize}} & \frac{1}{6}(x_{s_1} + x_{s_2} + x_{d_2}) \\ \text{s.t.} & c^T x \leq b \\ & x_{s_1} \leq x_1 \\ & x_{s_1} \leq x_2 \\ & x_{s_2} \leq x_2 \\ & x_{s_2} \leq x_3 \\ & x_{d_2} \leq x_{s_1} + x_{s_2} \\ & x_{d_2} \leq 1 \\ & x_i \in \{0, 1\} \end{array} \tag{9}$$

4 Solving Problem with Branch-and-Bound Algorithm

The sensor placement problem stated in previous section is solved by Algorithm 1. It is modification of depth-first branch-and-bound algorithm. This algorithm is often used to solve BILP problems.

In order to solve the original problem this algorithm solves a series of relaxed LP problems. Constraint $x \in \{0, 1\}$ is replaced with $0 \leq x \leq 1$. In Algorithm 1 constraints are represented as 4 matrices A, B, A_{eq}, B_{eq}, such that $Ax \leq B$ for inequality constraints and $A_{eq}x = B_{eq}$ for equality constraints. If a solution to the relaxed problem is not feasible in the original problem (x_i is not integer) then 2 new LP problems (nodes) are created. One with constraint $x_i = 0$ and the other with $x_i = 1$. Both problems are solved. This operation is called branching. Usually, there is more than one non-integer variable. There are many heuristics focused on choosing one. In Algorithm 1 the most infeasible variable is chosen. A variable feasibility is calculated as $abs(0.5 - x)$. The variable was given a value closest to 0.5.

Algorithm 1 $x = BandB(f, A, b, Aeq, beq, x_B)$

```
if relaxed problem is infeasible then return 0
end if
x ← solveLP(f,A,b,Aeq,beq)
if f^T x < f^T x_b then return 0                                  ▷ Bounding
else
    if isInteger(x) then
        if f^T x = f^T x_b then
            if C(x) < C(x_B) then
                return x
            else
                return 0
            end if
        else
            return x
        end if
    else                                                          ▷ Branching
        i ← chooseInfisible(x)
        [Aeq_1 Beq_1] ← addConstraint(0, i, Aeq, beq)
        [Aeq_2 Beq_2] ← addConstraint(1, i, Aeq, beq)
        x_1 ← BandB(f, A, b, Aeq_1, beq_1, x_B)
        if f^T x_1 ≥ f^T x_B then x_B ← x_1
        end if
        x_2 ← BandB(f, A, b, Aeq_2, beq_2, x_B)
        if f^T x_2 ≥ f^T x_B then x_B ← x_2
        end if
        return x_B
    end if
end if
```

The algorithm retains current best integer solution. A solution to relaxed problem is always equal or better than solution to the original problem. Therefore if optimal solution to relaxed problem is worse than current best, then such node can be discarded. Node is also discarded if after branching operation LP problem is no longer feasible. Those operations are called bounding.

Algorithm 1 considers one more situation. Let $C(x)$ be total cost of solution x. If solution in current node is integer and it has equal value of objective function as the best solution, then total costs are compared and the cheaper solution is selected. This ensures that the cheapest solution offering optimal isolability is selected. Unfortunately, it has negative impact on the number of analysed nodes. In the worst case, the algorithm will visit most nodes from $2^{card(x)}$ possible. It will occur only when there are multiple solutions with an equal objective function value.

Algorithm 1 is implemented with MATLAB and standard simplex solver is used to solve LP problems.

5 Fuel Cell Stack System

Fuel cells are electrochemical devices that convert the chemical energy from a gas fuel into electricity. In this example PEMFC (Polymer Electrolyte Membrane Fuel Cell) is analysed. Hydrogen is supplied to an anode and oxygen to a cathode. In the result of a chemical reaction water and electric energy are produced. Detailed description can be found in [3]. In [7] simplified structural model can be found. Variables used in it are presented in Table 3. In the same paper approximate sensor costs were proposed.

The method proposed in this paper requires information about possible residual generators and new sensors required by them. One way to obtain it, is to utilize the available expert knowledge. There are also automated methods to generate such structures. The structures presented in Table 4 and their FDI performance in Table 5 were obtained using a casual graph method presented in [10]. SN_0 means that any of following sensors can be used: p_{ca}, W_{cp}, $W_{sm,out}$, ω_{cp}, p_{sm}. Some of this signals require only sensors that are already installed. To manage this situation already measured values were added to optimization problem with cost equal to 0.

Using method presented in previous section BILP problem was formulated using Tables 4 and 5. Resulting problem has 59 variables and 132 inequality constraints. There are 2^{59} possible solutions. In Fig. 1, optimal values of isolability measure for different budget constraints are presented. The total cost of best performing FDI system is 7. Further increase of a budget does not improve isolability (Table 2). A number of nodes created by branch-and-bound algorithm is shown in Fig. 3. It is worth noticing that even in the worst case total number of nodes was much smaller than theoretical 2^{59} or even 2^8 (where 8 is a number of new sensors that can be installed).

Table 3 Model variables

Control variables, already measured		
V_{cm}	Compressor voltage	
W_{cp}	Air flow through the compressor	
I_{st}	Stack current	
V_{st}	Stack voltage	
Unmeasurable variables		
τ_{cm}	Compressor motor torque	
τ_{cp}	Load torque	
$W_{v,inj}$	Humidifier injector flow	
Possible sensor locations		Costs
ω_{cp}	Compressor angular speed	2
p_{sm}	Supply manifold pressure	1
$W_{sm,out}$	Supply manifold exit flow	5
p_{ca}	Cathode pressure	1
$W_{ca,out}$	Cathode output flow	5
p_{an}	Anode pressure	1
$W_{an,in}$	Anode input flow	5
$W_{rm,out}$	Return manifold exit flow	5
System faults		
$f_{p_{sm}}$	Compressor fault	
$f_{W_{sm,out}}$	Supply manifold fault	
$f_{W_{rm,out}}$	Return manifold fault	
$f_{I_{st}}$	Fuel Cell Stack fault	
f_n	Cell fault	

Obtained results can be compared to those from [7]. There, the optimal solution obtained with three different methods was $S^* = \{p_{ca}, p_{an}\}$ with total cost $C(S^*) = 2$. This is the same result as obtained by author with budget constraint $2 \leq B < 6$. In Fig. 1 one can see that it is possible to improve value of isolability measure. It is because, with additional measurements, some weakly isolating pairs of faults can become strongly isolating. With budget $B = 6$ sensors $\{p_{an}, W_{rm,out}\}$ are chosen and with budget $B \geq 7$ sensors $\{p_{ca}, p_{an}, W_{rm,out}\}$. Obtained results depend on a set of new model structures considered during optimization procedure. If this set is incomplete, then results may not be optimal.

Table 4 Analyzed models for residual generation

	Output	Inputs
1	$W_{rm,out}$	V_{cm},I_{st}
2	$W_{rm,out}$	I_{st},SN_0
3	p_{an}	I_{st}
4	p_{ca}	W_{cp}
5	p_{ca}	V_{cm}
6	W_{cp}	V_{cm}
7	W_{cp}	p_{ca},V_{cm}
8	V_{st}	p_{an},I_{st},SN_0
9	V_{st}	p_{an},V_{cm},I_{st}
10	V_{st}	V_{cm},I_{st}
11	V_{st}	I_{st},SN_0

Table 5 BDM for the Fuel Cell Stack System

	$f_{p_{sm}}$	$f_{W_{sm,out}}$	$f_{W_{rm,out}}$	$f_{I_{st}}$	f_n
s_1	1	1	1		
s_2			1		
s_3					1
s_4		1			
s_5	1	1			
s_6	1	1	1		
s_7	1	1			
s_8				1	
s_9	1	1		1	
s_{10}	1	1		1	1
s_{11}				1	1

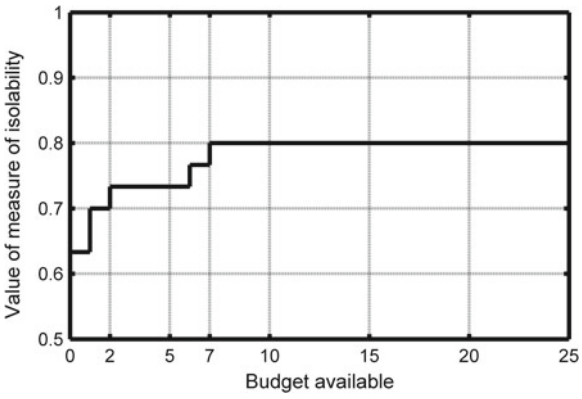


Fig. 1 Values of isolability measure of optimal solutions for different budget values

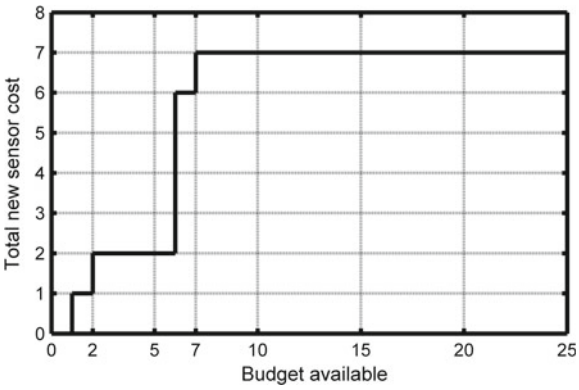


Fig. 2 Total costs of optimal solutions for different budget values

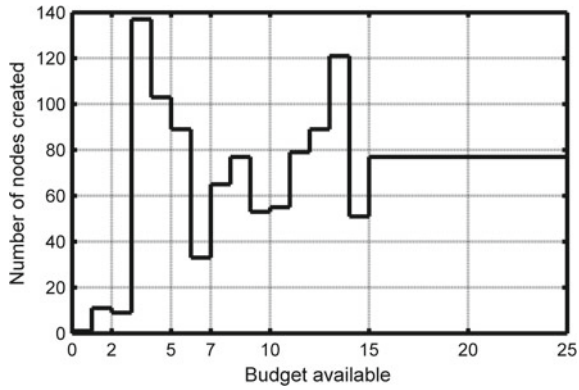


Fig. 3 Number of nodes created to obtain optimal solutions for different budget values

6 Conclusion

In this paper the sensor placement problem was addressed. A key contribution of this work is introduction of a new measure of fault isolability as an objective function to Binary Integer Programming problem. A strategy of introducing new variables which allows to obtain BILP problem was presented. This allows to use efficient tools to find optimal sensors sets.

The method was applied to a Binary Diagnostic Matrix, but proposed measure of fault isolability is able to describe polyvalent systems such as Fault Information Systems (FIS). Further work in this area is required.

References

1. Gertler, J.: Fault Detection and Diagnosis. Marcel Dekker Inc., New York (1998)
2. Nejjari, F., Sarrate, R., Rosich, A.: Optimal sensor placement for fuel cell system diagnosis using Bilp formulation. In: 18th Mediterranean Conference on Control and Automation (MED), pp. 1296–1301. IEEE (2010)
3. Pukrushpan, J.T.: Modeling and Control of Fuel Cell Systems and Fuel Processors. University of Michigan, Ann Arbor (2003)
4. Rosich, A., Sarrate, R., Nejjari, F.: Optimal sensor placement for fdi using binary integer linear programming. In: 20th International Workshop on Principles of Diagnosis (2009)
5. Rosich, A., Sarrate, R., Puig, V., Escobet, T.: Efficient optimal sensor placement for model-based FDI using an incremental algorithm. In: 46th IEEE Conference on Decision and Control, pp. 2590–2595. IEEE (2007)
6. Rostek, K.: Measure of fault isolability of diagnostic system. In: 25th International Workshop on Principles of Diagnosis (2014)
7. Sarrate, R., Nejjari, F., Rosich, A.: Model-based optimal sensor placement approaches to fuel cell stack system fault diagnosis. Fault Detect. Superv. Saf. Tech. Process. **8**(1) (2012)
8. Sarrate, R., Nejjari, F., Rosich, A.: Sensor placement for fault diagnosis performance maximization under budgetary constraints. In: 2nd International Conference on Systems and Control (2012)
9. Sarrate, R., Puig, V., Escobet, T., Rosich, A.: Optimal sensor placement for model-based fault detection and isolation. In: 46th IEEE Conference on Decision and Control, pp. 2584–2589. IEEE (2007)
10. Sztyber, A.: Model based diagnosis using causal graph. Pomiary, Automatyka, Robotyka **17**, 83–88 (2013)
11. Travé-Massuyes, L., Escobet, T., Olive, X.: Diagnosability analysis based on component-supported analytical redundancy relations. IEEE Trans. Syst. Man Cybern. Part A: Syst. Hum. **36**(6), 1146–1160 (2006)
12. Yassine, A., Ploix, S., Flaus, J.M.: A method for sensor placement taking into account diagnosability criteria. Int. J. Appl. Math. Comput. Sci. **18**(4), 497–512 (2008)

Part II
Estimation and Identification

Discrete-Time Estimation of Nonlinear Continuous-Time Stochastic Systems

Mariusz Domżalski and Zdzisław Kowalczuk

Abstract In this paper we consider the problem of state estimation of a dynamic system whose evolution is described by a nonlinear continuous-time stochastic model. We also assume that the system is observed by a sensor in discrete-time moments. To perform state estimation using uncertain discrete-time data, the system model needs to be discretized. We compare two methods of discretization. The first method uses the classical forward Euler method. The second method is based on the continuous-time simulation of the deterministic part of the nonlinear system between consecutive times of measurement. For state estimation we apply an unscented Kalman Filter, which—as opposed to the well known Extended Kalman Filter—does not require calculation of the Jacobi matrix of the nonlinear transformation associated with this method.

Keywords Continuous-time stochastic systems · Nonlinear dynamics · Discrete-time measurements · State estimation · Kalman filtering

1 Introduction

The main tool for state estimation of dynamic systems is Kalman filtering [4, 8, 9]. If the system observed is described by continuous-time equations there are two main approaches to state estimation. In the first method one derives discrete-time model describing the process and uses a standard method for estimating it [4, 10], namely the Kalman Filter for a linear approach, or one of its variants for a nonlinear approach: for example the Extended Kalman Filter [2, 4], the Unscented Kalman

M. Domżalski (✉) · Z. Kowalczuk
Gdańsk University of Technology, WETI, Narutowicza 11/12,
80-233 Gdańsk, Poland
e-mail: mardo@eti.pg.gda.pl
url: http://eti.pg.edu.pl

Z. Kowalczuk
e-mail: kova@eti.pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_7

Filter [7, 14], or the Particle Filter [1, 3, 6, 12]. The second method consists in directly utilizing a continuous-time estimator: the Kalman-Bucy Filter [4] in a linear approach or various nonlinear filters [3, 13].

Both approaches have their merits. Implementation of discrete-time filters is straightforward, yet the discrete-time model is only an approximation dependent on the sampling period. Thus another layer of design uncertainty is introduced. On the other hand, implementation of continuous-time filters is more complex (moreover, the measurement equations are also given in continuous time).

For most physical systems the respective continuous-time (CT) description gives best approximation of the actual phenomena which govern the process behavior. On the other hand, it is the discrete-time (DT) measurement process which is typically associated with the principle of operation of sensors. There is therefore a need for a so-called hybrid CT/DT method of state estimation.

In this article we will investigate some hybrid method in which a predictive part of the estimation algorithm is performed by continuous-time simulation of a deterministic part of a stochastic differential equation that describes an analyzed system. In the stochastic differential equations modeling the process, the stochastic part has the zero mean (in the probabilistic sense), thus we assume that it does not contribute to the evolution of the prediction[1]. We will compare this method with a classical one using the standard forward Euler discretization. Though the simulation-based method is computationally more expensive than the Euler method, it gives accurate results independent of the sampling time.

The paper is organized as follows. In Sect. 2 a nonlinear continuous-time stochastic system model and a discrete-time measurement equation are presented. Two methods for discrete-time discretization are described in Sect. 3. The Unscented Kalman filter which is a basis for state estimation is recalled in Sect. 4. A simulation example is presented in Sect. 5. Section 6 contains conclusions.

2 System Model

We consider a dynamic system described by the following nonlinear stochastic differential equation:

$$d\boldsymbol{x}(t) = \boldsymbol{a}(\boldsymbol{x}(t))dt + \boldsymbol{b}(\boldsymbol{x}(t))d\boldsymbol{w}(t), \quad t \in \mathbb{R}^+ = [0, \infty) \tag{1}$$

where an independent variable t is interpreted as time, $d\boldsymbol{x} \in \mathbb{R}^n$ is an infinitesimal increment of the system state $\boldsymbol{x} \in \mathbb{R}^n$, $d\boldsymbol{w} \in \mathbb{R}^r$ is an infinitesimal increment of r-dimensional Wiener process $\boldsymbol{w} \in \mathbb{R}^r$ describing the uncertainty, $\boldsymbol{a} : \mathbb{R}^n \to \mathbb{R}^n$ is

[1]This assumption is only valid for linear models, since in the general case, one can not find an explicit equation describing the evolution of the mean value for a stochastic differential equation. For nonlinear systems this means an approximation whose impact will be studied in future work.

a nonlinear vector function describing the dynamics of the system, and $\boldsymbol{b} : \mathbb{R}^n \to L(\mathbb{R}^r, \mathbb{R}^n)$ is a nonlinear map describing the influence of the noise on the system[2], where $L(\mathbb{R}^r, \mathbb{R}^n)$ is the space of $n \times r$ matrices.

3 Model Discretization

To be able to perform state estimation using Kalman methods, we need to obtain a discrete-time version of the model (1). However, due to the nonlinearity of (1), only approximate methods are applicable. As has been mentioned we will apply two different methods: the forward Euler method and the simulation method.

The well-known forward Euler method results in the following discrete-time version of (1):

$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + T\boldsymbol{a}(\boldsymbol{x}_{\boldsymbol{k-1}}) + \boldsymbol{b}(\boldsymbol{x}_{\boldsymbol{k-1}})\Delta\boldsymbol{w}_{k-1}, \quad k \in \mathbb{N} = \{1, 2, 3, \ldots\} \tag{2}$$

where T is the sampling time (in seconds), $\boldsymbol{x}_k \in \mathbb{R}^n$ is the state of the discrete-time model (2) at time kT, and $\Delta\boldsymbol{w}_k \in \mathbb{R}^r$ is a vector random variable with a multivariate normal distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{Q})$, with the zero mean and the covariance matrix

$$\boldsymbol{Q} = T\boldsymbol{I}_r \tag{3}$$

where $\boldsymbol{I}_r$ is the $r \times r$ identity matrix.

Based on (2), since $\Delta\boldsymbol{w}$ is a zero mean noise, the model-based prediction equation needed for estimation has the following (homogeneous) form:

$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + T\boldsymbol{a}(\boldsymbol{x}_{\boldsymbol{k-1}}), \quad k \in \mathbb{N} \tag{4}$$

The second discretization method is based on simulation of the deterministic part of model (1). In this method the following (homogeneous) deterministic ordinary differential equation is simulated between consecutive sampling instants:

$$\frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{a}(\boldsymbol{x}(t)), \quad \text{for } t \in [(k-1)T, kT], \quad k \in \mathbb{N} \tag{5}$$

A numerical solution to equation (5) can be computed, for example, using one of the standard Runge-Kutta methods, to obtain a prediction of the state x_k for the time instant kT based on the state x_{k-1} computed at the time instant $(k-1)T$.

Besides the fact that the simulation method gives more accurate results of the prediction, it has two drawbacks, as compared to the forward Euler discretization. First, it is computationally more expensive. Second, the simulation method ignores

[2]We assume that maps $\boldsymbol{a}$ and $\boldsymbol{b}$ fulfill the necessary conditions so an appropriate solution to (1) exists for $t \geq 0$.

the random part of the model (1), and thus the noise covariance matrix of the discrete noise is not computed. A simple, yet not faultless solution to this problem is to assume the same noise component $\boldsymbol{b}(\boldsymbol{x_{k-1}})\Delta\boldsymbol{w}_{k-1}$ as for the Euler method (2). Other, more suitable solutions of this problem will be investigated in the future work.

Finally, we supplement both the discrete-time models with a standard discrete-time measurement equation

$$\boldsymbol{y}_k = \boldsymbol{c}(\boldsymbol{x}_k) + \boldsymbol{\zeta}_k \tag{6}$$

where $\boldsymbol{y}_k \in \mathbb{R}^p$ is the measurement vector, $\boldsymbol{c} : \mathbb{R}^n \to \mathbb{R}^p$ is (in general) a nonlinear vector function describing the measurement principle, and $\boldsymbol{\zeta}_k \in \mathbb{R}^p$ is a vector random variable (measurement noise) with a multivariate normal distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$, with the zero mean and a known covariance matrix $\boldsymbol{R}$.

4 State Estimation

To perform state estimation we use the discrete-time model (2)–(6) and an Unscented Kalman Filter described below.

4.1 Unscented Kalman Filter

Calculation of the Jacobi matrix for the nonlinear functions $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ can be a difficult task (for the discrete-time model based on the Euler method) or even not feasible (for the simulation method, since the explicit form of the discrete-time model is not available). To overcome this problem we utilize an approach which does not require the computation of Jacobi matrix—the Unscented Kalman Filter (UKF).

The UKF is based on an unscented transform that is used to determine the mean value and the covariance matrix of a random variable subjected to a nonlinear transformation. In this method, the multivariate normal probability density function of a random variable (before the nonlinear transformation) is represented by a specific, small set of the so-called σ-points. Next, each of these points is transformed using the nonlinear function (the Euler discrete-time model (4)) or each is simulated using the differential equation (5). From the transformed points, the mean value of the random variable and its covariance matrix can next be easily computed.

The UKF has some similar features to particle filters (PF). There are, however, two significant differences described below.

1. The number of the UKF σ-points is not large, and they are chosen 'optimally' so as to best preserve the shape of the multivariate normal distribution, whereas in the particle filter the number of particles is large and they are sampled from some initial distribution. Thus the UKF is less computationally expensive.

2. In the UKF, the initial (pre nonlinear transformation) and the final (post nonlinear transformation) random variables are assumed to have multivariate normal distribution, thus the σ-points (post nonlinear transformation) are 'fitted back' into the multivariate normal distribution, whereas in the PF the particles determine the resulting probability distribution function (it can thus be arbitrary, not normal, multimodal, etc.). Therefore, using UKF it is not possible to model other probability distributions than multivariate normal distribution. It is a significant simplification, as a nonlinear transformation of a normal random variable is generally not normal.

One step of the UKF algorithm for discrete-time models presented in Sect. 2 is described below.

First, based on the results from the previous step, i.e. the state $\boldsymbol{x}_{k-1|k-1}$ and the corresponding covariance matrix $\boldsymbol{P}_{k-1|k-1}$, a new set of σ-points is computed. If the state is an n-dimensional multivariate normal random variable, then this set contains $(2n+1)$ σ-points $\boldsymbol{x}^i_{k-1|k-1}$, $i = 0, 1, 2, \ldots, 2n$ with the corresponding weights W^i, calculated as [7]

$$\begin{aligned}
\boldsymbol{x}^0_{k-1|k-1} &= \hat{\boldsymbol{x}}_{k-1|k-1}, \qquad W^0 \in (-1, 1) \\
\boldsymbol{x}^j_{k-1|k-1} &= \hat{\boldsymbol{x}}_{k-1|k-1} + \left(\sqrt{\frac{n}{1-W^0}\boldsymbol{P}_{k-1|k-1}}\right)^j, \qquad W^j = \frac{1-W^0}{2n} \\
\boldsymbol{x}^{j+n}_{k-1|k-1} &= \hat{\boldsymbol{x}}_{k-1|k-1} - \left(\sqrt{\frac{n}{1-W^0}\boldsymbol{P}_{k-1|k-1}}\right)^j, \qquad W^{j+n} = \frac{1-W^0}{2n} \\
& \qquad j = 1, 2, \ldots, n
\end{aligned} \tag{7}$$

where a new index j is introduced to emphasize the symmetry of the points, the vector

$$\left(\sqrt{\frac{n}{1-W^0}\boldsymbol{P}_{k-1|k-1}}\right)^j$$

is the j-th row (and the j-th column) of the square root (in the matrix sense) of the matrix

$$\frac{n}{1-W^0}\boldsymbol{P}_{k-1|k-1}$$

The value of the weight $W^0 \in (-1, 1)$ controls the position of σ-points. For $W^0 > 0$ the σ-points are further from the central σ-point x^0, whereas for $W^0 < 0$ they are closer to the central σ-point x^0. For a more detailed discussion on the choice of weights refer to [7] and [14]. Note that, naturally, the weights satisfy the relationship $\sum_{i=0}^{2n} W^i = 1$.

Next, the σ-points $\left\{ \boldsymbol{x}^j_{k-1|k-1} \right\}$ are transformed to the new state according to (4) or using (5). In such a way a new set of the transformed points $\left\{ \boldsymbol{x}^i_{k|k-1} \right\}$ is obtained (the weights of the σ-points remain unchanged).

The predicted state $\hat{\boldsymbol{x}}_{k|k-1}$ is computed as the weighted sum of the transformed σ-points

$$\hat{\boldsymbol{x}}_{k|k-1} = \sum_{i=0}^{2n} W^i \boldsymbol{x}^i_{k|k-1} \tag{8}$$

To compute a predicted measurement, the set of the transformed σ-points is transformed again, this time using the nonlinear function adequate for observations (6). This results in the 'measurement' points

$$\boldsymbol{z}^i_{k|k-1} = \boldsymbol{c}\left(\boldsymbol{x}^i_{k|k-1} \right), \quad i = 0, \ldots, 2n \tag{9}$$

The predicted measurement is calculated similarly as the predicted state:

$$\hat{\boldsymbol{z}}_{k|k-1} = \sum_{i=0}^{2n} W^i \boldsymbol{z}^i_{k|k-1} \tag{10}$$

Using the above results we calculate the covariance matrix of the predicted state

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{Q} + \sum_{i=0}^{2n} W^i \left(\boldsymbol{x}^i_{k|k-1} - \hat{\boldsymbol{x}}_{k|k-1} \right) \left(\boldsymbol{x}^i_{k|k-1} - \hat{\boldsymbol{x}}_{k|k-1} \right)^{\mathsf{T}} \tag{11}$$

where $\boldsymbol{Q}$ is the covariance matrix (3), and the covariance matrix of the predicted measurement

$$\boldsymbol{S}_k = \boldsymbol{R} + \sum_{i=0}^{2n} W^i \left(\boldsymbol{z}^i_{k|k-1} - \hat{\boldsymbol{z}}_{k|k-1} \right) \left(\boldsymbol{z}^i_{k|k-1} - \hat{\boldsymbol{z}}_{k|k-1} \right)^{\mathsf{T}} \tag{12}$$

where $\boldsymbol{R}$ is the covariance matrix of the measurement noise $\boldsymbol{\zeta}_k$ in (6).

The gain of the UKF is:

$$\boldsymbol{K}_k = \left[\sum_{i=0}^{2n} W^i \left(\boldsymbol{x}^i_{k|k-1} - \hat{\boldsymbol{x}}_{k|k-1} \right) \left(\boldsymbol{z}^i_{k|k-1} - \hat{\boldsymbol{z}}_{k|k-1} \right)^{\mathsf{T}} \right] \boldsymbol{S}_k^{-1} \tag{13}$$

The UKF innovation is

$$\boldsymbol{v}_k = \boldsymbol{z}_k - \hat{\boldsymbol{z}}_{k|k-1} \tag{14}$$

where $\boldsymbol{z}_k$ is the 'true' measurement collected by a sensor.

The updated state estimate is calculated according to the following equation:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k \boldsymbol{v}_k \tag{15}$$

with the corresponding covariance matrix

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k \boldsymbol{S}_k \boldsymbol{K}_k^{\mathsf{T}} \tag{16}$$

5 Simulation Example

In this section we present an exemplary process model and a simulation example.

5.1 Process Model

We consider a nonlinear continuous-time stochastic model of a mobile robot shown in Fig. 1. The robot has two wheels of radius r, which are connected by axle of an length l. Both wheels can rotate at different speeds, thereby changing the orientation of the robot. We assume that the wheels do not slip and that the robot moves only in the direction of the body orientation. The angle between the body orientation and the x-axis is denoted as θ. The model is based on [5, 11, 15], however we add two additional state variables (ω_1 and ω_2) describing the angular velocities of the wheels.

The position x-y of each wheel ($i = 1, 2$) of the robot evolves according to the equations

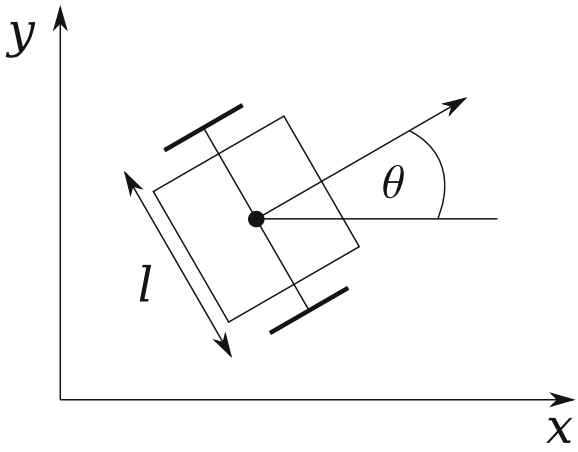


Fig. 1 Mobile robot on the Cartesian plane

$$\begin{aligned} dx_i &= r\cos\theta d\phi_i \\ dy_i &= r\sin\theta d\phi_i \end{aligned} \tag{17}$$

where $(\phi_i, i = 1, 2)$ are the angles by which the wheels rotate about their axes.

Thus the equations describing the motion of the robot are

$$\begin{aligned} dx &= \frac{dx_1 + dx_2}{2} \\ dy &= \frac{dy_1 + dy_2}{2} \\ ld\theta &= rd\phi_1 - rd\phi_2 \end{aligned} \tag{18}$$

where x and y yield the position of the middle point of the axle, l is the length of the axle and θ is the angle of orientation of the robot.

By substituting (17) into (18) one obtains

$$\begin{aligned} dx &= \frac{r\cos\theta d\phi_1 + r\cos\theta d\phi_1}{2} = \frac{r\cos\theta(d\phi_1 + d\phi_1)}{2} \\ dy &= \frac{r\sin\theta d\phi_1 + r\sin\theta d\phi_1}{2} = \frac{r\sin\theta(d\phi_1 + d\phi_1)}{2} \\ d\theta &= \frac{rd\phi_1 - rd\phi_2}{l} \end{aligned} \tag{19}$$

The angles $(\phi_i, i = 1, 2)$ by which the wheels rotate about their axes are described by the following differential equations

$$\begin{aligned} d\phi_1 &= \omega_1(t)dt \\ d\phi_2 &= \omega_2(t)dt \end{aligned} \tag{20}$$

where ω_1 and ω_2 are the angular velocities of the corresponding wheels.

We assume that the angular velocities of the wheels can be described by the following stochastic differential equations:

$$\begin{aligned} d\omega_1 &= \kappa_1(\mu_1 - \omega_1) + \sqrt{D}dw_1 \\ d\omega_2 &= \kappa_2(\mu_2 - \omega_2) + \sqrt{D}dw_2 \end{aligned} \tag{21}$$

where κ_i $(i = 1, 2)$ describes the rate of the ith velocity mean reversion, μ_i is the long-term mean of the ith velocity, dw_1 and dw_2 are infinitesimal increments of two independent one-dimensional Wiener processes, respectively, and $\sqrt{D}$ is a (modeling) noise scaling factor. Using the above equations, the trajectory of the mobile robot can be described by the following set of stochastic differential equations:

$$\begin{bmatrix} dx \\ dy \\ d\theta \\ d\omega_1 \\ d\omega_2 \end{bmatrix} = \begin{bmatrix} r\frac{\omega_1(t)+\omega_2(t)}{2}\cos\theta(t) \\ r\frac{\omega_1(t)+\omega_2(t)}{2}\sin\theta(t) \\ \frac{r}{l}(\omega_1(t)-\omega_2(t)) \\ \kappa_1(\mu_1-\omega_1) \\ \kappa_2(\mu_2-\omega_2) \end{bmatrix} dt + \sqrt{D}\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} dw_1 \\ dw_2 \end{bmatrix} \tag{22}$$

where the state of the robot is its position (x, y), the angle of the body orientation θ and the angular velocities of the wheels (ω_1, ω_2).

We assume that the position (x, y) and the angle θ are measured. Therefore the map $\boldsymbol{c}(\boldsymbol{x}_k)$ in (6) is linear and is independent of the state x_k, and can be consequently described by the following matrix:

$$\boldsymbol{c}(\boldsymbol{x}_k) = \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{23}$$

The covariance matrix of the measurement noise ζ is

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.05 \end{bmatrix} \tag{24}$$

The initial state is

$$x_0 = \left[0.0,\, 0.0,\, \pi/4,\, 1.0,\, 1.0\right]^{\mathsf{T}} \tag{25}$$

and the other parameters of the model are as follows:

$$\begin{aligned} D &= 0.04 \\ r &= 1, & l &= 4 \\ \kappa_1 &= 0.01, & \kappa_2 &= 0.01 \\ \mu_1 &= 2.0, & \mu_2 &= 2.4 \end{aligned} \tag{26}$$

The process was simulated for $0 \le t \le 50$, and 500 trajectories were generated. The estimation results represent the average of the results obtained in all the simulation runs.

An exemplary trajectory of the robot positions (x, y) on the plane with the corresponding measurements is presented in Fig. 2.

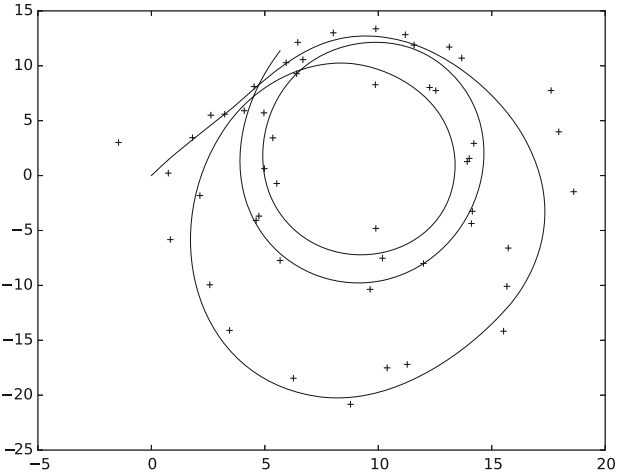


Fig. 2 Exemplary trajectory (x, y) of the robot with the corresponding measurements

5.2 Estimation Results

The outcomes[3] of the estimation of position x, position y, angle θ and the angular velocity ω_1, for both methods are presented in Figs. 3, 4, 5 and 6, respectively. The results for the forward Euler method are marked with dashed lines and the results for the simulation-based method are denoted by solid lines.

The observed correlation between the errors of both methods results from the fact that the two estimators were tested for the same set of 500 trajectories.

Based on the presented example we can conclude that the estimation errors for the simulation-based approach are smaller than the errors gained with the forward Euler method. Clearly, the resulting difference in performance depends on the process under estimation and the sampling period T. Nevertheless, the estimates for the angular velocity are almost the same for both methods. This is because the velocities were modeled as the Wiener processes, for which better prediction has no effect.

[3] As the results concerning the velocity ω_2 are almost identical to the results for ω_1, they are not included here.

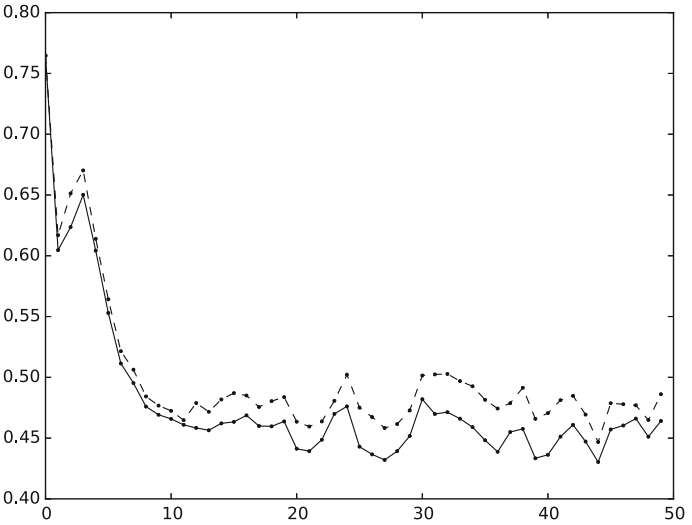


Fig. 3 Position x estimation errors obtained from: the forward Euler method (*dashed line*) and the simulation-based method (*solid line*)

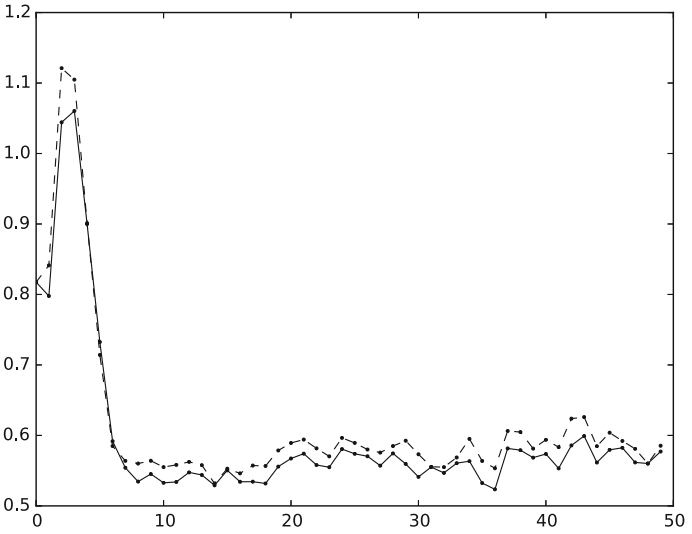


Fig. 4 Position y estimation errors obtained from: the forward Euler method (*dashed line*) and the simulation-based method (*solid line*)

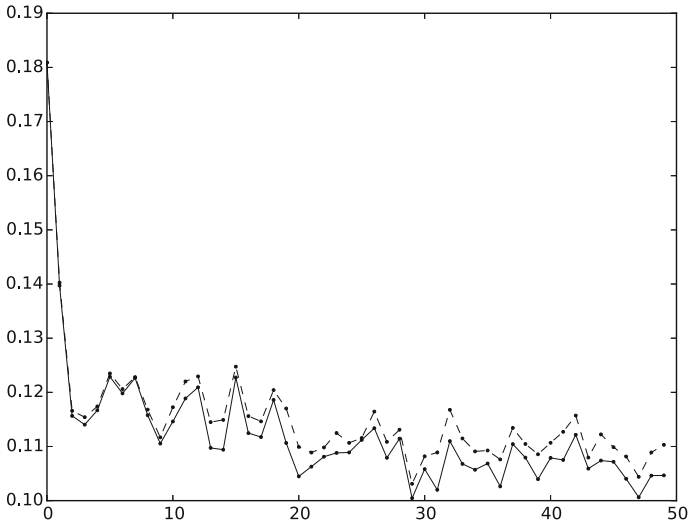


Fig. 5 Angle θ estimation errors obtained from: the forward Euler method (*dashed line*) and the simulation-based method (*solid line*)

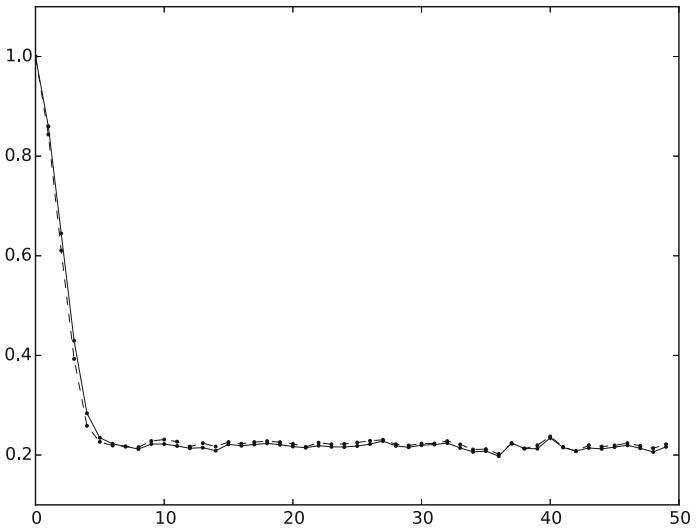


Fig. 6 Angular velocity ω_1 estimation errors obtained from: the forward Euler method (*dashed line*) and the simulation-based method (*solid line*)

6 Conclusions

To draw a general conclusion about the performance quality of both methods, an analytic method for determining bounds on the errors is needed. For the Euler method, the one-step local error is of the order $O(T^2)$. Thus, for example, a two times smaller sampling period leads to a four times smaller prediction error, approximately. Whereas in the simulation-based method the local error can be made arbitrary small, depending on the chosen integration step, which can be much smaller than T. With proper optimization using adaptive methods of integration of ordinary differential equations (eg. ODE23, ODE45), the increase in the computational cost of the simulation-based method need not to be high. Moreover, the simulation-based method yields an additional degree of freedom. By choosing a method of integration and its parameters one can trade-off between better estimation performance and lower computational costs.

References

1. Arulampalam, S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online non-linear/non-gaussian Bayesian tracki ng. IEEE Trans. Signal Process. **50**(2), 174–188 (2002)
2. Athans, M., Wishner, R., Bertolini, A.: Suboptimal state estimation for continuous-time nonlinear systems from discrete noisy measurements. IEEE Trans. Autom. Control **13**, 504–514 (1968)
3. Bain, A., Crisan, D.: Fundamentals of Stochastic Filtering. Springer, New York (2009)
4. Bar-Shalom, Y., Li, X.R.: Estimation and Tracking: Principles, Techniques, and Software. Artech House, Boston (1993)
5. Chirikjian, G.S.: Stochastic Models, Information Theory, and Lie Groups. Birkhauser, Boston, USA (2009)
6. Gordon, N., Salmond, D., Smith, A.: Novel approach to nonlinear/non-gaussian bayesian state estimation. IEEE Proc. F, Radar Signal Process. **140**(2), 107–113 (1993)
7. Julier, S., Uhlmann, J.: A new extension of the Kalman Filter to nonlinear systems. In: Proceedings of the 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls (1997)
8. Kalman, R.: A New Approach to Linear Filtering and Prediction Problems. J. Basic Eng. Trans. ASME'82 (Series D), 34–45 (March 1960)
9. Kay, S.: Fundamentals of Statistical Signal Processing: Estimation Theory, vol. I. Prentice Hall, Englewood Cliffs (1993)
10. Kowalczuk, Z., Domżalski, M.: Optimal asynchronous estimation of 2D Gaussian-Markov processes. Int. J. Syst. Sci. **43**(8), 1431–1440 (2012)
11. Long, A.W., Wolfe, K.C., Mashner, M.J., Chirikjian, G.S.: The Banana Distribution is Gaussian: A Localization Study with Exponential Coordinates, pp. 265–272. MIT (2013)
12. Ristic, B., Arulampalam, S., Gordon, N.: Beyond the Kalman Filter: Particle Filters for Tracking Applications. Artech House, Boston (2004)
13. Sarkka, S.: On unscented Kalman filtering for state estimation of continuous-time nonlinear systems. IEEE Trans. Autom. Control **52**(9), 1631–1641 (2007)

14. Wan, E., van der Merwe, R.: The unscented Kalman filter for nonlinear estimation. In: The IEEE 2000 adaptive systems for signal processing, communications, and control symposium AS-SPCC. Lake Louise, Alberta, USA (October 2000)
15. Zhou, Y., Chirikjian, G.S.: Probabilistic models of dead-reckoning error in nonholonomic mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'03, Taipei, Taiwan (September 2003)

Identification of Models and Signals Robust to Occasional Outliers

Janusz Kozłowski and Zdzisław Kowalczuk

Abstract In this paper estimation algorithms derived in the sense of the least sum of absolute errors are considered for the purpose of identification of models and signals. In particular, off-line and approximate on-line estimation schemes discussed in the work are aimed at both assessing the coefficients of discrete-time stationary models and tracking the evolution of time-variant characteristics of monitored signals. What is interesting, the procedures resulting from minimization of absolute-error criteria appear to be insensitive to sporadic outliers in the processed data. With this fundamental property the deliberated absolute-error method provides correct results of identification, while the classical least-squares estimation produces outcomes, which are definitely unreliable in such circumstances. The quality of estimation and the robustness of the discussed identification procedures to occasional measurement faults are demonstrated in a few practical numerical tests.

Keywords Non-stationary systems · Discrete-time models · Parameter estimation · Least-squares estimator · Least absolute-error estimator

1 Introduction

In many industrial automation systems the measurement data are often corrupted by perturbations or even destructive disturbances. The influence of high-frequency additive noises can be partly eliminated using simple low-pass filters. In case of the

J. Kozłowski (✉) · Z. Kowalczuk
Faculty of Electronics, Telecommunications and Informatics,
Department of Robotics and Decision Systems,
Gdańsk University of Technology, Gdańsk, Poland
e-mail: jk23@eti.pg.gda.pl

Z. Kowalczuk
e-mail: kova@eti.pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_8

so-called systematic errors, in turn, proper calibration of sensors and employment of dedicated compensation techniques allow us to overcome the bias problem. Unfortunately, with sporadic faults in measurement data, commonly referred to as outliers, the classical LS (least-squares) estimates of the parameters of identified models are most often unreliable. In order to cure this problem, new estimation algorithms synthesized in the sense of the least sum of absolute errors (LA) that are robust to such outliers, can be put into practice. What is more, certain mechanisms of weighting, or forgetting, can be implemented in these procedures, for the purpose of obtaining on-line tracking of time-variant parameters of observed non-stationary processes.

The dynamics of supervised industrial objects can be modeled using continuous-time or discrete-time representations (like transfer functions, state space descriptions, etc.). In the case of continuous models involved approximation techniques have to be used to numerically implement the identification schemes, but the resultant estimates of 'physical' parameters (in definite units) usually supply pertinent information about the monitored processes. On the other hand, identification of easily implementable discrete-time models is straightforward, but the values of the obtained 'mathematical' coefficients have no physical interpretation and depend on the sampling frequency. In conclusion, it can be stated that continuous models should be preferred in situations, when the identified parameters themselves hold relevant diagnostic information. If, however, the parameter estimates are supposed to be temporary results necessary for evaluation of some aggregate quantities (e.g. correlation functions, probability distributions, spectral densities), simple discrete-time modeling is by all means justified.

In the subsequent sections the following issues are discussed. First, classical algebraic and recursive realizations of the weighted least-squares algorithm are recalled in Sect. 2. And next, a detailed presentation of the identification approach resulting from minimization of absolute-error criteria is given. In Sect. 3 few practical examples are used to demonstrate the robustness of the LA estimator to sporadic outliers in the data processed. Ultimately, in Sect. 4, final conclusions are drawn and hints for further investigations in the area of fault-tolerant identification are given.

2 LS and LA Estimation Algorithms

In this study the dynamics of a supervised industrial process, or the evolution of a monitored signal, is modeled using the common regression description

$$\gamma(k) = \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \boldsymbol{\theta} + e(k) \tag{1}$$

$$\boldsymbol{\varphi}(k) = [\,\varphi_1(k) \quad \varphi_2(k) \quad \ldots \quad \varphi_n(k)\,]^{\mathrm{T}} \tag{2}$$

$$\boldsymbol{\theta} = [\,\theta_1 \quad \theta_2 \quad \ldots \quad \theta_n\,]^{\mathrm{T}} \tag{3}$$

where $\gamma\,(k)$ stands for a reference signal, $e(k)$ represents an equation (prediction) error, while $\boldsymbol{\varphi}(k)$ and $\boldsymbol{\theta}$ contain certain regression data and process parameters, respectively. The classical LS estimator is obtained directly from minimization of the well-known quadratic criterion [3]

$$V_{\mathrm{LS}}(\boldsymbol{\theta}) = \sum_{l=1}^{k} w_l [e(l)]^2 = \sum_{l=1}^{k} w_l [\gamma(l) - \varphi^{\mathrm{T}}(l)\boldsymbol{\theta}]^2 \tag{4}$$

where the introduced non-negative ($w(k-l) = w_l \geq 0$) and non-increasing ($w_l \geq w_{l+1}$) sequence of weights w_l satisfies the following 'finite memory' condition: $\sum w_l < \infty$ for $l = -\infty, \ldots, k$. Among different shapes of such a forgetting window the exponential profile is probably most convenient for practical implementations

$$w_l = w(k-l) = \lambda^{k-l} \tag{5}$$

with the forgetting factor λ usually set within the range of [0.9¸ 1]. The effective number of observations, also referred to as the memory length of the estimator, can be computed as $\Gamma = 1/(1 - \lambda)$.

Analytical minimization of the quality index (4) results in the following algebraic form of the exponentially weighted LS procedure [3]

$$\boldsymbol{\theta}(k) = \left[\sum_{l=1}^{k} \lambda^{k-l} \boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)\right]^{-1} \left[\sum_{l=1}^{k} \lambda^{k-l} \boldsymbol{\varphi}(l)\gamma(l)\right] \tag{6}$$

It is easy to demonstrate that the non-weighted LS estimator (for $\lambda = 1$) generates asymptotically consistent/unbiased estimates of (3) provided the residual process $e\,(k)$ in (1) presents zero-mean white noise.

The algebraic LS estimator (6) suffers from the inconvenience of having to invert the so-called information matrix. In order to overcome this issue the expressions in brackets are rewritten into their recursive forms and the 'matrix inversion lemma' is applied. As a consequence, an equivalent recursive scheme involving the calculation of the prior prediction error $\varepsilon(k)$, evaluation of the covariance matrix $\mathbf{P}(k)$ and update of the estimates of $\boldsymbol{\theta}$ can be described as [3]

$$\varepsilon(k) = \gamma(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \boldsymbol{\theta}(k-1) \tag{7}$$

$$\mathbf{P}(k) = \frac{1}{\lambda}\left[\mathbf{P}(k-1) - \frac{\mathbf{P}(k-1)\ \boldsymbol{\varphi}(k)\ \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \mathbf{P}(k-1)}{\lambda + \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \mathbf{P}(k-1)\ \boldsymbol{\varphi}(k)}\right] \tag{8}$$

$$\boldsymbol{\theta}(k) = \boldsymbol{\theta}(k-1) + \mathbf{P}(k)\ \boldsymbol{\varphi}(k)\ \varepsilon(k) \tag{9}$$

For numerical reasons, it is recommended to initiate the recursive LS scheme (7)–(9) with a huge diagonal covariance matrix, e.g. $\mathbf{P}(0) = \text{diag}\ (10^5, \ldots, 10^5)$.

Both the algebraic and recursive LS procedures are highly sensitive to outliers in processed data. As a result, the appearance of such faults makes the computed estimates of $\boldsymbol{\theta}$ completely unreliable.

The above-described minimization of the squared errors index (4) is analytically simple, and the quadratic criterion expresses a loss of energy behind the system/diagnostics based on parametric identification. There are, however, practical situations in which such an 'energetic interpretation' is not suitable, for instance, in many problems related to market and economy, where a more balanced approach involving absolute-error criteria appears to be more adequate (like in the evaluation of profits or losses in the stock market). What is more, algorithms synthetized using the Least Absolute (LA) rule are known to be robust to occasional outliers in measurement data. This fundamental gain of the LA approach is attributed to equal emphasis paid to all observations. Note that, on the contrary, the impact of outliers is magnified due to the higher contribution of large errors in the LS method.

The criterion penalizing the errors in the absolute way, along with its appropriate approximation, can be written down as follows [1]:

$$V_{\text{LA}}(\boldsymbol{\theta}) = \sum_{l=1}^{k} w_l |e(l)| \approx \sum_{l=1}^{k} w_l \frac{[\gamma(l) - \boldsymbol{\varphi}^{\text{T}}(l)\ \boldsymbol{\theta}]^2}{|\hat{e}(l)|} \tag{10}$$

where an estimate of the prediction error $e(l)$ is assumed to be available (from a running-in parallel auxiliary estimator, for instance).

Analytical minimization of the above 'quasi-quadratic' index (10) directly leads to an approximate LA estimate of the parameters (3)

$$\boldsymbol{\theta}(k) = \left[\sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\,\boldsymbol{\varphi}^{\text{T}}(l)}{|\hat{e}(l)|}\right]^{-1} \left[\sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\,\gamma(l)}{|\hat{e}(l)|}\right] \tag{11}$$

with the weighting mechanism $w(l)$ represented again by the exponential window (5).

The 'rough' result used in (11) can be further improved by using a specific iterative processing of the regression data. Namely, an approximation of the prediction error $e(k) = \gamma(k) - \boldsymbol{\varphi}^{\text{T}}(k)\ \boldsymbol{\theta}$ can be found using the recent estimates of the parameter vector $\boldsymbol{\theta}$. Such repeated calculations result in the following procedure of successive approximations ($p = 0, 1, \ldots$) [1, 5]:

$$\hat{e}^{/p/}(l) = \gamma(l) - \boldsymbol{\varphi}^{\text{T}}(l)\ \boldsymbol{\theta}^{/p/} \tag{12}$$

$$\boldsymbol{\theta}^{/p+1/} \approx \left[\sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\text{T}}(l)}{|\hat{e}^{/p/}(l)|}\right]^{-1} \left[\sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\gamma(l)}{|\hat{e}^{/p/}(l)|}\right] \tag{13}$$

where the start-up value of $\boldsymbol{\theta}$ (i.e. for $p = 0$) is obtained from the auxiliary LS scheme.

The processing (12)–(13) is terminated, if further improvement in the estimates (resulting from the computed values of the quality index (10)) is negligibly small, i.e.

$$\left|V_{\mathrm{LA}}(\boldsymbol{\theta}^{\langle p+1\rangle}) - V_{\mathrm{LA}}(\boldsymbol{\theta}^{\langle p\rangle})\right| < \Delta_{\min} \tag{14}$$

The above terminal condition (14) using a threshold criterion value $\Delta_{\min}$ is justified, since the index (10) is always decreasing in the successive iterations (12)–(13). The proof of this fundamental property can be found in [5].

It is important to take into consideration the characteristic form of the quality index (10). Namely, one should take into account that the minima of the absolute-error criteria are usually located at the points (the so-called 'peaks'), where the criterion function is not differentiable. This implies that in consecutive steps of the algorithm (12)–(13), one of the errors (12) is certain to tend to zero, so the problem of small divisors is sure to appear in (13).

The core problem of the numerical conditioning of the iterative procedure (12)–(13) can be partly overcome by replacing close-to-zero absolute values of the errors (12) with an assumed fixed thresholding error value $\varepsilon_{\min}$ ($\varepsilon_{\min} > 0$). The suggested hint, commonly referred to as regularization, can be found in many numerical algorithms.

Assuming that processing (12)–(13) is bounded to a single iteration only, an approximate recursive version of the LA algorithm can easily be obtained. Again, by employing the 'matrix inversion lemma' to handle (11), the ultimate procedure with the calculation of the prediction error, update of the covariance matrix, and correction of estimates follows [4] immediately

$$\varepsilon(k) = \gamma(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \boldsymbol{\theta}(k-1) \tag{15}$$

$$\mathbf{P}(k) = \frac{1}{\lambda}\left[\mathbf{P}(k-1) - \frac{\mathbf{P}(k-1)\ \boldsymbol{\varphi}(k)\ \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \mathbf{P}(k-1)}{\lambda|\varepsilon(k)| + \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \mathbf{P}(k-1)\ \boldsymbol{\varphi}(k)}\right] \tag{16}$$

$$\boldsymbol{\theta}(k) = \boldsymbol{\theta}(k-1) + \mathbf{P}(k)\ \boldsymbol{\varphi}(k)\mathrm{sgn}[\varepsilon(k)] \tag{17}$$

In a precise initiation of the recursive LA procedure the processing (12)–(13) can be limited to a single stage yielding approximate results for the sampled-time process LA in its early instants ($1 \ \ldots \ k_0$). Eventually, starting from $k = k_0 + 1$, the covariance matrix (the bracketed expression being inverted) and the estimates themselves (13) can be immediately applied in the target LA scheme (15)–(17). One should be aware, however, that the proposed start-up routine (13) involving the matrix inversion is, in general, numerically inconvenient.

In order to facilitate this initiation, the LS algorithm (7)–(9) is normally used in a start-up phase of the recursive LA scheme, and the LS results obtained in the

sampling instants $1 \ldots k_0$ are applied as temporary results of the LA identification. Then, beginning with the next sampling moment $k = k_0 + 1$, the recent LS covariance matrix (8) and the parameter estimates (9) are utilized in the target LA procedure (15)–(17). This simplified solution is justified for the weighted estimators only ($\lambda < 1$), since for the finite memory ($\Gamma = 1 / (1 - \lambda)$) the exponentially forgotten 'rough' initial data do not influence the future results of identification.

The above-presented iterative and recursive LA algorithms will be subsequently used in numerical experiments. The obtained results of practical tests should illustrate the robustness of these procedures to occasional outliers in measurement data. Yet, the employed forgetting mechanism allows us to effectively track time-variant parameters of the identified models.

3 Estimation Methods in Numerical Experiments

In this section practical implementation of the estimation procedures will be presented. First, linear regressions in the LS and LA sense will be used in identification of basic physical models. Next, identification of parameters of an autoregressive (AR) process will be shown in a convenient evaluation of the power spectral density function. Finally, recursive estimators will be used to isolate and track changes in component frequencies of a multi-harmonic signal. It is important that all the subsequent examples can easily be related to practical issues found in industrial and medical diagnostics. Other interesting examples in this area can be found in [5–7].

3.1 Linear Regression

Linear approximation of measurement data (referred to as linear regression) is a fundamental method, in which definite linear models are adjusted using estimation procedures. Despite the fact that the method is straightforward, it finds many applications in experimental determination of the parameters of physical (white-box) models. This is so, because linear equations are often used to describe the nature of physical phenomena. In case of piezoelectricity, for example, the electric charge accumulated in crystalline materials increases proportionally to the value of the applied mechanical stress. It is also well known that static friction grows linearly in response to the applied normal force. On a similar basis, the deformation of Hookean materials is proportional to the generated stress.

In the first performed numerical test a simple linear model

$$y(t) = a_1 t + a_2 \tag{18}$$

with the design parameters a_1 and a_2 is fitted to the set of discrete-time measurements $y(k)$. Obviously, Eq. (18) can be written down in the common regression form

$$y(k) = \gamma(k) = \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \boldsymbol{\theta} + e(k) \tag{19}$$

$$\boldsymbol{\varphi}(k) = [\,k \quad 1\,]^{\mathrm{T}} \tag{20}$$

$$\boldsymbol{\theta} = [\,a_1 \quad a_2\,]^{\mathrm{T}} \tag{21}$$

where $e(k)$ is the residual error and k is used to index the data samples.

The variance of the corrupting normally-distributed white noise sequence $e(k)$ has been assumed as $\sigma_e^2 = 6.4 \times 10^{-3}$. The non-weighted ($\lambda = 1$) algebraic LS and iterative LA ($\Delta_{\min} = 1.0 \times 10^{-4}$, $\varepsilon_{\min} = 1.0 \times 10^{-12}$) estimation schemes were used to evaluate the parameter vector (21). Sporadic outliers were also inserted into the processed data (as $y(2) = 5$, $y(7) = 4$, $y(9) = 6$). The exactness of linear approximation is shown in Fig. 1.

It is evident that irrespective of the outliers in data, the true parameters $a_1 = 1$ and $a_2 = 0$ are evaluated reliably ($a_1 = 0.9784$, $a_2 = 0.1147$) using the LA method. The LS results ($a_1 = 0.6945$, $a_2 = 1.4226$), in turn, appear to be greatly influenced by faults.

One should realize that the same, basic linear regression can be easily applied for identification of specific exponential models of the following type:

$$y(t) = \beta\ \exp(\mu t) \tag{22}$$

This issue is practical, because the function (22) is suitable to describe various well-known physical laws. In the Newton's formula of cooling, for instance, the

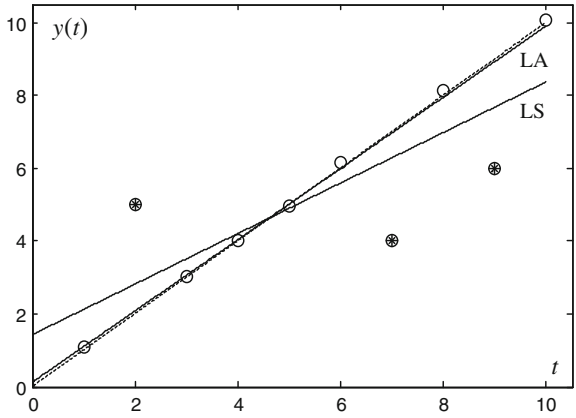


Fig. 1. The LS and LA approximation of measurement data (o) in the presence of outliers (*), where the *dotted line* represents the theoretical linear model

temperature of a body decays exponentially in time. The Beer-Lambert-Bouguer law of absorption, in turn, introduces such exponential relation between the light intensity and the thickness of the material the light is traveling through. The 'current-voltage' characteristics of semiconductors (e.g. diodes) can also be modeled using (22).

By performing logarithmic transformation on both sides of (22), the regression equation of (19)–(21) can be employed with the parameters $a_1 = \mu$, $a_2 = \ln \beta$, and the reference signal described by $\gamma\ (k) = \ln\ [\ y(k)]$.

3.2 Approximation of the Spectral Density Function

There is a wide spectrum of medical applications, where classical autoregressive representations are used to model vital experimental data. Among others, the electroencephalography (EEG) is a fundamental health check supported by biomedical engineering. In this case, the modeling of physiological signals, subject to electrical activity along the patient's scalp, delivers important information about possible brain disorders. Since the values of the identified AR coefficients cannot be directly applied to elaborate the diagnosis, post-calculated aggregate indices, like the power spectral density function, are usually considered in associated medical check-ups.

In the subsequent example, an AR model is used to imitate EEG recordings

$$y(k) + a_1 y(k-1) + \cdots + a_n y(k-n) = e(k) \tag{23}$$

The Eq. (23) assumes the common regression form

$$y(k) = \gamma(k) = \boldsymbol{\varphi}^{\mathrm{T}}(k)\ \boldsymbol{\theta} + e(k) \tag{24}$$

$$\boldsymbol{\varphi}(k) = [\ -y(k-1) \quad \ldots \quad -y(k-n)\ ]^{\mathrm{T}} \tag{25}$$

$$\boldsymbol{\theta} = [\ a_1 \quad \ldots \quad a_n\ ]^{\mathrm{T}} \tag{26}$$

where, by definition, the residual error $e(k)$ is a zero-mean white noise sequence.

Also in this experiment the non-weighted ($\lambda = 1$) algebraic LS and iterative LA ($\Delta_{\min} = 1.0 \times 10^{-4}$, $\varepsilon_{\min} = 1.0 \times 10^{-12}$) routines were applied to evaluate the parameters ($a_1 = -2.1552$, $a_2 = 3.2399$, $a_3 = -3.3426$, $a_4 = 2.4304$, $a_5 = -1.2056$, $a_6 = 0.4032$) of the simulated AR process ($n = 6$). The output sequence was generated using the normally distributed white noise ($\sigma_e^2 = 1$), and the identification was based on data records with $M = 1000$ samples and simulated 'zero' faults as $y\ (k) = 0$ for $k = 801 \ldots 880$.

Finally, the spectral density function was estimated using the parametric approach:

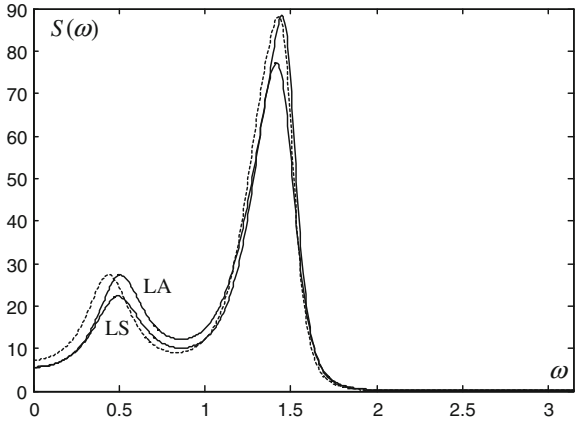


Fig. 2. The LS and LA approximation of the power spectral density function, with the dotted line representing the theoretical result

$$\hat{S}(\omega) = \frac{\hat{\sigma}_e^2}{|1 + \hat{a}_1 \exp(-j\omega) + \cdots + \hat{a}_n \exp(-j\omega n)|^2} \tag{27}$$

It is clear that the LS estimate (27) is influenced by faults. Contrary to this, the LA method generates an acceptable result irrespective of the losses in data (Fig. 2).

Unfortunately, in both methods the following measure of the variance σ_e^2 of $e(k)$:

$$\hat{\sigma}_e^2 = \frac{1}{M-n} \sum_{k=n+1}^{M} [y(k) + \hat{a}_1 y(k-1) + \cdots + \hat{a}_n y(k-n)]^2 \tag{28}$$

is corrupted by occasional faults. In the case of the LA method, however, this problem is meaningless. This is so, because (28) is a scaling factor only, while with the reliable LA estimates ($a_1 = -2.0708$, $a_2 = 3.0980$, $a_3 = -3.1204$, $a_4 = 2.2701$, $a_5 = -1.0939$, $a_6 = 0.3699$) the shape of the spectral density function (27) is evaluated properly (e.g. its resonance 'peaks'). Evidently, application of erroneous LS estimates ($a_1 = -1.9972$, $a_2 = 2.9320$, $a_3 = -2.9099$, $a_4 = 2.0767$, $a_5 = -0.9970$, $a_6 = 0.3444$) leads to a less accurate evaluation of $S(\omega)$.

3.3 Tracking the Frequencies of Multi-Harmonic Signals

There are applications, where diagnostics of monitored signals is based on isolation of particular sinusoids from a composed multi-harmonic signal. This issue is relevant, for instance, in the evaluation of the quality of electrical grids. Theoretically, in the classical AC (alternating current) power systems, the observed voltage should be varying in a sinusoidal way at an assumed frequency (e.g. $f = 50$ Hz). Due to possible non-linear electric loads, the existent higher order harmonics (i.e. $2f$, $3f$, …) often cause problems in the power system. In order to evaluate the distortion

induced by such harmonics, the analysis involving the Fourier series is usually put into practice. Effective applications of this standard solution are possible, provided the fundamental frequency is precisely known and invariant. Unfortunately, slow drifts of the respective frequency are common in the monitored voltages. Hence, isolation and tracking the component frequencies contained in multi-harmonic signals become important.

Consider a multi-harmonic signal containing two sinusoids

$$y(t) = A_1 \sin \omega_1 t + B_1 \cos \omega_1 t + A_2 \sin \omega_2 t + B_2 \cos \omega_2 t \tag{29}$$

where ω_1 and ω_2 stand for the respective angular frequencies. It is easy to verify that (29) satisfies the following differential equation:

$$\frac{d^4 y(t)}{dt^4} + (\omega_1^2 + \omega_2^2)\, \frac{d^2 y(t)}{dt^2} + (\omega_1^2 \omega_2^2)\, y(t) = 0 \tag{30}$$

Overall, a discrete-time mechanization of an ordinary (nth order) differential equation can be obtained using the following finite-horizon integrating filter [2, 8]:

$$J_r^n x(t) = \int_{t-h}^{t} \int_{t_1-h}^{t_1} \dots \int_{t_{n-1}-h}^{t_{n-1}} x^{(r)}(t_n) dt_n dt_{n-1} \dots dt_1 \tag{31}$$

with r ($0 \le r \le n$) denoting the order of differentiation of the transformed signal and h ($h > 0$) representing the length of the integration horizon. Numerical implementation of (31) can be acquired by employing the bilinear (Tustin's) operator for the evaluation of multiple integrals [8]. As a result, the following simple integrating filter of the finite impulse response (FIR) type establishes a discrete counterpart of (31):

$$J_r^n x(t)\,\big|_{t=kT} \approx I_r^n(q^{-1}) x(kT) = \left(\frac{T}{2}\right)^{n-r} (1+q^{-1})^{n-r} (1-q^{-1})^r \left(1 + \sum_{i=1}^{L-1} q^{-i}\right)^n x(kT) \tag{32}$$

where the operator q^{-1} represents the backward shift ($q^{-1}\, x(kT) = x(kT - T)$) of the sampled signal $x(kT)$, T is the sampling period, and $L = h / T$ defines the horizon length in terms of the number of samples falling in the integration window.

The above numerical integration ($n = 4$) performed equally on both sides of the differential Eq. (30) leads to a discrete-time model retaining the system parameters

$$I_4^4(q^{-1}) y(k) = \gamma(k) = \boldsymbol{\varphi}^{\mathrm{T}}(k) \boldsymbol{\theta} + e(k) \tag{33}$$

$$\boldsymbol{\varphi}(k) = [\, -I_2^4(q^{-1}) y(k) \quad -I_0^4(q^{-1}) y(k)\,]^{\mathrm{T}} \tag{34}$$

$$\boldsymbol{\theta} = [\, a_2 \quad a_0 \,]^{\mathrm{T}} \tag{35}$$

where $a_2 = \omega_1^2 + \omega_2^2$, $a_0 = \omega_1^2\, \omega_2^2$ and, for notational brevity, the sampling moment kT is shorthanded as k.

Ultimately, the estimates of the isolated frequencies ω_1 and ω_2 can be calculated based on the identified values of the auxiliary parameters (35)

$$\hat{\omega}_{1,2} = \sqrt{\frac{\hat{a}_2 \mp \sqrt{\hat{a}_2^2 - 4\hat{a}_0}}{2}} \tag{36}$$

The exponentially weighted ($\lambda = 0.999$) LS and LA algorithms in their recursive forms were applied to track gradual changes of frequencies represented in the multi-harmonic signal (29): $\omega_1 = 0.5 \ldots 0.48$ and $\omega_2 = 1 \ldots 0.96$. The normally distributed additive white noise ($\sigma_v^2 = 1.6 \times 10^{-3}$) corrupted the discrete-time data $y(k)$, that is identification was based on $y(k) + v(k)$, and a sequence of 'zero' faults appeared in the measurements ($y(k) = 0$ for $k = 7001 \ldots 7025$, within the interval 350 s $< t <$ 351.25 s). In the discrete-time mechanization (32) of the model (30) the integration horizon was fixed as $h = 1.75$ s ($L = 35$), at the sampling time $T = 0.05$ s. The analyzed multi-harmonic signal was monitored for 600 s (12,000 sampling instants).

The performed tests confirm the former conclusion that the LA method (also in its approximate recursive realization) is robust to faults in data (Fig. 4). The influence of such measurement errors is clearly visible in the LS estimates (Fig. 3).

It is instructive to see that the presented reasoning can easily be generalized for the case of more complex harmonic signals than (29)—with m sinusoids ($\omega_1, \omega_2, \ldots, \omega_m$), the resultant regression equation (with $n = 2$ m) is given by

$$I_n^n(q^{-1})y(k) = \gamma(k) = \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta} + e(k) \tag{37}$$

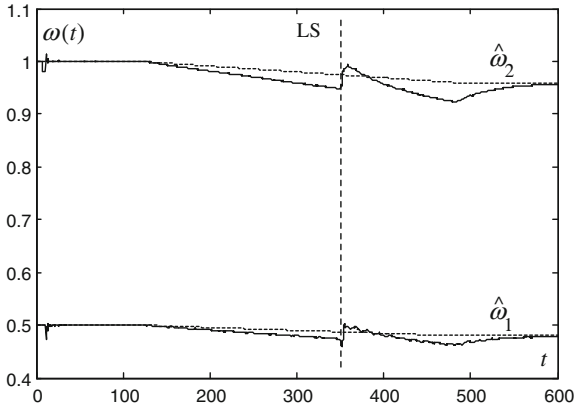


Fig. 3. Tracking the changes in frequencies of a multi-harmonic signal using the LS method (the moment of fault appearance is indicated by the *vertical line*)

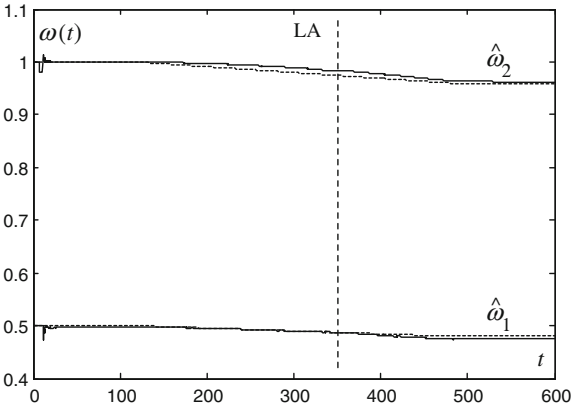


Fig. 4. Tracking the changes in frequencies of a multi-harmonic signal using the LA method (the moment of fault appearance is indicated by the *vertical line*)

$$\boldsymbol{\varphi}(k) = [\, -I^n_{n-2}(q^{-1})y(k) \quad -I^n_{n-4}(q^{-1})y(k) \quad \ldots \quad -I^n_0(q^{-1})y(k)\,]^{\mathrm{T}} \tag{38}$$

$$\boldsymbol{\theta} = [\, a_{n-2} \quad a_{n-4} \quad \ldots \quad a_0\,]^{\mathrm{T}} \tag{39}$$

and the estimates of frequencies are equal to positive solutions of the equation

$$\omega^n - \hat{a}_{n-2}\omega^{n-2} + \hat{a}_{n-4}\omega^{n-4} + \cdots + (-1)^{n/2}\hat{a}_0 = 0 \tag{40}$$

It is also worth noticing that we can suggest an alternative solution—without employment of the FIR transformation (32) of the original representation. Namely, instead of (30) a simple difference equation can be used to describe the evolution of the already sampled signal (29). Indeed, the totally discrete approach is numerically convenient, but the evaluated parameters are then expressed as cosine functions of the target frequencies (i.e. $a_1 = \cos\,\omega_1 T$, $a_2 = \cos\,\omega_2 T$). It is thus evident that tracking the changes of such quantities becomes problematic: e.g. for $\omega_1 = 0.5 \ldots 0.48$ the associated coefficient $\cos\,\omega_1 T$ falls within a very narrow range [0.99969, 0.99971].

In conclusion, the continuous-time approach used for estimation of the parameters (35) directly related to the target frequencies represented in (29), is recommended.

4 Conclusions and Further Investigations

In this study the fundamental properties of LA procedures have been discussed and verified in numerical tests. As has been demonstrated, the LA schemes are robust to occasional outliers in the processed data, while the performance of the classical LS routines is heavily disrupted by such faults.

The developed realizations of the LA estimator appear to be conceptually simple, and the approximate recursive LA algorithm proves to be effective in diagnostics of harmonic signals. Moreover, possible weighting mechanisms allow us to track parameters of non-stationary processes.

It has also to be taken into consideration that the obtained LS and LA estimates are consistent provided that the residual error $e(k)$ is zero-mean white noise. In practical situations, however, this assumption can often be violated (e.g. due to FIR processing of measurement data in computer mechanization of continuous-time models). Modification of the described LA approach based on the idea of an instrumental variable appears to be a promising area of further investigations. Moreover, the iterative LA scheme suffering from close-to-zero divisors, calls for some numerical improvements. At this stage, however, finding a prospective remedy is still an open problem.

References

1. Janiszowski, K.: To estimation in sense of the seast sum of absolute errors. In: Proceedings 5th International Symposium on Methods and Models in Automation and Robotics, vol. 2, pp. 583–588. Międzyzdroje, Poland (1998)
2. Kowalczuk, Z.: Discrete approximation of continuous-time systems—a survey. IEE Proc. -G, Circ. Devices Syst. **140**, 264–278 (1993)
3. Kowalczuk, Z., Kozłowski, J.: Continuous-time approaches to identification of continuous-time systems. Automatica **36**(8), 1229–1236 (2000)
4. Kowalczuk, Z., Kozłowski, J.: Non-quadratic quality criteria in parameter estimation of continuous-time models. IET Control Theor. Appl. (Inst. Eng. Technol.) **5**(13), 1494–1508 (2011)
5. Kozłowski, J., Kowalczuk, Z.: Robust to measurement faults, parameter estimation algorithms in problems of systems diagnostics (In Polish: Odporne na Przekłamania Pomiarowe Algorytmy Estymacji Parametrycznej w Zagadnieniach Diagnostyki Systemów). In: Kowalczuk, Z., Wiszniewski, B. (eds.) Intelligent Information Extraction for Diagnostic Purposes, pp. 221–240. PWNT, Gdańsk (2007)
6. Kozłowski, J., Kowalczuk, Z.: Continuous-time delay system identification insensitive to measurement faults. In: Kowalczuk, Z. (ed.) Diagnosis of Processes and Systems, Chapter 15, pp. 177–184. PWNT, Gdańsk (2009)
7. Kozłowski, J., Kowalczuk, Z.: On-line parameter and delay estimation of continuous-time dynamic systems. Int. J. Appl. Math. Comput. Sci. **25**(2), 223–232 (2015)
8. Sagara, S., Zhao, Z.Y.: Numerical integration approach to on-line identification of continuous-time systems. Automatica **26**(1), 63–74 (1990)

Adaptive Actuator Fault Estimation for DC Servo Motor

Mariusz Buciakowski and Marcin Witczak

Abstract The paper present the problem of robust adaptive actuator fault estimation for linear discrete-time systems. The main part of this paper presents problem of design a robust observer that will be able to estimate state vector of the system, actuator fault and decoupling the effect of an unknown input. For that purpose, the structure for robust observer was proposed. The first part of the paper deals with the design of observer. The observer is designed in such a way that a prescribed attenuation level is achieved with respect to the fault estimation error and state estimation error. The subsequent part of the paper deals with laboratory system of DC servo motor that will be used in experiment. The final part of the paper shows the experimental results for DC servo motor system, which confirm the effectiveness of the proposed approach.

Keywords Fault estimation · Actuator fault · Robustness · DC servo motor · Robust observer

1 Introduction

The problems of fault estimation [1, 2, 4, 7, 11, 12, 14] for linear systems have been intensively studied in recent years resulting in further developments for linear systems [19]. Generally, each systems can be divided into three parts: sensors, plant (or system dynamics) and actuator diagnosis [8, 17]. It is evident that each of them can be influenced by faults. The paper deals with the problem of an actuator fault estimation. In particular, active actuator fault estimation is considered. Various schemes

M. Buciakowski (✉) · M. Witczak
Institute of Control and Computation Engineering, University of Zielona Góra,
ul. Podgórna 50, 65-246 Zielona Góra, Poland
e-mail: M.Buciakowski@issi.uz.zgora.pl
url: http://www.issi.uz.zgora.pl

M. Witczak
e-mail: M.Witczak@issi.uz.zgora.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_9

that can overcome this challenging problem can be found in the literature [1, 2, 11–14, 16, 18, 20, 21]. All of them use the same deign guidelines. The fault estimation should be realized taking into account inevitable disturbances and noise as well as modeling uncertainty.

Therefore, the existing schemes can be divided with respect to the solution to the robustness issue. The first group applies the so-called unknown input decoupling (see, e.g., [8, 17] and the references therein) while the second one is based on the minimization of the unappealing disturbance effect (see, e.g. [11] and the references therein).

This paper proposes a unique approach that employs both paradigms, i.e., the unknown input is suitably decoupled while the effect of undecoupled part is suitably minimized. The proposed approach allows obtaining possibly accurate state and fault estimates, which are further used for fault-tolerant control. The proposed strategy can be perceived as a combination of approaches presented in [3, 17] along with [5] and [10].

The paper is organized as follows. Section 2 presents preliminaries regarding system and observer description. In Sect. 3, the proposed state and fault estimation strategy is described in detail. Section 4 presents an illustrative example, which shows the performance of the proposed approach. Finally, the last section concludes the paper.

2 Preliminaries

Let us consider a linear discrete-time system

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{D}\boldsymbol{d}_k + \boldsymbol{B}\boldsymbol{f}_k + \boldsymbol{W}_1\boldsymbol{w}_k \tag{1}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{W}_2\boldsymbol{w}_k \tag{2}$$

where $\boldsymbol{x}_k \in \mathbb{R}^n$ is the state, $\boldsymbol{u}_k \in \mathbb{R}^r$ stands for the input, $\boldsymbol{y}_k \in \mathbb{R}^m$ denotes the output, $\boldsymbol{f}_k \in \mathbb{R}^s$ stands for the fault, $\boldsymbol{d}_k \in \mathbb{R}^q$ is an unknown input disturbance, $\boldsymbol{w}_k \in l_2$ is a an exogenous disturbance vector and $\boldsymbol{W}_1 \in \mathbb{R}^{n\times n}$, $\boldsymbol{W}_2 \in \mathbb{R}^{m\times n}$ stand for its distribution matrices, while

$$l_2 = \left\{ \mathbf{w} \in \mathbb{R}^n | \ \|\mathbf{w}\|_{l_2} < +\infty \right\}, \|\mathbf{w}\|_{l_2} = \left(\sum_{k=0}^{\infty} \|\boldsymbol{w}_k\|^2 \right)^{\frac{1}{2}} \tag{3}$$

Using the system model (1)–(2), the problem is to design a observer that will be able to simultaneously estimate the state $\boldsymbol{x}_k$, fault $\boldsymbol{f}_k$ and decoupling the effect of an unknown input $\boldsymbol{d}_k$. For this purpose, the following structure is proposed:

$$z_{k+1} = \boldsymbol{N}\boldsymbol{z}_k + \boldsymbol{G}\boldsymbol{u}_k + \boldsymbol{L}\boldsymbol{y}_k + \boldsymbol{T}\boldsymbol{B}\hat{\boldsymbol{f}}_k \tag{4}$$

$$\hat{\boldsymbol{x}}_k = z_k - \boldsymbol{E}\boldsymbol{y}_k \tag{5}$$

$$\hat{\boldsymbol{f}}_{k+1} = \hat{\boldsymbol{f}}_k + \boldsymbol{F}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k) \tag{6}$$

Following [5, 17], let us assume that the system is observable and the following rank condition is satisfied:

$$\text{rank}(\boldsymbol{D}) = \text{rank}(\boldsymbol{CD}) = q, \quad q \leq n \tag{7}$$

Thus, it can be show that the state estimation error can be calculated as follow

$$\begin{aligned} \boldsymbol{e}_k &= \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k = \boldsymbol{x}_k - z_k + \boldsymbol{ECx}_k + \boldsymbol{EW}_2\boldsymbol{w}_k \\ &= \boldsymbol{Tx}_k - z_k + \boldsymbol{EW}_2\boldsymbol{w}_k \end{aligned} \tag{8}$$

where $\boldsymbol{T} = \boldsymbol{I} + \boldsymbol{EC}$, and hence

$$z_k = \boldsymbol{Tx}_k + \boldsymbol{EW}_2\boldsymbol{w}_k - \boldsymbol{e}_k \tag{9}$$

Following (8) the state estimation error is given by

$$\boldsymbol{e}_{k+1} = \boldsymbol{Tx}_{k+1} - z_{k+1} + \boldsymbol{EW}_2\boldsymbol{w}_{k+1} \tag{10}$$

Substituting (1) and (4) into (10) gives the state estimation error dynamics

$$\begin{aligned} \boldsymbol{e}_{k+1} &= \boldsymbol{Ne}_k + [\boldsymbol{TA} - \boldsymbol{NT} - \boldsymbol{LC}]\,\boldsymbol{x}_k \\ &\quad + \left[\boldsymbol{TW}_1 - \boldsymbol{NEW}_2 - \boldsymbol{LW}_2\right]\boldsymbol{w}_k \\ &\quad + \boldsymbol{TB}\left[f_k - \hat{f}_k\right] + [\boldsymbol{TB} - \boldsymbol{G}]\,\boldsymbol{u}_k \\ &\quad + \boldsymbol{TDd}_k + \boldsymbol{EW}_2\boldsymbol{w}_{k+1} \end{aligned} \tag{11}$$

Setting

$$\boldsymbol{TD} = \boldsymbol{0} \tag{12}$$

$$\boldsymbol{TA} = \boldsymbol{NT} + \boldsymbol{LC} \tag{13}$$

$$\boldsymbol{TB} - \boldsymbol{G} = \boldsymbol{0} \tag{14}$$

$$\boldsymbol{e}_{f,k} = \boldsymbol{f}_k - \hat{\boldsymbol{f}}_k \tag{15}$$

gives

$$\begin{aligned} \boldsymbol{e}_{k+1} &= \boldsymbol{Ne}_k + \boldsymbol{TBe}_{f,k} \\ &\quad + \left[\boldsymbol{TW}_1 - \boldsymbol{NW}_2 - \boldsymbol{LW}_2\right]\boldsymbol{w}_k + \boldsymbol{EW}_2\boldsymbol{w}_{k+1} \end{aligned} \tag{16}$$

Substituting $\boldsymbol{T} = \boldsymbol{I} + \boldsymbol{EC}$ into (13) gives

$$\begin{aligned} \boldsymbol{TA} &= \boldsymbol{NT} + \boldsymbol{LC} = \boldsymbol{N}\left[\boldsymbol{I} + \boldsymbol{EC}\right] + \boldsymbol{LC} \\ &\Rightarrow \boldsymbol{N} = \boldsymbol{TA} - \boldsymbol{NEC} - \boldsymbol{LC} = \boldsymbol{TA} - \boldsymbol{KC} \end{aligned} \tag{17}$$

where

$$\boldsymbol{K} = \boldsymbol{NE} + \boldsymbol{L} \tag{18}$$

It is evident from (18) that:

$$\begin{aligned} \boldsymbol{e}_{k+1} &= \boldsymbol{N}\boldsymbol{e}_k + \boldsymbol{TB}\boldsymbol{e}_{f,k} \\ &\quad + \left[\boldsymbol{TW}_1 - [\boldsymbol{NE} + \boldsymbol{L}]\boldsymbol{W}_2\right]\boldsymbol{w}_k + \boldsymbol{EW}_2\boldsymbol{w}_{k+1} \end{aligned} \tag{19}$$

Bearing in mind that $\boldsymbol{NE} + \boldsymbol{L} = \boldsymbol{NE} + \boldsymbol{K} - \boldsymbol{NE} = \boldsymbol{K}$ Eq. (19) can be written as follows

$$\begin{aligned} \boldsymbol{e}_{k+1} &= [\boldsymbol{TA} - \boldsymbol{KC}]\boldsymbol{e}_k + \boldsymbol{TB}\boldsymbol{e}_{f,k} \\ &\quad + \left[\boldsymbol{TW}_1 - \boldsymbol{KW}_2\right]\boldsymbol{w}_k + \boldsymbol{EW}_2\boldsymbol{w}_{k+1} \end{aligned} \tag{20}$$

Similarly, fault estimation error can be defined as

$$\boldsymbol{e}_{f,k+1} = \boldsymbol{f}_{k+1} - \hat{\boldsymbol{f}}_{k+1} \tag{21}$$

Setting

$$\boldsymbol{\varepsilon}_{f,k} = \boldsymbol{f}_{k+1} - \boldsymbol{f}_k \tag{22}$$

gives

$$\begin{aligned} \boldsymbol{e}_{f,k+1} &= \boldsymbol{f}_{k+1} - \hat{\boldsymbol{f}}_{k+1} \\ &= \boldsymbol{f}_{k+1} - \hat{\boldsymbol{f}}_k - \boldsymbol{FCe}_k - \boldsymbol{FW}_2\boldsymbol{w}_k \\ &= \boldsymbol{f}_{k+1} - \boldsymbol{f}_k + \boldsymbol{f}_k - \hat{\boldsymbol{f}}_k - \boldsymbol{FCe}_k - \boldsymbol{FW}_2\boldsymbol{w}_k \\ &= -\boldsymbol{FCe}_k + \boldsymbol{e}_{f,k} + \boldsymbol{\varepsilon}_k - \boldsymbol{FW}_2\boldsymbol{w}_k \end{aligned} \tag{23}$$

Additionally, it is assumed that

$$\boldsymbol{\varepsilon}_k \in l_2$$

Having obtained the state estimation error dynamics (20) and fault estimation error (23) are, then define

$$\bar{\boldsymbol{e}}_{k+1} = \left[\boldsymbol{e}_{k+1}^T, \boldsymbol{e}_{f,k+1}^T\right]^T, \tag{24}$$

$$\boldsymbol{v}_k = \left[\boldsymbol{w}_k^T, \boldsymbol{\varepsilon}_k^T, \boldsymbol{w}_{k+1}^T\right]^T \tag{25}$$

Thus, join state and fault estimation error can be described as

$$\bar{\boldsymbol{e}}_{k+1}^{T} = \begin{bmatrix} \boldsymbol{TA} - \boldsymbol{KC} & \boldsymbol{TB} \\ -\boldsymbol{FC} & \boldsymbol{I} \end{bmatrix} \bar{\boldsymbol{e}}_k + \begin{bmatrix} \boldsymbol{TW}_1 - \boldsymbol{KW}_2 & \boldsymbol{0} & \boldsymbol{EW}_2 \\ -\boldsymbol{FW}_2 & \boldsymbol{I} & \boldsymbol{0} \end{bmatrix} \boldsymbol{v}_k \tag{26}$$

For the purpose of further analysis equation (26) can be described in a more compact form

$$\bar{\boldsymbol{e}}_{f,k+1} = \boldsymbol{X}\bar{\boldsymbol{e}}_{f,k} + \boldsymbol{Z}\boldsymbol{v}_k \tag{27}$$

where

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{TA} & \boldsymbol{TB} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} - \begin{bmatrix} \boldsymbol{K} \\ \boldsymbol{F} \end{bmatrix} \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} = \bar{\boldsymbol{A}} - \bar{\boldsymbol{K}}\bar{\boldsymbol{C}} \tag{28}$$

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{TW}_1 & \boldsymbol{0} & \boldsymbol{EW}_1 \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \end{bmatrix} - \begin{bmatrix} \boldsymbol{K} \\ \boldsymbol{F} \end{bmatrix} \begin{bmatrix} \boldsymbol{W}_2 & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} = \bar{\boldsymbol{W}} - \bar{\boldsymbol{K}}\bar{\boldsymbol{V}} \tag{29}$$

3 Observer Design

Theorem 1 *For a prescribed disturbance attenuation level μ the observer design problem for the system (1)–(2) is solvable if there exist $\boldsymbol{N}$, $\boldsymbol{P} \succ \boldsymbol{0}$ such that the following condition is satisfied:*

$$\begin{bmatrix} \boldsymbol{I} - \boldsymbol{P} & \boldsymbol{0} & \bar{\boldsymbol{A}}^T\boldsymbol{P} - \bar{\boldsymbol{C}}^T\boldsymbol{N}^T \\ \boldsymbol{0} & -\mu^2\boldsymbol{I} & \bar{\boldsymbol{W}}^T\boldsymbol{P} - \bar{\boldsymbol{C}}^T\bar{\boldsymbol{V}}^T \\ \boldsymbol{P}\bar{\boldsymbol{A}} - \boldsymbol{N}\bar{\boldsymbol{C}} & \boldsymbol{P}\bar{\boldsymbol{W}} - \boldsymbol{N}\bar{\boldsymbol{V}} & -\boldsymbol{P} \end{bmatrix} \prec \boldsymbol{0} \tag{30}$$

with $\boldsymbol{N} = \boldsymbol{P}\bar{\boldsymbol{K}}$.

Proof The problem of $\mathcal{H}_\infty$ [9] observer design is to determine the matrix $\boldsymbol{K}$, such that

$$\lim_{k\to\infty} \bar{\boldsymbol{e}}_k = \boldsymbol{0} \quad \text{for } \boldsymbol{w}_k = \boldsymbol{0} \tag{31}$$

$$\|\bar{\boldsymbol{e}}_k\|_{l_2} \le \mu\|\boldsymbol{v}_k\|_{l_2} \quad \text{for } \boldsymbol{w}_k \neq \boldsymbol{0},\ \bar{\boldsymbol{e}}_0 = \boldsymbol{0} \tag{32}$$

In order to settle the above problem it suffices to find a Lyapunov function V_k such that:

$$\Delta V_k + \bar{\boldsymbol{e}}_{f,k}^T\bar{\boldsymbol{e}}_{f,k} - \mu^2\boldsymbol{v}_k^T\boldsymbol{v}_k < 0,\ k = 0, \ldots \infty \tag{33}$$

where

$$\Delta V_k = V_{k+1} - V_k \tag{34}$$

$$V_k = \bar{\boldsymbol{e}}_k^T \boldsymbol{P} \bar{\boldsymbol{e}}_k \tag{35}$$

$$\Delta V_k = \bar{\boldsymbol{e}}_{k+1}^T \boldsymbol{P} \bar{\boldsymbol{e}}_{k+1} - \bar{\boldsymbol{e}}_k^T \boldsymbol{P} \bar{\boldsymbol{e}}_k \tag{36}$$

Consequently, using (27)

$$\begin{aligned} &\Delta V_k + \bar{\boldsymbol{e}}_{f,k}^T \bar{\boldsymbol{e}}_{f,k} - \mu^2 \boldsymbol{v}_k^T \boldsymbol{v}_k = \\ &\bar{\boldsymbol{e}}_k^T \left(\boldsymbol{X}^T \boldsymbol{P} \boldsymbol{X} + \boldsymbol{I} - \boldsymbol{P} \right) \bar{\boldsymbol{e}}_k + \\ &\bar{\boldsymbol{e}}_k^T \left(\boldsymbol{X}^T \boldsymbol{P} \boldsymbol{Z} \right) \boldsymbol{v}_k + \\ &\boldsymbol{v}_k^T \left(\boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{X} \right) \bar{\boldsymbol{e}}_k + \\ &\boldsymbol{v}_k^T \left(\boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Z} - \mu^2 \boldsymbol{I} \right) \boldsymbol{v}_k < \boldsymbol{0} \end{aligned} \tag{37}$$

By defining

$$\bar{\boldsymbol{v}}_k = [\bar{\boldsymbol{e}}_k^T, \boldsymbol{v}_k^T]^T \tag{38}$$

it can be shown that (37) is equivalent to

$$\bar{\boldsymbol{v}}_k^T \begin{bmatrix} \boldsymbol{X}^T \boldsymbol{P} \boldsymbol{X} + \boldsymbol{I} - \boldsymbol{P} & \boldsymbol{X}^T \boldsymbol{P} \boldsymbol{Z} \\ \boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{X} & \boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Z} - \mu^2 \boldsymbol{I} \end{bmatrix} \bar{\boldsymbol{v}}_k < \boldsymbol{0} \tag{39}$$

The matrix (39) must be negative definite and writing it as

$$\begin{bmatrix} \boldsymbol{X}^T \\ \boldsymbol{Z}^T \end{bmatrix} \boldsymbol{P} \begin{bmatrix} \boldsymbol{X} & \boldsymbol{Z} \end{bmatrix} + \begin{bmatrix} \boldsymbol{I} - \boldsymbol{P} & \boldsymbol{0} \\ \boldsymbol{0} & -\mu^2 \boldsymbol{I} \end{bmatrix} < \boldsymbol{0} \tag{40}$$

and by applying Schur complements, one has

$$\begin{bmatrix} \boldsymbol{I} - \boldsymbol{P} & \boldsymbol{0} & \boldsymbol{X}^T \boldsymbol{P} \\ \boldsymbol{0} & -\mu^2 \boldsymbol{I} & \boldsymbol{Z}^T \boldsymbol{P} \\ \boldsymbol{P} \boldsymbol{X} & \boldsymbol{P} \boldsymbol{Z} & -\boldsymbol{P} \end{bmatrix} < \boldsymbol{0} \tag{41}$$

Define the following identities

$$\boldsymbol{P} \boldsymbol{X} = \boldsymbol{P} \bar{\boldsymbol{A}} - \boldsymbol{P} \bar{\boldsymbol{K}} \bar{\boldsymbol{C}} = \boldsymbol{P} \bar{\boldsymbol{A}} - \boldsymbol{N} \bar{\boldsymbol{C}} \tag{42}$$

$$\boldsymbol{P} \boldsymbol{Z} = \boldsymbol{P} \bar{\boldsymbol{W}} - \boldsymbol{P} \bar{\boldsymbol{K}} \bar{\boldsymbol{C}} = \boldsymbol{P} \bar{\boldsymbol{W}} - \boldsymbol{N} \bar{\boldsymbol{V}} \tag{43}$$

where $\boldsymbol{N} = \boldsymbol{P} \bar{\boldsymbol{K}}$. Using (42)–(43) and substituting $\boldsymbol{N} = \boldsymbol{P} \bar{\boldsymbol{K}}$ gives (30) which completes the proof. □

Finally, the design procedure boils down to solving (30) with respect to, $\boldsymbol{N}$, $\boldsymbol{P}$ and then calculating $\boldsymbol{K} = \boldsymbol{P}^{-1}\boldsymbol{N}$. It should be emphasized that the observer design problem can be treated as a minimization task, i.e.:

$$\mu^* = \min_{\mu>0, \boldsymbol{P}>\boldsymbol{0}, \boldsymbol{N}} \mu \tag{44}$$

4 Illustrative Example

In order to verify the proposed approach, let us consider a DC servo motor portrayed in Fig. 1. The DC servo motor is designed for simulating the real industrial servo system in the laboratory conditions. The DC servo motor consists of a DC motor, tachogenerator, encoder, magnetic brake, gearbox and a system to simulate the load inertia. The considered DC servo motor has been designed to operate with an external, PC-based digital controller. The control computer communicates with the encoder, tachogenerator and DC motor via a dedicated I/O board and the power interface. The I/O board is controlled by the real-time software, which operates in the Matlab/Simulink environment. The system can be used to practically verify both position and speed control, as well as identification and diagnostics methods. DC motor can be described by the following continuous state space equation [6]:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{T} \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 \\ \frac{K}{T} \end{bmatrix}, \quad \boldsymbol{C} = \boldsymbol{I} \tag{45}$$

where T is motor time constant and K is motor gain. The first state x_1 is the rotor position and second state x_2 describes the rotor speed. The parameters were identified based on the procedure described in [15]. Figure 2 shows comparison between the actual position of the motor's rotor and the response of the model. Figure 3 shows comparison between the actual speed of the motor's rotor and the response of the model. The model (45) was discretized using Euler's method and sampling time $T_s = 0.01s$. Finally, the system matrices are defined as follows

Fig. 1 DC Servo system

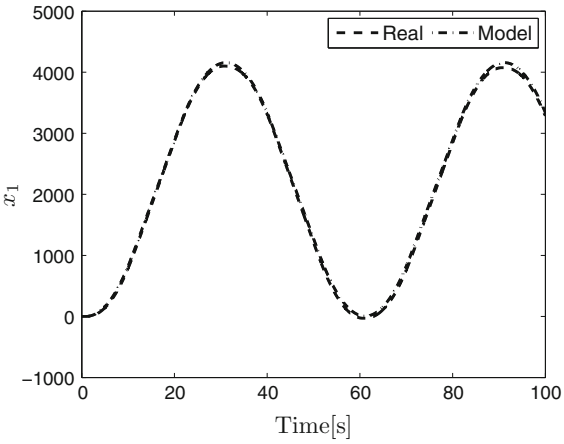


Fig. 2 Comparison between the actual position of the motor's rotor and the response of the model

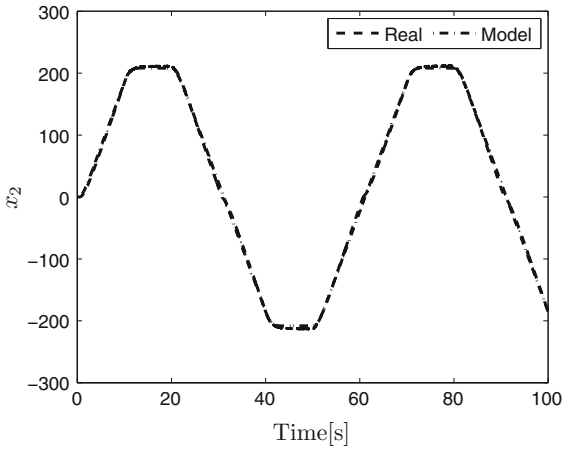


Fig. 3 Comparison between the actual speed of the motor's rotor and the response of the model

$$\boldsymbol{A} = \begin{bmatrix} 1.0000 & 0.0100 \\ 0 & 0.9905 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 \\ 2.0067 \end{bmatrix}, \quad \boldsymbol{C} = \boldsymbol{I} \tag{46}$$

As a result of solving the problem (30), the following matrices and parameters were obtained:

$$\mu = 0.976$$

$$\boldsymbol{D} = \begin{bmatrix} 0.0011 \\ 0.1080 \end{bmatrix}, \quad \boldsymbol{N} = \begin{bmatrix} -1.6157 & 0.0206 \\ 0.0163 & -0.0002 \end{bmatrix}, \quad \boldsymbol{G} = \begin{bmatrix} -0.0203 \\ 0.0002 \end{bmatrix}$$

$$\boldsymbol{L} = \begin{bmatrix} 2.6156 & -0.0163 \\ -0.0264 & 0.0002 \end{bmatrix}, \quad \boldsymbol{T} = \begin{bmatrix} 0.9999 & -0.0101 \\ -0.0101 & 0.0001 \end{bmatrix}, \quad \boldsymbol{E} = \begin{bmatrix} -0.0001 & -0.0101 \\ -0.0101 & -0.9999 \end{bmatrix}$$

$$\boldsymbol{F} = \begin{bmatrix} -79.7856 & 0.8050 \end{bmatrix}$$

The actuator faults scenarios, i.e., performance decrease of the rotor, are described as follows:

S1: An abrupt fault: 10 % performance decrease of the first actuator:

$$\boldsymbol{f}_{1,k} = \begin{cases} -0.1\boldsymbol{u}_{f,1,k} & 40 \leq k \leq 70 \\ 0 & \text{otherwise} \end{cases}$$

S2: An incipient fault: 50 % performance decrease of the first actuator:

$$\boldsymbol{f}_{1,k} = \begin{cases} 0.5(k-70), & \text{for } 40 \leq k \leq 70 \\ 0 & \text{otherwise} \end{cases}$$

The analysis starts with the fault-free. Figures 4 and 5 present the state estimation. From these results, it is evident that the state estimation is performed with a good

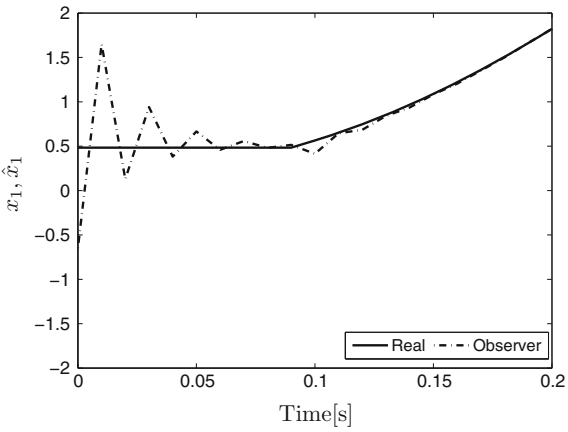


Fig. 4 Position x_1 (*solid line*), and its estimates $\hat{x}_1$ (*dashed line*) (for $t = 0, \ldots, 0.25$)

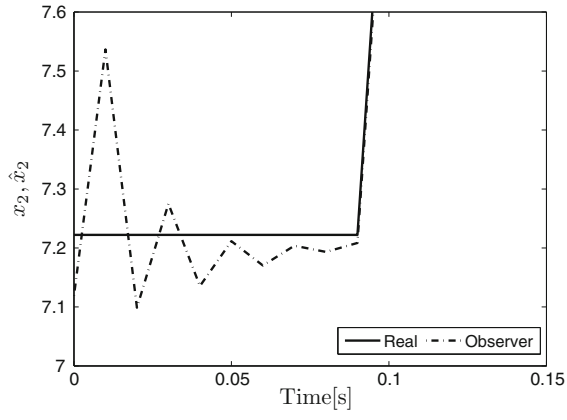


Fig. 5 Speed x_2 (*solid line*), and its estimates $\hat{x}_2$ (*dashed line*) (for $t = 0, \ldots, 0.15$)

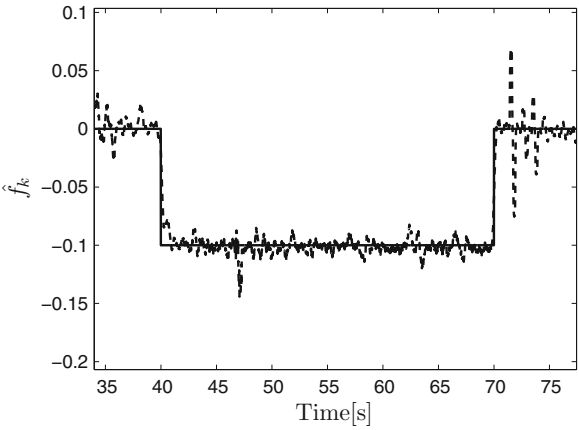


Fig. 6 S1: Fault $\boldsymbol{f}_k$ (*solid line*), and its estimates (*dashed line*)

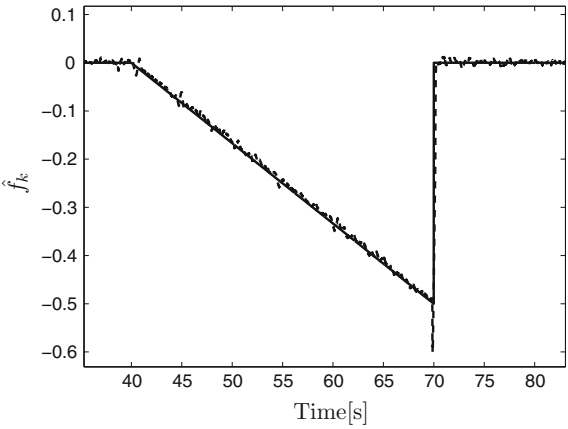


Fig. 7 S2: Fault $\boldsymbol{f}_k$ (*solid line*), and its estimates (*dashed line*)

quality. Now, let us consider the faulty case, i.e., $\boldsymbol{f}_k \neq \boldsymbol{0}$. Figure 6 shows the fault and its estimate for the faulty case, for the first and second fault scenarios. The obtained fault estimation results also confirm the high performance of the proposed methodology and recommend it for the FTC applications.

5 Conclusions

The fault estimation for DC servo motor was proposed. The $\mathcal{H}_\infty$ framework was used to design robust observer to simultaneously estimate the state $\boldsymbol{x}_k$, fault $\boldsymbol{f}_k$, and decoupling the effect of an unknown input $\boldsymbol{d}_k$. The proposed design procedure for observer design is relatively simple and boils down to solving a set of linear matrix inequalities. Finally, the paper shows an illustrative example demonstrating the effectiveness of the proposed approach in the fault estimation. The future work will focus on the fault estimation for non-linear systems.

Acknowledgments The authors would like to express their sincere gratitude to the referees, whose constructive comments contributed significantly toward the current shape of the paper. The work was supported by the National Science Center of Poland under grant no. 2013/11/B/ST7/01110.

References

1. Buciakowski, M., de Rozprza-Faygel, M., Ochalek, J., Witczak, M.: Actuator fault diagnosis and fault-tolerant control: application to the quadruple-tank process. In: European Workshop on Advanced Control and Diagnosis, vol. 570, p. 10. IOPScience, Berlin (2014)
2. Buciakowski, M., Witczak, M., Luzar, M.: Robust Fault-tolerant Control for a Multi-tank System. In: 11th International Conference on Diagnostics of Processes and Systems, p. 12. Lagow Lubuski, Poland (2013)
3. Chen, J., Patton, R.: Robust Model-Based Fault Diagnosis for Dynamic Systems. Kluwer Academic Publishers, Boston (1999)
4. Gertler, J.: Fault Detection and Diagnosis in Engineering Systems. Marcel Dekker, New York (1998)
5. Gillijns, S., de Moor, B.: Unbiased minimum-variance input and state estimation for linear discrete-time systems. Automatica **43**, 111–116 (2007)
6. INTECO: Modular Servo System—user's Manual. INTECO, www.inteco.com.pl (2013)
7. Isermann, R.: Fault Diagnosis Applications: Model Based Condition Monitoring, Actuators, Drives, Machinery, Plants, Sensors, and Fault-tolerant Systems. Springer, Berlin (2011)
8. Korbicz, J., Kościelny, J., Kowalczuk, Z., Cholewa, W. (eds.): Fault Diagnosis. Models, Artificial Intelligence, Applications. Springer, Berlin (2004)
9. Li, H., Fu, M.: A linear matrix inequality approach to robust h_∞ filtering. IEEE Trans. Signal Process. **45**(9), 2338–2350 (1997)
10. Nobrega, E., Abdalla, M., Grigoriadis, K.: Robust fault estimation of unceratain systems using an LMI-based approach. Int. J. Robust Nonlinear Control **18**(7), 1657–1680 (2008)
11. Noura, H., Theilliol, D., Ponsart, J., Chamseddine, A.: Fault-Tolerant Control Systems: Design and Practical Applications. Springer, Berlin (2003)
12. de Oca, S., Puig, V., Witczak, M., Dziekan, L.: Fault-tolerant Control strategy for actuator faults using LPV techniques: application to a two degree of freedom helicopter. Int. J. Appl. Math. Comput. Sci. **22**(1), 161–171 (2012)
13. Ossmann, D., Varga, A.: Detection and identification of loss of efficiency faults of flight actuators. Int. J. Appl. Math. Comput. Sci. **25**(1), 53–63 (2015)
14. Puig, V.: Fault diagnosis and fault tolerant control using set-membership approaches: application to real case studies. Int. J. Appl. Math. Comput. Sci. **20**(4), 619–635 (2010)
15. Rotondo, D., Nejjari, F.F., Puig, V.: Quasi-LPV modeling, identification and control of a twin rotor mimo system. Control Eng. Pract. **21**(6), 829–846 (2013)
16. Rotondo, D., Nejjari, F., Puig, V.: Robust quasi-LPV model reference FTC of a quadrotor UAV subject to actuator faults. Int. J. Appl. Math. Comput. Sci. **25**(1), 7–22 (2015)
17. Witczak, M.: Modelling and Estimation Strategies for Fault Diagnosis of Non-linear Systems. Springer, Berlin (2007)
18. Witczak, M., Buciakowski, M., Aubrun, C.: Predictive actuator fault-tolerant control under ellipsoidal bounding. Int. J. Adapt. Control Signal Process. (2015)
19. Witczak, M., Buciakowski, M., Mrugalski, M.: An H-infinity approach to fault estimation of non-linear systems: application to one-link manipulator. In: Methods and Models in Automation and Robotics, pp. 456–461. Miedzyzdroje, Poland (2014)
20. Witczak, M., Buciakowski, M., Puig, V., Rotondo, D., Nejjari, F.: An LMI approach to robust fault estimation for a class of nonlinear systems. Int. J. Robust Nonlinear Control (2015)
21. Witczak, M., Mrugalski, M., Korbicz, J.: Towards robust neural-network-based sensor and actuator fault diagnosis: application to a tunnel furnace. Neural Processing Letters (2014)

Evaluating the Position of a Mobile Robot Using Accelerometer Data

Zdzisław Kowalczuk and Tomasz Merta

Abstract This paper analyzes the problem of determining the position of a robot using an accelerometer, which is an essential part of inertial measurement units (IMU). The information gained from such a gauge, however, requires double integration of sensor data. To assure an expected effect, a mathematical model of a low-cost accelerometer of the MEMS type is derived. Moreover, in order to improve the performance of positioning based on acceleration, we propose to construct the designed location system using a mathematical model of the considered mobile robot controlled by a DC motor. Computational and simulation case studies of the resulting observer-based system, in deterministic and stochastic settings, are performed to test the method, to determine its limitations, and, in particular, to verify if the system can work properly for low-cost accelerometers of standard precision.

Keywords IMU · Accelerometer · Mobile robots · Positioning

1 Introduction

In the recent years the price of inertial sensors has dropped significantly. Among many electronic devices, the MEMS sensors are used frequently due to their low-cost production and small sizes. These sensors are widely spread also because of the rapid development of mobile telephony and cell phones. Accelerometers are most often used not only in smartphones, but in smart watches, tablets, and GPS receivers, as well. The main aim of using accelerometers in these devices is to determine orientation. Other applications that rely on detecting vibration or free fall, are used in photo cameras, game consoles, and hard disks [18].

Z. Kowalczuk (✉) · T. Merta
Faculty of Electronics, Telecommunications and Informatics,
Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland
e-mail: kova@pg.gda.pl

T. Merta
e-mail: tomasz.merta@pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_10

More developed tasks usually require more expensive dedicated sensors. In robotic applications accelerometers are effective in detecting large forces or free fall (to avoid collision) and in measuring vibrations[1] (to suppress them). Complex tasks require specialized systems equipped with diversified sensors [5] (for instance, accelerometer can support sonar or lidar). One can also give examples relating to the aid during a surgical incision [10].

In missions connected with image acquisition, the MEMS sensors are typically used for image stabilization. A single accelerometer may suffice to compensate vibrations of a photo camera. Improved video requires taking into account (eliminating) the changes of both the position and orientation of the video camera. When the camera is not static, and acquisition conditions are poor, its image shift can be reduced through processing supported by inertial measurements [6].

The issues related to robot localization (position determination) are widely discussed in SLAM (Simultaneous Localization and Mapping). SLAM solutions often use fusion of data gathered from various sensors, including IMU (Inertial Measurement Unit). It is convenient to use an external or global system in evaluating the position of a robot. The precision of a global system (like GPS, with a global view from a static camera) or a pre-arranged map with landmarks is generally fixed. A relative position can be necessary if a global system is unavailable, or if it is not applicable for a given task. To improve robot navigation the IMU sensory data can be backed by vision feature detection algorithms [13, 17]. Despite many enhancements of the IMU-based measurements it is not an easy task to find the precise position of a robot. On the contrary, the orientation of the robot can be accurately obtained solely with the use of a typical IMU.

The only possibility to determine the position of the robot based on data from IMU is double integration of the accelerometer data [8]. However, even small inaccuracies in acceleration data may lead to significant errors in the estimated position. And this effect increases with the time of integration. Such an accelerometer-based positioning technique is also extremely ineffective due to the noise and drift characteristics of the MEMS sensors.

In general, we are interested in mapping of a limited environment using mobile robots with vision facilities. In this research a key point is to know the accurate orientation and position of the applied moving camera. Such a mapping process is founded on the idea of taking different stereo images in different robot relocations [9]. In the proposed system when the robot relocation is complete the actual orientation is determined by fusing data from inertial sensors. Magnetometer, accelerometer and gyroscope placed on a single board (IMU) provide data to calculate actual rotation (angular position) of the robot. At this stage of research there are two algorithms of sensor fusion tested on our robotic platform. The problem of determining position was first founded on integration of accelerometer data. However, the encountered difficulties with double integration directed us to analyze the MEMS accelerometer and signal characteristics after integration. An additional motivation for the inclusion of an accelerometer model in our further analysis were the conclusions gained from

[1] The maximum detected vibration reaches 2000 g.

different researchers who announced a success in a double integration or denied this [3, 7, 8, 11, 12, 15].

In our recent study we investigated crucial factors for obtaining proper double integration results including sensor features, noise drift, influence of errors of angular position, etc. The results have shown that only very low frequencies in acceleration signals are vital for correct double integration effects. Generally, filtration of accelerometer signals using low-pass filters is not effective. The difference between the filtered and non-filtered signals has been merely noticeable for very long FIR filters (with the cutoff frequency $f_{stop} < 0.5$ Hz for the sampling frequency $f_{sa} = 400$ Hz). Therefore, in this work we use the more appropriate state-space approach based on modeling the dynamics of an electrically controlled mobile robot. In this we also take into consideration the characteristics of MEMS accelerometers consisting in low frequency noise/drifts.

2 Accelerometer Features

To minimize errors associated with position estimation based on accelerometer data, it is important to analyze all the pertinent factors and describe a proper model of accelerometer. Those factors include noise and nonlinearities typical for digital MEMS sensors, gravity, influence of orientation accuracy, and placement of the sensor on the robot. Noise and nonlinearities are discussed on the basis of a low-cost 3-axis digital accelerometer (Pololu IMU with LSM303D), and the experimental results presented in the following are related to this accelerometer.

2.1 Nonlinearities and Noise

Nonlinearities of MEMS sensors are related to many factors. The least problematic of them, although very important, are the scale factor and bias [14]. The scale factor is the ratio of the change at the sensor output to the change at its input that is intended to be measured. Bias is known as a nonzero offset of the sensor output signal from the expected true value. Both factors are associated with inaccuracies of material and construction. Other factors are related to changes of operation conditions [2]. As a result, one observes alterations of accelerometer parameters caused by variation of temperature, pressure; such as sensitivity change versus temperature, zero-g level conversion versus temperature (resulting in a bias drift). In stable operating conditions the parasitic effects of the scaling factor and bias can be minimized by appropriate calibration.

Noise in MEMS accelerometers is related to electrical and mechanical nonlinearities [1]. A detailed analysis of all important noise characteristics pertinent in modeling an accelerometer is a difficult task. We will focus here on two important types of noise: 'material' and 'circuit' noise (or thermo-mechanical and 'flicker' noise [20]).

The first results in fluctuations at a rate much greater than the sampling rate applied in the sensor. Thus, measurements gained from the sensor are perturbed by approximately *white noise*. The second, the flicker noise, generated by integrated electronic circuits (e.g. operational amplifiers), causes other parasitic signals, contributing to *white Gaussian noise* at high frequencies and *pink noise* (of $1/f$-type) characteristics at low frequencies [4]. Further consequences of the flicker noise are bias fluctuations. Both the discussed noise contributors may be integrated into one signal, which can be modeled as a *random walk*. The random movement of bias creates a second-order random walk in velocity, and a third-order random walk in position. As a consequence, double integration of acceleration signals with fluctuating biases causes an error in position, which grows quadratically with time [15].

2.2 DC and Rotation

In static cases accelerometer data contain the DC gravity component g equal $9.81\, m/s^2$, which generally does not change over a given local area [22]. In dynamic conditions when the accelerometer is being rotated, gravity changes according to sensor rotations. Rotations around the x, y and z axes are referred to as *roll*(ϕ), *pitch*(θ) and *yaw*(ψ), respectively. It is easy to calculate orientation for the static case–we just need to determine the actual direction (rotation) of the gravity vector. Clearly, there are situations when a certain rotation does not influence the gravity vector, and other issues (like the gimbal lock problem which may occur while working with rotations [20]). To fully detect the robot orientation, an additional sensor is necessary.

A completely different approach must be applied to obtain linear acceleration. The presence of gravity has a perturbing (warping) affect on the measurements what should be minimized. In stationary situations, where the object is affected only by gravity (and orientation does not change) it is possible to calculate a DC offset. After the DC removal, any nonzero acceleration is clearly a result of sensor noise. However, when the object moves its orientation may change so that the gravity substantially interferes with the measurements of linear acceleration. Even small inaccuracies of orientation are significant for integration results [9]. Assuming, for instance, the usage of a 1-axis accelerometer, where the axis is orthogonal to the gravity vector, a rotation around this axis by 1 deg results in an additional $0.17\ \mathrm{m/s^2}$ acceleration contribution.

In reality a mobile robot can encounter rapid rotations for a short period of time. This situation is typical when the robot passes through a small obstacle or travels on a rough ground. If the sensor is mounted not precisely at the cross point, where all the 3 rotation axes of the object intersect, an additional acceleration appears. The reason for this effect lies in ground topography, robot trajectory (circular motion), or IMU mounting displacement. In real applications the displacement of IMU sensors is small (the Pololu IMU-9 v.1 chips are shifted about 1 cm [16]). Real movement on rough ground can contribute to a relocation of one wheel up or down and to a shift of the center of robot rotation. It is difficult to determine the size of this shift. At the

same time, the impact of dynamics of rotations on the accelerometer measurements is significant [9].

When displacement (r) is nonzero the centrifugal and tangential accelerations should be taken into consideration. The centrifugal acceleration a_{cf} is normal to the rotation direction and nonzero when a respective angular velocity, $\dot{\phi}, \dot{\theta}, \dot{\psi}$, is nonzero. The tangential acceleration a_{tg} is parallel to the rotation direction and takes a nonzero value when the angular acceleration ($\ddot{\phi}, \ddot{\theta}, \ddot{\psi}$) is nonzero.

2.3 Accelerometer Model

After presenting all important features of accelerometer we can describe an appropriate model. To avoid unambiguous symbols in the model it is essential to define the expressions in a proper coordinate system. Let us assume that the global coordinates are expressed as a_x, a_y, a_z and the objects coordinates as a_l, a_s, a_t (longitudinal, side, transversal components, respectively). Symbols $\tilde{a}_l, \tilde{a}_s, \tilde{a}_t$ stand for acceleration measurements to distinguish measurements and accelerations which are the effects of forces. The transformation between global and object Cartesian coordinates can be expressed as

$$\begin{bmatrix} a_l \\ a_s \\ a_t \end{bmatrix} = T_g^o \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \tag{1}$$

This transformation can be expressed with a translation vector and rotation matrices. The translation vector is the sought position of the robot, thus the transformation reduces to rotations. Let us assume that we deal with the right-handed system. Rotation around the axis x, y, and z ($roll(\phi)$, $pitch(\theta)$, $yaw(\psi)$) can be expressed as a single combined rotation matrix (2), where the trigonometric functions are written using abbreviated notations ($s\alpha = \sin\alpha, c\alpha = \cos\alpha$):

$$\begin{aligned} T_g^o &= R_{xyz} = R_x(\phi)R_y(\theta)R_z(\psi) = \\ &= \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ -c\phi s\psi + s\phi s\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & s\phi c\theta \\ s\phi s\psi + c\phi s\theta c\psi & -s\phi c\psi + c\phi s\theta s\psi & c\phi c\theta \end{bmatrix} \end{aligned} \tag{2}$$

For the purpose of describing the model in object coordinates, we need to transform all important factors of external origins. Acceleration measured by a MEMS accelerometer includes the true linear acceleration of the object, gravity, centrifugal and tangential acceleration, and accelerometer noise. Only gravity has external origin. It is constant in global coordinates thus we need to transform gravity to the object coordinates [9]. The accelerometer can thus be modeled as

$$\begin{bmatrix} \tilde{a}_l \\ \tilde{a}_s \\ \tilde{a}_t \end{bmatrix} = \begin{bmatrix} a_l \\ a_s \\ a_t \end{bmatrix} + g \begin{bmatrix} -s\theta \\ s\phi c\theta \\ c\phi c\theta \end{bmatrix} + \begin{bmatrix} (\dot{\theta}^2 - \ddot{\psi})r_l \\ (\dot{\phi}^2 + \dot{\psi}^2)r_s \\ (\ddot{\phi} - \ddot{\theta})r_t \end{bmatrix} + \begin{bmatrix} \eta_l \\ \eta_s \\ \eta_t \end{bmatrix} \tag{3}$$

To find the position from accelerometer data one needs to determine the true linear acceleration and to integrate the results twice. In such computations gravity, the dynamics of rotation, and the noise factor ought to be discounted from accelerometer measurements. Due to (3) the gravity removal requires accurate data about the object orientation (angles θ, ϕ). At this stage of research we gain the angular position by fusing the data of IMU. The angular position has been provided with the precision about 0.5–1°. The error associated with gravity and the dynamics of rotation can be treated as noise of angular position.

The last factor related to the measurement noise of accelerometers is described by the manufacturer. In the case of low-cost accelerometers there is only a general information about linear acceleration noise density which is a square root of the power spectral density measured in $\mu g/\sqrt{\text{Hz}}$. This parameter represents information about noise variance (the thermo-mechanical white noise). It is also related to the signal bandwidth and anti-aliasing filter applied. Linear acceleration typical zero-g level offset accuracy describe signal fluctuations when there is no acceleration. This parameter is critical since it has a great impact in static operational conditions.

Simulations using this model show that very high precision is necessary to calculate the position by integrating the accelerometer data. The performed analysis of integration [9] proves that large errors in position are caused by very low frequency drift in noise. Thus additional information about lower-frequency dynamics is compulsory to improve the results of positioning. Moreover, the assumed purely kinematic approach does not take into account any restrictions or inertia in motion of the robot. Therefore, we propose using a dynamic approach along with a mathematical model of the moving robot.

3 Robot Model and its State Observers

In our practical research a specific chassis of a mobile robot has been considered for testing both the IMU and the vision system. It is a typical construction with 2 wheels driven by DC motors and a free moving rear caster. Such a solution can be easily found in many commercial tenders. The main difference lies in the specific engine parameters, wheel sizes and the robot's weight. There are number of simplifications used to derive a mathematical description as an analog model of such a structure that would be useful for synthesis and simulation. We assume that the robot is equipped with a single DC motor with one shaft which drives two wheels (the robot's turns are not modeled). The mass is distributed evenly between the 2 wheels, and the influence of the free moving rear caster on the robot dynamics is neglected. The resulting robot

model is described in state space with voltage V as an input, and the motor current i and the rotor angular velocity ω [19] integrated in the state vector:

$$\begin{aligned} \begin{bmatrix} \dot{i} \\ \dot{\omega} \end{bmatrix} &= \begin{bmatrix} -\frac{R_r}{L_r} & -\frac{-k_e}{L_r} \\ \frac{k_m}{J} & -\frac{B}{J} \end{bmatrix} \cdot \begin{bmatrix} i \\ \omega \end{bmatrix} + \begin{bmatrix} \frac{1}{L_r} \\ 0 \end{bmatrix} \cdot V \\ y &= \begin{bmatrix} 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} i \\ \omega \end{bmatrix} + [0] \end{aligned} \tag{4}$$

where
k_m, k_e–mechanical, electrical constant
R_r, L_r–rotor resistance, inductance
J–moment rotor inertia
B–viscous friction coefficient (of the rotor shaft).

Apparently, such a system has no relation to acceleration. Therefore, assuming observability of the system, we convert it to a convenient state-space controllable canonical form, where the consecutive derivatives of an appropriate instrumental variable have the necessary interpretation of angular velocity and angular acceleration (as the components of the state vector). This system can be expressed as

$$\begin{aligned} \begin{bmatrix} \dot{\omega} \\ \dot{\epsilon} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\frac{k_m k_e}{J L_r} & -\frac{R_r}{L_r} \end{bmatrix} \cdot \begin{bmatrix} \omega \\ \epsilon \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k_m}{J L_r} \end{bmatrix} \cdot V \\ y &= \begin{bmatrix} 0 & \frac{k_m}{J L_r} \end{bmatrix} \cdot \begin{bmatrix} \omega \\ \epsilon \end{bmatrix} + [0] \end{aligned} \tag{5}$$

The above description can be used as the model of the applied mobile robot, if we know the values of all the pertinent model parameters, including gear ratio, wheel size, and inertia of the whole robot. Data on the size of wheels and the gear ratios are provided by the manufacturer. The robot's inertia requires additional measurement (e.g. [21]). The necessary mechanical and electrical constants can be calculated using the stall torque and stall current of the DC motor. To analyze on-line the state vector in the presence of small uncertainties we can use a simple (Luenberger) state observer. The poles of observation (designed via a gain matrix H) should be set rightly to ensure a proper asymptotic behavior of our observer. Clearly, poles positioning can also improve (to a certain extent) observation robustness to the presence of measurement noise. For computational analysis and simulation experiments the proposed continuous-time model (suitable for the Laplace transform) has been converted to its discrete-time equivalent (appropriate for the Z-transformation), and a digital Leuenberger observer has been designed. The sampling frequency f_s of the digital system has been matched to the frequency capabilities of the MEMS

accelerometer. Moreover, in the search of further improvements in our robot position estimation, we have also implemented a simple Kalman filter suitable for more stochastic settings of modeling and measurement.

4 Experiments

The performance of the presented models of the mobile robot and their observers has been assessed using MATLAB computations/simulations. The analog model with the analog state-space observer, as well as the discrete-time robot model with the digital Luenberger observer and the digital Kalman filter mentioned above have been considered. The robot model parameters were evaluated using the datasheet of the DC motor (manufactured by DAGU). Certain parameters like moment of robot inertia were calculated on the basis of total weight (note that precise determination of robot inertia requires a specific laboratory equipment). The resulting model parameters are given in Table 1.

Other important parameters connected with the robot model are: wheel radius $r_w = 0.03$ m, and engine gear ratio $ratio = 1{:}19$. The DC motor is usually stimulated with a PWM signal. However for simulation readability the model was excited with the input signal presented in Fig. 1a, and the corresponding (ideal) position for this input is shown in Fig. 1b.

The designed signal slope is not rapid, therefore any undervaluation of the moment of inertia and the mechanical (motor) time constants has not affected the robot motion dynamics considerably. Based on the motor datasheet and the weight of the robot, we have assumed that the robot starts to move at a minimum voltage of 0.5 V. Thus the input signal is shown in Fig. 1 already without the DC component of 0.5 V.

The acceleration signal was generated based on the output of the computational model of the robot. However, this signal was further modified by angular-position noise and measurement noise selected with the use of the accelerometer model described above. The measurement noise level was determined according to the accelerometer datasheet, where it is suggested that the noise offset will not exceed ± 0.06 g (average g is equal 9.81 m/s^2). To set the level of the angular-position noise we applied fusion algorithms that calculate orientation. Based on our own observations and [12], the angular position noise was assumed to be not grater than 0.7°. Moreover, the digital system sampling frequency was set to $f_s = 800$ Hz, to provide a common sampling rate for the accelerometer and to ensure that we can effectively

Table 1 Robot model parameters used in the system evaluation

Model parameter	k_m [$\frac{Nm}{A}$]	k_e [$\frac{Vs}{rad}$]	J[kg · m^2]	B[$\frac{\text{kg}}{\text{ms}}$]	R_r [Ohm]	L_r [H]
Value	0.7	0.04	$3 \cdot 10^{-5}$	$1 \cdot 10^{-4}$	2	$1.5 \cdot 10^{-3}$

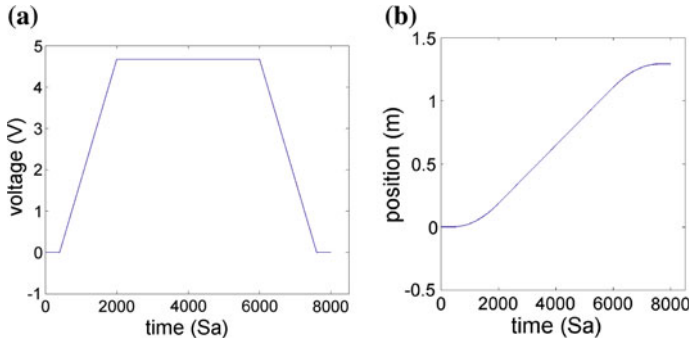


Fig. 1 Experimantal setting: (**b**) ideal position of the robot corresponding to (**a**) the applied control input

observe the model of a robot (the motor time constant of the robot with a load is greater than 3 ms).

4.1 Luenberger Observer

In simulations the Luenberger observer has been responsible for estimation of the state variables for the applied input signal and accelerometer data. The ideal states of the robot (velocity and acceleration) are presented in Fig. 2a.

A typical accelerometer noise process (η_{acc}) has been calculated as

$$\begin{aligned} \eta_d(n) &= \eta_d(n-1) + N(0, \sigma_d) \\ \eta_{acc}(n) &= \eta_d(n) + N(0, \sigma_n) \end{aligned} \tag{6}$$

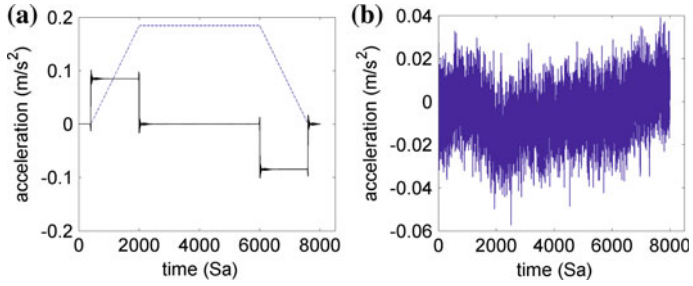


Fig. 2 Ideal vector state and noise typical for a low-cost MEMS accelerometer: **a** Ideal velocity and acceleration (*dashed blue* and *solid black*); **b** Accelerometer noise

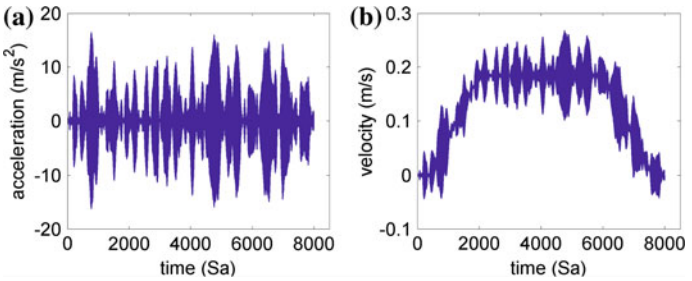


Fig. 3 State estimation by the Luenberger observer for typical accelerometer noise: **a** Estimated acceleration; **b** Estimated velocity

where η_d is a random walk (drift) generated by integrating white noise (σ_d). The resultant disturbance (6), shown in Fig. 2b, contributes to the modeled robot accelerometric signal. The influence of the noise on the state vector is visualized in Fig. 3. The estimated acceleration and velocity are significantly distorted. However, the position calculated by integrating the velocity has no error (differences are less than 10^{-3}m). With the assumed restrictions the Luenberger observer has thus turned out to be sufficient for position estimation.

4.2 Kalman Filter

In order to improve the results of estimation we have also applied the Kalman Filter (KF) matched to the described discrete-time mobile robot model with the following parameters: the measurement noise variance $\sigma_m^2 = 1$ and the processing noise variance $\sigma_p^2 = 0.1$. The influence of typical noise of the MEMS accelerometer on the state vector estimation is presented in Fig. 4.

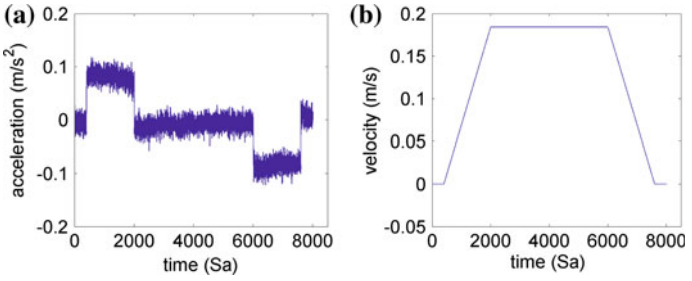


Fig. 4 State estimation by the Kalman Filter for typical accelerometer noise: **a** Estimated acceleration; **b** Estimated velocity

Despite the noise shown in Fig. 4a, there are no visible effects on the velocity trajectory (Fig. 4b). As compared to the Luenberger observer, a considerable improvement in the assessments of the variables can be observed in the KF-based system. The accuracy of the positioning itself is, however, satisfactory in both cases.

4.3 Sensitivity Study

In order to evaluate the elaborated robot-position estimation systems in various modeling and measurement noise uncertainties, the performance of the KF-based system was simulated in the presence of very high noise with a random walk sequence (drift presented in Fig. 5a). As can be clearly seen in Fig. 5b, the estimated velocity is also strongly distorted. Nevertheless, despite the applied high level of the noise (of the order of 100 m/s^2) the order of the magnitude of the estimated velocity is still proper.

In our further research the efficiency of the Luenberger observer and the Kalman filter was examined in the presence of various higher white noise and random walk. The resulting positional errors of both the observers are illustrated in Fig. 6. The error was calculated by averaging displacement after 10 s for 20 realizations of the noise sequences. The Kalman filter shows a noticeable improvement in performance

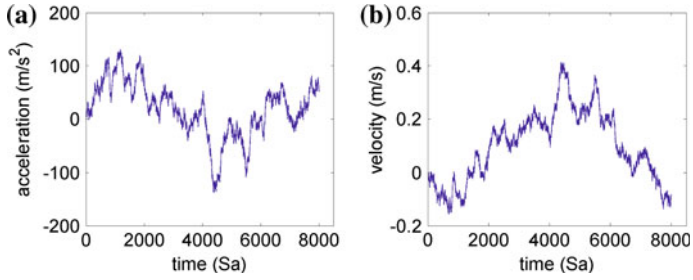


Fig. 5 State estimation by the Kalman Filter for very high noise in an accelerometer: **a** Very high noise; **b** Estimated velocity

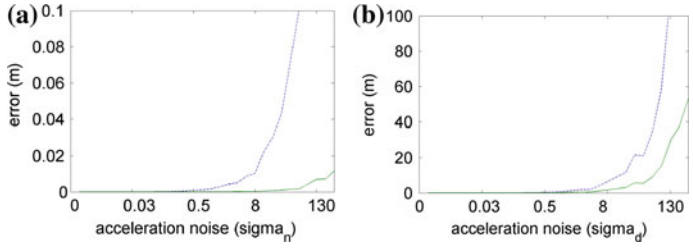


Fig. 6 Sensitivity study for the two observers: the Kalman filter (*solid green*) and the Luenberger observer (*dashed blue*): **a** Position error for variable white noise variance; **b** Position error for variable random walk variance (6)

as compared to the Luenberger observer. It is, however, worth noting that the two discrete-time observers are characterized by small position errors for small noise-signal modeling uncertainties.

5 Conculsions

The developed model of a mobile robot driven by a DC motor and applied in the observation system quite satisfactorily restricts the errors connected with the known disadvantageous accelerometer data characteristics. Modeling using the electric input signal that controls the robot brings about the removal of the influence of low frequency drift. The proposed observer systems appears to be robust to measurement noise typical for digital MEMS accelerometers. Despite the noticeable errors in the estimated state vector the error in position appears to be acceptable.

The usage of Kalman filter evidently improves the estimated quantities. Even a high level of noise sequences/processes (as compared to typical noise of low-cost MEMS accelerometers) does not devastate the results of position determination, what happens when using signal processing methods which are not based on a system model [9]. The model robustness to accelerometer noises is very promising for our future work with the VisRobot hardware system of computer/robot vision which is founded on 2-wheel electrically-driven mobile robots.

References

1. Agrawal, D.K., Woodhouse, J., Seshia, A.A.: Modeling nonlinearities in MEMS oscillators. IEEE Trans. Ultrason. Ferroelectr. Freq. Control **60**(8), 1646–1659 (2013)
2. Albarbar, A., Badri, A., Sinha, J.K., Starr, A.: Performance evaluation of MEMS accelerometers. Meas.: J. Int. Meas. Confed. **42**(5), 790–795 (2009)
3. Balasubramanian, R., Howe, R., Matsuoka, Y.: Task performance is prioritized over energy reduction. IEEE Trans. Biomed. Eng. **56**, 1310–1317 (2008)
4. Beeby, S.P., Ensel, G., Kraft, M., White, N.M.: MEMS Mechanical Sensors. Artech House, Norwood (2004)
5. Chapuis, Y.A., Zhou, L., Fukuta, Y., Mita, Y., Fujita, H.: FPGA-based decentralized control of arrayed MEMS for microrobotic application. IEEE Trans. Ind. Electron. **54**, 1926–1936 (2007)
6. Fleps, M., Mair, E., Ruepp, O., Suppa, M., Burschka, D.: Optimization based IMU camera calibration. In: International Conference on Intelligent Robots and Systems, pp. 3297–3304 (2011)
7. Gilbert, H., Celik, O., O'Malley, M.: Long-term double integration of acceleration for position sensing and frequency domain system identification. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, July 2010, pp. 453–458 (2010)
8. Khoo, S.Y., Khosla, P.K., Riviere, C.N.: Physical model of a MEMS accelerometer for low-g motion tracking applications. Int. Conf. Robot. Autom. **2**, 1345–1351 (2004)
9. Kowalczuk, Z., Merta, T.: Modelling an accelerometer for robot position estimation. In: 19th International Conference On Methods and Models in Automation and Robotics, September 2014, pp. 909–914 (2014)

10. Leniowski, R., Leniowska, L.: In: International Conference on Methods and Models in Automation and Robotics, pp. 508–513 (2013)
11. Lewis, C.P., Ball, R.: An instrument for the measurement of structural vibrations. J. Strain Anal. Eng. Des. **14**, 165–169 (1979)
12. Madgwick, S.O.H., Harrison, A.J.L., Vaidyanathan, A.: Estimation of IMU and MARG orientation using a gradient descent algorithm. In: International Conference on Rehabilitation Robotics, January 2011, pp. 1–7. IEEE (2011)
13. Maki, Y., Kagami, S., Hashimoto, K.: Localization and tracking of an accelerometer in a camera view based on feature point motion analysis. In: Proceedings of SICE Annual Conference, pp. 293–294 (2012)
14. Mohd-Yasin, F., Nagel, D.J., Korman, C.E.: Noise in MEMS. Meas. Sci. Technol. **21**(1), 12–21 (2009)
15. Pang, G., Liu, H.: Evaluation of a low-cost MEMS accelerometer for distance measurement. J. Intell. Robot. Syst. **30**(3), 249–265 (2001)
16. STMicroelectronics: Sensor Module LSM303 Datasheet. Technical Report Doc ID 023312 Rev 1, STMicroelectronics (2011)
17. Szeliski, R.: Computer Vision. Algorithms and Applications. Springer, Berlin (2010)
18. Wang, Y., Adams, S., Thorp, J., MacDonald, N., Hartwell, P., Bertsch, F.: Chaos in MEMS, parameter estimation and its potential application. IEEE Trans. Circuits Syst. I: Fundam. Theory Appl. **45**, 1013–1020 (1998)
19. Wolm, P., Chen, X., Chase, J., Pettigrew, W., Hann, C.: Analysis of a PM DC motor model for application in feedback design for electric powered mobility vehicles. In: 15th International Conference on Mechatronics and Machine Vision in Practice, December 2008, pp. 640–645 (2008)
20. Woodman, O.J.: An introduction to inertial navigation. Technical Report 696, University of Cambridge, Computer Laboratory, Cambridge, UK (2007)
21. Xiang, C., Wang, X., Ma, Y., Xu, B.: Practical modeling and comprehensive system identification of a BLDC motor. In: Mathematical Problems in Engineering, pp. 1–11 (2015)
22. Zainuddin, M., Ambak, N., Yahya, M., Saadon, M.: Acceleration due to gravity changes during solar eclipse phases. In: 2011 IEEE International Conference on Space Science and Communication, July 2011, pp. 170–173 (2011)

Decentralized Scheduling of Sensor Networks for Parameter Estimation of Spatio-Temporal Processes

Adam Romanek, Maciej Patan and Damian Kowalów

Abstract The activation scheduling problem for a scanning sensor network monitoring a spatio-temporal process is considered. The configuration of an activation schedule for network nodes measuring the system state is formulated in a sense of a suitable criterion quantifying an estimation accuracy for system parameters. Then, a decomposition of the scheduling problem is provided and a proper distribution of total computational effort and consensus between the network nodes is achieved via information flooding based on a pairwise communication scheme. As a result, a simple exchange algorithm is outlined to solve the design problem in a decentralized fashion. The proposed approach is illustrated on an example of sensor network configuration for monitoring an atmospheric pollution transport process.

Keywords Sensor networks · Distributed parameter systems · Parameter estimation · Optimum experimental design

1 Introduction

Requirements imposed by modern process control in the area of spatio-temporal physical systems also called distributed parameter systems (DPSs) are associated with using very accurate models in which spatial dynamics cannot be neglected and has to be included in addition to the temporal one. Such processes are usually described by systems of partial differential equations (PDEs) as the lumped descriptions often become unsatisfactory for the purpose of monitoring and control. One of the crucial problems encountered while trying to design an experimental set-up for

A. Romanek (✉) · M. Patan · D. Kowalów
University of Zielona Góra, Zielona Góra, Poland
e-mail: a.romanek@weit.uz.zgora.pl

M. Patan
e-mail: m.patan@issi.uz.zgora.pl

D. Kowalów
e-mail: d.kowalow@issi.uz.zgora.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_11

parameter estimation of a distributed system is an appropriate configuration of the monitoring system in terms of a proper selection of both, the sensor spatial locations and time intervals to take observations. This question becomes of vital importance in the context of recent advances in distributed sensor networks (SNs) which constitute a natural tool for monitoring distributed systems [3, 6, 7, 13, 22]. On one hand, SNs have great potential to revolutionize observation systems, but on the other completely new challenges related to design problems have to be solved.

Over past years, a laborious research on the development of strategies for efficient sensor placement has been conducted, but still the number of sensor placement techniques developed to manage the problems of practical scale is very limited (cf. [8, 13, 20, 22]). However, some effective approaches have been proposed to cover various experimental settings, including stationary [9, 14, 23], scanning [10–12, 16, 25] or moving observations [4, 5, 13, 18, 21, 24, 26].

The main aim of this work is to extend the decentralized approach to scanning sensor configuration reported in [12] to the setting of multi-level sensor networks, where the observation system comprises multiple subnetworks and it is desired to activate only a subset of their nodes during a given time interval, while the other sensors remain dormant. The main idea here is based on the introduction of a hierarchical structure to the sensor network allowing for low-level optimization within the subnetworks and higher level communication with data exchange and distribution. Thus, we come up with a solution which shares the advantage of decentralized data propagation with reduced communication load between network nodes.

Additionally, the investigation explores one of the research ideas indicated in [19] and [15] to develop a more effective token exchange scheme in order to fully utilize each iteration of the algorithm. All that boils down to exchanging multiple tokens during each communication instead of one, as it is in the original concept. This makes possible to substantially increase the rate at which an acceptable solution to the problem at hand is established.

2 Optimal Scheduling Problem

Let $y = y(x, t; \theta)$ be the scalar state of a given DPS at a spatial point $x \in \Omega \subset \mathbb{R}^d$ and time instant $t \in T = [0, t_f]$, $t_f < \infty$. Here θ represents an unknown constant m-dimensional parameter vector which must be estimated using observations of the system. Further, assume that the state y is observed directly by N pointwise sensors, from among only n are activated at time instants $0 < t_0 < t_1 < \cdots < t_K = t_f$ and will collect continuous measurements for the duration of each subinterval $T_k = (t_{k-1}, t_k]$, $k = 1, \ldots, K$. This activation schedule (also called 'scanning') leads to the collection of observations described by the statistical measurement model in the form

$$z_{\mathrm{m}}^{\ell}(t) = y(x_k^{\ell}, t; \theta) + \varepsilon(x_k^{\ell}, t), \quad t \in T_k, \quad \ell = 1, \ldots, n, \quad k = 1, \ldots K \tag{1}$$

where $z_{\mathrm{m}}^{\ell}(t)$ is the scalar output and $x_k^{\ell} \in X$ stands for the location of the ℓth sensor at time subinterval T_k, X signifies the part of the spatial domain Ω where the measurements can be made and $\varepsilon(x_k^{\ell}, t)$ denotes the measurement noise, which is customarily assumed to be zero-mean, Gaussian, spatial uncorrelated and white [13, 22].

Given the model response $y(x_k^{\ell}, t; \theta)$ and the outcomes of the measurements $z_{\mathrm{m}}^{\ell}(\cdot)$, $\ell = 1, \ldots, n$ on time intervals T_k, θ is estimated by $\widehat{\theta}$, a global minimizer of the output least-squares criterion [12, 22]. Unavoidably, the covariance matrix $\mathrm{cov}(\widehat{\theta})$ of this least-squares estimator depends on the active sensor locations x_k^{ℓ}. Therefore, choosing the right ones is the key to success in obtaining good estimates of the system parameters. In order to compare individual locations, a quantitative measure of the 'goodness' of particular sensor configurations is required. Such a measure Ψ is customarily based on the concept of the *Fisher Information Matrix* (FIM) which is widely used in optimum experimental design theory for lumped systems [1, 22] as its inverse constitutes a good approximation of $\mathrm{cov}(\widehat{\theta})$.

The optimal sensor scheduling problem consists in seeking in each time subinterval T_k the best subset of n locations from among the N given potential ones. More precisely, the problem is to divide for each time subinterval the N available sensor nodes into n active ones and the remaining $N - n$ dormant ones, so as to maximize the criterion (4) associated with the parameters to be estimated. Let us introduce each possible location x^i $(i = 1, \ldots, N)$ a set of variables v_k^is, each of them taking the value 1 or 0 depending on whether or not a sensor residing at x^i is activated during T_k. Therefore, in our setting, owing to the character of noise in (1), the FIM is given by [12]

$$M(v^1, \ldots, v^N) = \sum_{i=1}^{N} \sum_{k=1}^{K} v_k^i M_k(x^i) \tag{2}$$

where $v^i = (v_1^i, \ldots, v_K^i)$, $M_k(x^i) = \frac{1}{t_f} \int_{T_k} g(x^i, t) g^{\mathsf{T}}(x^i, t)\, \mathrm{d}t$, and

$$g(x, t) = \left[\frac{\partial y(x, t; \vartheta)}{\partial \vartheta_1}, \ldots, \frac{\partial y(x, t; \vartheta)}{\partial \vartheta_m} \right]^{\mathsf{T}}_{\vartheta = \theta^0} \tag{3}$$

stands for the so-called *sensitivity vector*. Since in the nonlinear case g depends on the estimated parameters, some preliminary estimate θ^0 is required for its calculation. Usually some known nominal values of the parameters θ can be used or we can apply estimates obtained from previous experiments [12, 17, 22]. As for Ψ the most common choice used in applications is the so-called D-optimality criterion

$$\Psi(M) = \log \det(M) \tag{4}$$

The interpretation of the resulting D-optimum sensor configuration is that it leads to the minimum volume of the uncertainty ellipsoid for the system parameter estimates.

Thus, the design problem can be formalized as follows:

Problem 1 Find $v = (v^1, \ldots, v^N)$ maximizing $\mathcal{P}(v) = \Psi(M)$, subject to

$$\sum_{i=1}^{N} v_k^i = n, \quad k = 1, \ldots, K \tag{5}$$

$$v_k^i \in \{0, 1\}, \quad i = 1, \ldots, N, \quad k = 1, \ldots, K \tag{6}$$

Problem 1 is in fact a discrete programming task which needs efficient solutions. However, if the FIM corresponding to an optimal solution $v^\star = (v^{1\star}, \ldots, v^{N\star})$ is nonsingular, then the optimality conditions take the following form:

Proposition 1 *Suppose that the matrix $M(v^\star)$ is nonsingular. The vector $v^\star$ is a global solution to Problem 1 iff there exist numbers $\mu_k^\star$, $k = 1, \ldots, K$ such that*

$$\phi(i, k, v^\star) \begin{cases} \geq \mu_k^\star & \text{if } v_k^{i\,\star} = 1 \\ \leq \mu_k^\star & \text{if } v_k^{i\,\star} = 0 \end{cases} \tag{7}$$

where

$$\phi(i, k, v) = \operatorname{tr}\left[M^{-1}(v) M_k(x^i)\right] \tag{8}$$

The proof can be obtained directly from a more general result provided by Patan [13] (Proposition 6.1, p. 162). The function ϕ is widely known in the theory of experimental design and has the interpretation of the variance of the prediction of the system model response. Proposition 1 reveals one characteristic features of the optimal solutions, namely that, when identifying them, the function ϕ turns out to be crucial and optimality means separability of the components in terms of the values of this function. From a practical point of view, the above result means that at all the active pairs (i, k) (with $v_k^i = 1$) of an optimal schedule $v^\star$ the mapping $\phi(\cdot, \cdot, v^\star)$ should be greater than anywhere else, i.e., preferably active locations of $v^\star$ should coincide with maximum points of $\phi(\cdot, \cdot, v^\star)$. In practice, this amounts to allocating observations to spatial points and time subintervals at which we know least about the system response.

3 Decentralized Multi-exchange Algorithm

In order to demonstrate the usefulness of the above-mentioned idea, a numerical algorithm needs to be implemented. Although, there exist some approaches to effectively solve Problem 1 based on its continuous approximation and making use of the notion of the so-called directly constrained design measures (see [25] and further refinements [13, 22]), their fully centralized character makes them very vulnerable with respect to failures of individual network nodes. On the other hand, the fully

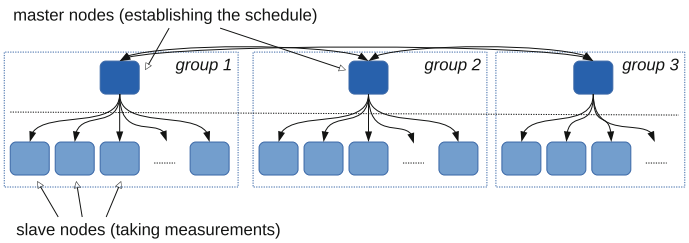


Fig. 1 Two level structure of the sensor network nodes partitioned into groups

distributed approach of [12] with computations spread over the network nodes may suffer in some situations from an excessive number of communications between network nodes required for convergence. In order to fill this gap and develop a procedure providing a compromise between a reasonable rate of communication and the distributed scheme of computations a numerical scheme is proposed here, being a fusion of the fully centralized and fully distributed approaches.

The central idea behind a good 'trade-off' is to form a two-level structure of the sensor network nodes as it is illustrated in Fig. 1. First, we introduce the partitioning of the network into G disjoint groups of sensors (subnetworks) with N_p sensors in the pth group, in such a way that $\sum_{p=1}^{G} N_p = N$. The optimal partitioning is beyond the research reported here, so we assume that this is done arbitrarily (in fact, in many applications we encounter a fixed network infrastructure). It is worth to note that the number of groups G as well as a particular partitioning scheme may significantly influence the performance or even the ability to obtain an optimal solution, so the issue is of paramount importance and should be taken into consideration separately. The subnetworks are the result of the partitioning of network nodes form the lower level of our structure. Next, in each such a group we choose one superior entity further called the *master node*, responsible for establishing the observation schedule within the scope of its own group. The master nodes form the higher level of the network which serves as a routing layer for exchanging data between subnetworks, store the local activation schedules and finally perform all the computations. The network nodes from the lower level within each group, also referred to as *worker nodes*, communicate only with their master node to upload sensor readings during the experiment. Thus, they do not take part in the network configuration process. In such a way, we obtain a mixed structure: centralized on the lower level of groups and decentralized on the upper level of master nodes responsible for distributing computations.

In the following we assume the asynchronous time model for the configuration process. Let $r = 0, 1, 2, \ldots$ be the discrete time index, which is used to partition the time horizon into time slots $Z_r = (z_{r-1}, z_r]$. Owing to Proposition 1, $v_k^\star$ should be nonzero in the areas where $\phi_k(\cdot, \cdot, v^\star)$ takes on larger values. Thus, the central idea when constructing a computational algorithm for sensor density optimization is to move at iteration r some observational effort from areas with smaller values

of $\phi(\cdot, \cdot, v^{(r)})$ to those with larger values, as we expect that such a procedure will improve the current solution $v^{(r)}$. The only component of $\phi(\cdot, \cdot, v^{(r)})$ which cannot be calculated independently of other nodes is the global information matrix (2) being a weighted average of the local information matrices $M_k(x^i)$. In such a way, our task is closely related to the problem of distributed averaging on a sensor network [2] whose primary difficulty consists in defining the scheme for exchanging information between the nodes.

One of the most commonly known techniques of distributed averaging is a pairwise communication flooding, also known as a *gossip* scheme. In typical applications the procedure is that at the rth time slot the pth sensor contacts some neighboring node q with probability P_{pq}, i.e. a pair $(p; q)$ is randomly and independently selected. When both nodes communicate with each other, they exchange the data they currently store and reset their values to the average of their current states.

Since not all the nodes contribute to the global estimate of the FIM at the rth configuration slot, we introduce an important enhancement to the classical distributed averaging problem. One of its elements is the introduction of local FIM estimates which are updated when nodes communicate with each other. However, to make this idea useful, the nodes need to store the global activation schedule $v^{(r)}$ and be equipped with mechanism of its update. This is achieved by the exchange of *tokens* representing the activation of the sensors at given subintervals T_k. Such tokens are transferred between nodes in the situation when a neighbor node at a specific observation subinterval T_k is more informative in the sense of the function ϕ calculated on the current estimates of the FIM (and, obviously, it is not activated yet). Such an approach has a very attractive property, namely that the individual elements comprising the schedule $v^{(r)}$ are distributed via tokens over the network of master nodes, therefore at the mth master node it is required to store only its part $v(m)^{(r)}$ related to the nodes of the mth subnetwork. This forms the core of the decentralized approach proposed here.

In [19] the simplest scheme of exchanging only one token per single communication between two master nodes gave evidence for the algorithm efficiency. The main contribution of this work is the introduction of a more complex scheme, which makes it possible to exchange multiple tokens between nodes of two communicating master nodes. The number of tokens that can be exchanged in each iteration of the algorithm is denoted as E. Given $E = 1$ we obtain the same characteristic of the algorithm as it is in [19]. Further, it can be shown that under a proper exchange of many tokens the algorithm is convergent with an arbitrary accuracy to the centralized version of the algorithm. Also, reasonably increasing the value of exchanged tokens E leads to a higher convergence rate making the network configuration process more effective.

Let M_j^m denotes the local estimate of the FIM from the jth subnetwork stored at the mth master node and r_j^m be the configuration time when M_j^m was updated for the last time. At $r = 0$ the sensor network starts with an arbitrary token allocation and the following initial values of the FIM estimates:

$$M_j^m = \begin{cases} \sum_{i=1}^{N_m} \sum_{k=1}^{K} v_k^i M_k(x^i), & j = m \\ \text{NULL}, & j \neq m \end{cases}$$

where NULL means that FIM estimate is unknown. In this way, at the mth master node the collection of pairs $M^m = \left(M_j^m, r_j^m\right)\right)_{j=1}^{G}$ is stored. Then, at each subsequent time slot Z_r a random pair (p, q) of master nodes performs communication. The scheme of calculations from the point of view of the pth node is embodied in Algorithm 1. The crucial operators of this procedure are:

- EXCHANGE stands for a pairwise duplex exchange of data between two master nodes
- RECENT is responsible for building a list of the most recent values from pairs generated by iterating element-wise along both operands and updating the older value of FIM (i.e. $M_j^p \leftarrow M_j^q$ if $r_j^p < r_j^q$ and $M_j^q \leftarrow M_j^p$ otherwise)
- AVG-NOT-NULL computes the average of input collection of information matrices rejecting those the NULL ones (NULL values are simply treated to be missing)
- PASS-TOKEN determines for each time subinterval T_k, the worst active sensor within the subnetworks p and q (in terms of the lowest current value of $\phi(\cdot, k, v^{(r)})$) and the best inactive sensor (in terms of the greatest current value of $\phi(\cdot, k, v^{(r)})$). If $\phi(\text{worst}, k, v^{(r)}) < \phi(\text{best}, k, v^{(r)})$ then $v_k^{\text{worst}} \leftarrow 0$ (deactivation of the worst active) and $v_k^{\text{best}} \leftarrow 1$ (activation of the best inactive).

The PASS-TOKEN operator is the one responsible for a single token exchange. Since we wanted to have greater control on this part of the algorithm, we introduce an additional loop in which the PASS-TOKEN operator is used E times in practice allowing multiple tokens to be exchanged.

Algorithm 1. *Distributed data exchange model. Indices p and q denote data from local repository and obtained from neighbor, respectively*

```
procedure EXCHANGE_PROTOCOL
    EXCHANGE(M^p, M^q)
    EXCHANGE(v(p), v(q))
    M^p ← RECENT(M^p, M^q)
    M_avg ← AVG-NOT-NULL(M^p)
    for k ← 1, K do
        for e ← 1, E do
            for ℓ ← 1, N_p do
                φ_k^ℓ ← tr [M_avg^{-1} M_k(x^ℓ)]
            end for
            EXCHANGE((φ_k^ℓ)_{ℓ=1}^{N_p}, (φ_k^ℓ)_{ℓ=1}^{N_q})
            PASS-TOKEN((φ_k^ℓ)_{ℓ=1}^{N_p}, (φ_k^ℓ)_{ℓ=1}^{N_q})
        end for
    end for
    M_g(i) = Σ_{i=1}^{N_G} Σ_{k=1}^{K} v_k(i) M_k(x(i))
    M_G^i(j) ← (M_g(i), r)
end procedure
```

4 Simulation Example

As an illustrative example consider the problem of sensor configuration for the purpose of parameter estimation in the process of air pollutant transport-chemistry over a given urban area Ω being a square with a side of length 1 km. Two active sources of pollution under the spatio-temporal changes of the wind velocity $v(x,t)$ affect the concentration of the pollutant $y = y(x,t)$ within the domain. The process over the observation interval $T = (0, 1000]$ (in seconds) can be mathematically described with following advection-diffusion-reaction equation:

$$\frac{\partial y(x,t)}{\partial t} + \nabla \cdot \big(v(x,t)y(x,t)\big) + \alpha y(x,t) = \nabla \cdot \big(\kappa \nabla y(x,t)\big) + f_1(x) + f_2(x), \quad x \in \Omega \tag{9}$$

subject to the boundary and initial conditions:

$$\frac{\partial y(x,t)}{\partial n} = 0 \text{ on } \partial\Omega \times T, \quad y(x,0) = y_0 \text{ in } \Omega \tag{10}$$

where the terms $f_\ell(x) = \mu_\ell \exp\big(-100\|x - \chi^\ell\|^2\big)$, $\ell = 1, 2$ represent the pollutant sources with emission intensities μ_ℓ located at the points $\chi^\ell = (\chi_1^\ell, \chi_2^\ell)$, $\ell = 1, 2$, and $\partial y/\partial n$ stands for the partial derivative of y with respect to the outward normal to the boundary $\partial\Omega$. Finally, $\alpha = 0.02\ \mathrm{s}^{-1}$ stands for the absorption rate modeling a slow decay of the pollutant and κ denotes an unknown turbulent diffusion coefficient. The complex changes in concentration according to various transport factors and the wind velocity field are illustrated in Fig. 2.

Given N possible sensor locations, our goal is to choose a subset consisting of n of them (separate for each time subinterval T_k), which will become active and take measurements leading to D-optimal estimates of unknown parameters θ of the distributed parameter system. It all boils down to solving Problem 1.

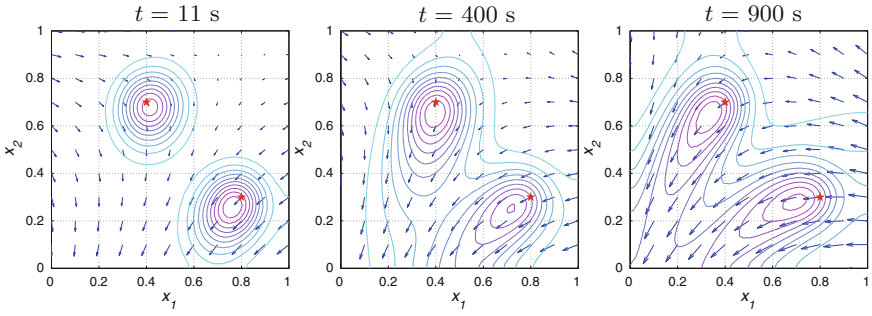


Fig. 2 Temporal changes in the wind velocity field and pollutant concentration (*stars* indicate the locations of pollutant sources)

For the purpose of this simulation we assumed that the observation horizon was partitioned into 5 even subintervals $T_k = (200(k-1), 200k]$, $k = 1, \ldots, 5$. We aimed at determining the locations of the pollutant sources, their emission intensities and the unknown diffusion coefficient denoted by the vector of unknown parameters $\theta = (\mu_1, \chi_1^1, \chi_2^1, \mu_2, \chi_1^2, \chi_2^2, \kappa)$. In order to estimate the parameter vector, a scanning sensor network was applied.

Given that the elements of the sensitivity vectors (3) are required in order to calculate the FIMs (2), we employed COMSOL MULTIPHYSICS 3.4 to solve our system of PDEs, with the initial values of the parameters to be identified assumed to be $\theta = (12\ \text{kg/s}, 0.4\ \text{km}, 0.7\ \text{km}, 15\ \text{kg/s}, 0.8\ \text{km}, 0.3\ \text{km}, 50\ \text{m}^2/\text{s})$. Calculations were performed for a spatial mesh composed of 978 triangles, 520 nodes and an evenly partitioned time interval (101 subintervals). Once the elements of the sensitivity vectors were obtained, we wrote a MATLAB program to verify Algorithm 1 in the context of the settings described above. The program was run in MATLAB 7 (R14) on a PC notebook equipped with Intel Core 2 Duo 2.53GHz processor and 4GB RAM running Ubuntu 14.04 (64-bit).

To provide some insight into the performance of the presented approach we investigated the following scenario. From among 520 nodes of the triangulation mesh only the points not situated on the outer boundary of the domain were selected as

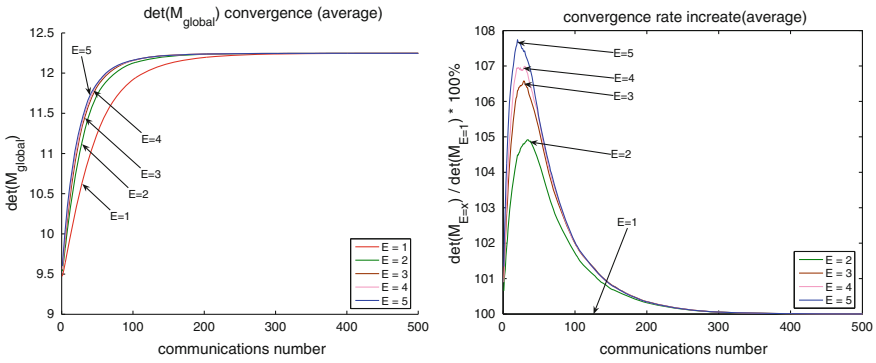


Fig. 3 Convergence of the global FIM determinants in absolute values (*left*) and the increase in convergence ratesshown as differences between the global FIM determinants for $E = 1$ and the other values of E (*right*)

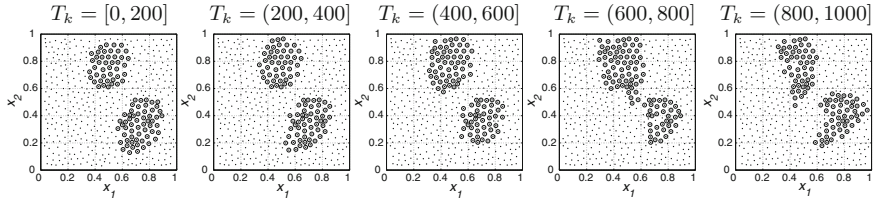


Fig. 4 Final allocation of active sensors at selected time subintervals for $E = 5$

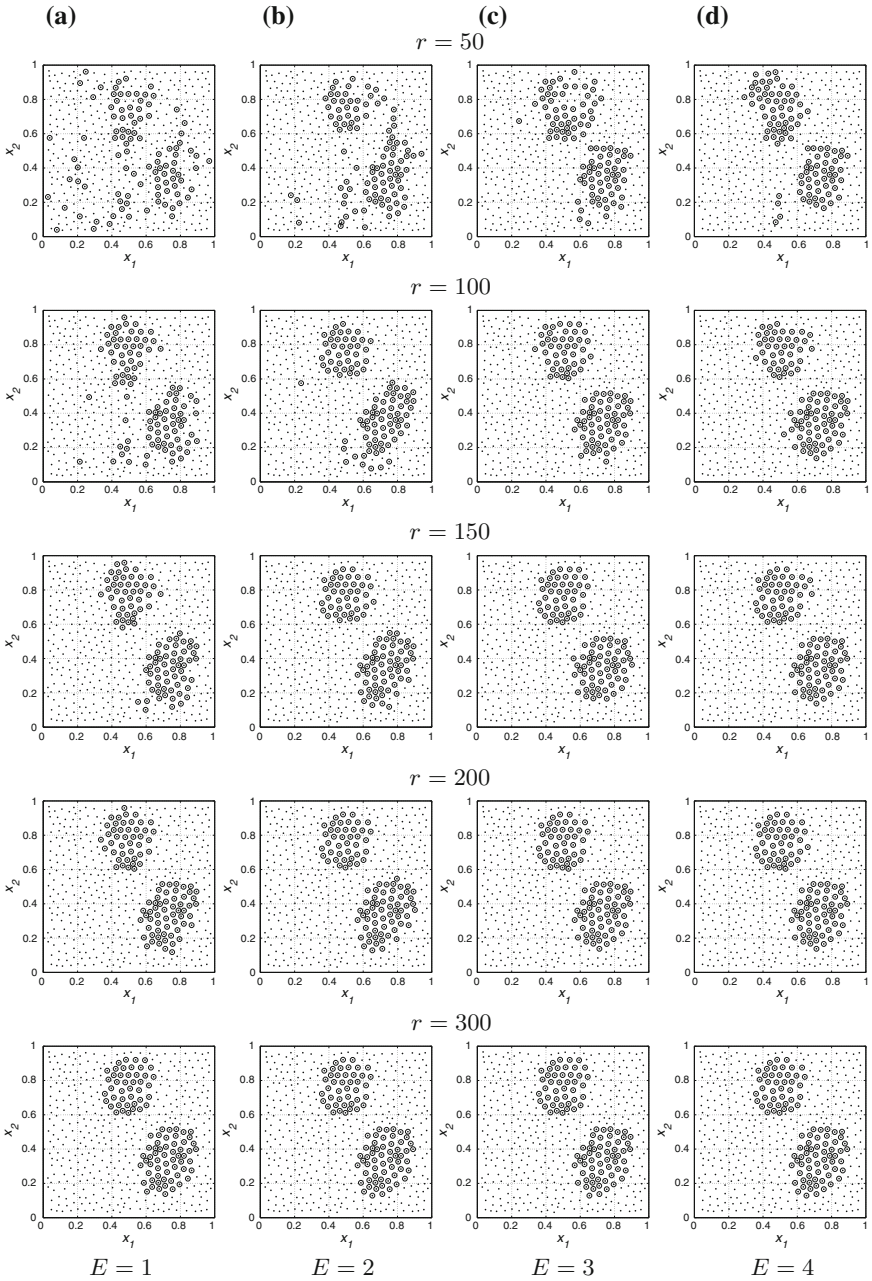


Fig. 5 Allocation of active sensors at first time subinterval ($T = [0, 200]$) in consecutive stages of network configuration for different values of the parameter E

prospective observation locations, giving a subset of 460 such points. Our task was to determine a D-optimal activation schedule using a subset of $n = 100$ out of $N = 460$ sensors that would become active and take measurements during each time subinterval T_k. The initial activation schedule was chosen randomly and was the same for each time subinterval T_k. Furthermore, it was assumed that all master nodes form a fully connected communication graph with uniform probability distribution of a single communication between a given pair of them.

An experiment was carried out to prove our initial assumption on the increase in the convergence rate with the increase of parameter E was true. The sensor nodes were arbitrarily divided into $G = 21$ groups. Further, the groups were assumed to be of similar cardinality, i.e. 21–22 nodes per group. Specifically, the assignment to distinct groups was defined as follows: $S_i = i \bmod G, i = 1, \ldots, N$, where S_i denotes the index of a group the ith sensor belongs to. Finally, the approach was verified with a set of possible values of the parameter E, namely the maximum number of tokens that can be exchanged in a single communication between a given pair of master nodes, i.e. $E = 1, \ldots, 5$. The algorithm described in Sect. 3 produced an acceptable solution after ca. 400 iterations which in practice took less than 5 seconds. As it is shown in Fig. 3 the convergence rate increases with an increase in the value of parameter E. As it was expected, the highest increase is observed in the early stage of the computations (ca. 50 iterations) as the activation schedule is far from optimal giving a lot of room for improvement by utilizing more token exchanges in a single iteration. Later on, when the schedule becomes close to the optimal, fewer tokens can be exchanged. The simulations provide the evidence that for distinct values of E the algorithm converges to the same final solution. The evolution of the sensor allocation schedule for different values of E is shown in Figs. 4 and 5.

5 Conclusions

The problem of decentralized sensor network scheduling for accurate parameter estimation of distributed-parameter systems has been investigated. Specifically, the idea of increasing the convergence rate of an algorithm, by allowing for an exchange of multiple activation tokens for worker nodes during a single pairwise communication between two master nodes has been put into practice. The main contribution here is a proper characterization of the data exchange protocol and the analysis of its characteristics related to the convergence rate. As a result, a very effective network configuration procedure is developed in terms of both, the robustness with respect to node failures and the number of communications required for convergence.

Nevertheless, some issues still remain addressed. In the future, the influence of the number of groups the sensor network is divided into has to be investigated. Another open problem is the necessity for optimizing the connection probabilities for particular master nodes in the fixed infrastructure in order to further increase the convergence rate.

Acknowledgments This work is partially supported by National Science Center in Poland under grant 2014/15/B/ST7/03208.

References

1. Atkinson, A.C., Donev, A.N., Tobias, R.D.: Optimum Experimental Designs, with SAS. Oxford University Press, Oxford (2007)
2. Boyd, S., Ghosh, A., Prabhakar, B., Shah, D.: Randomized gossip algorithms. IEEE Trans. Inf. Theory **52**(6), 2508–2530 (2006)
3. Cassandras, C.G., Li, W.: Sensor networks and cooperative control. Eur. J. Control **11**(4–5), 436–463 (2005)
4. Demetriou, M.A., Hussein, I.: Estimation of spatially distributed processes using mobile spatially distributed sensor Network SIAM. J. Control Optim. **48**(1), 266–291 (2009)
5. Jeremić, A., Nehorai, A.: Landmine detection and localization using chemical sensor array processing. IEEE Trans. Signal Process. **48**(5), 1295–1305 (2000)
6. Martínez, S., Bullo, F.: Optimal sensor placement and motion coordination for target tracking. Automatica **42**, 661–668 (2006)
7. Bauer, P.H.: New challenges in dynamical systems: the networked case. Int. J. Appl. Math. Comput. Sci. **18**(3), 271–277 (2008)
8. Kubrusly, C.S., Malebranche, H.: Sensors and Controllers Location in Distributed Systems - A Survey. Automatica **21**(2), 117–128 (1985)
9. Nehorai, A., Porat, B., Paldi, E.: Detection and localization of vapor-emitting sources. IEEE Trans. Signal Process. **43**(1), 243–253 (1995)
10. Patan, M.: Optimal activation policies for continuous scanning observations in parameter estimation of distributed systems. Int. J. Syst. Sci. **37**(11), 763–775 (2006)
11. Patan, M.: A Parallel sensor scheduling technique for fault detection in distributed parameter systems. In: Lecture Notes in Computer Science: Euro-Par 2008: Parallel Processing, vol. 5168, pp. 833–843 (2008)
12. Patan, M.: Distributed scheduling of sensor networks for identification of spatio-temporal processes. Int. J. Appl. Math. Comput. Sci. **22**(2), 299–311 (2012)
13. Patan, M.: Optimal sensor networks scheduling in identification of distributed parameter systems. Lecture Notes in Control and Information Sciences. Springer, Berlin (2012)
14. Patan, M., Uciński, D.: Configuring a sensor network for fault detection in distributed parameter systems. Int. J. Appl. Math. Comput. Sci. **18**(4), 513–524 (2008)
15. Patan, M., Romanek, A.: Decentralized time-constrained scheduling for sensor network in identification of distributed parameter systems. In: Stochastic Models, Statistics and Their Applications, vol. 122, pp. 415–423. Springer International Publishing (2015). doi:10.1007/978-3-319-13881-7, Wrocław, Poland
16. Patan, M., Uciński, D.: Time-constrained sensor scheduling for parameter estimation of distributed systems. In: 49th IEEE Conference on Decision and Control, pp. 7–12. Atlanta, USA (2010)
17. Rafajłowicz, E.: Optimal experiment design for identification of linear distributed-parameter systems: frequency domain approach. IEEE Trans. Autom. Control **28**(7), 806–808 (1983)
18. Rafajłowicz, E.: Optimum choice of moving sensor trajectories for distributed parameter system identification. Int. J. Control **43**(5), 1441–1451 (1986)
19. Romanek, A., Patan, M.: Decentralized multi-exchange scheduling of sensor networks for parameter estimation of distributed systems. In: 19th International Conference on Methods and Models in Automation and Robotics, pp. 624–629. Miedzyzdroje, Poland (2014)
20. van de Wal, M., de Jager, B.: A review of methods for input/output selection. Automatica **37**, 487–510 (2001)

21. Tricaud, C., Patan, M., Uciński, D., Chen, Y.: D-optimal trajectory design of heterogeneous mobile sensors for parameter estimation of distributed systems. In: American Control Conference, Seattle, WA, pp. 11–13 (2008)
22. Uciński, D.: Optimal Measurement Methods for Distributed-parameter System Identification. CRC Press, Boca Raton (2005)
23. Uciński, D.: Sensor network scheduling for identification of spatially distributed processes. Int. J. Appl. Math. Comput. Sci. **22**(1), 25–40 (2012)
24. Uciński, D., Chen, Y.: Time-optimal path planning of moving sensors for parameter estimation of distributed systems. In: 44th IEEE Conference on Decision and Control and European Control Conference, Seville, Spain (2005)
25. Uciński, D., Patan, M.: Optimal location of discrete scanning sensors for parameter estimation of distributed systems. In: 15th IFAC World Congress, Barcelona, Spain, July 2012, pp. 22–26 (2002)
26. Uciński, D., Patan, M.: Sensor network design for the estimation of spatially distributed processes. Int. J. Appl. Math. Comput. Sci. **20**(3), 459–481 (2010)

Part III
Robust and Fault Tolerant Control

MPC Framework for System Reliability Optimization

Jean C. Salazar, Philippe Weber, Fatiha Nejjari, Didier Theilliol and Ramon Sarrate

Abstract This work presents a general framework taking into account system and components reliability in a Model Predictive Control (MPC) algorithm. The objective is to deal with a closed-loop system combining a deterministic part related to the system dynamics and a stochastic part related to the system reliability from an availability point of view. The main contribution of this work consists in integrating the reliability assessment computed on-line using a Dynamic Bayesian Network (DBN) through the weights of the multiobjective cost function of the MPC algorithm. A comparison between a method based on the components reliability (local approach) and a method focused on the system reliability sensitivity analysis (global approach) is considered. The effectiveness and benefits of the proposed control framework are presented through a Drinking Water Network (DWN) simulation.

Keywords Reliability · Model predictive control · Dynamic Bayesian network

This work has been funded by the Spanish Ministry of Economy and Competitiveness through the CICYT project SHERECS (ref. DPI2011-26243), and by the European Commission through contract EFFINET (ref. FP7-ICT2011-8-318556).

J.C. Salazar (✉) · F. Nejjari · R. Sarrate
Universitat Politècnica de Catalunya, Research Center for Supervision Safety and Automatic Control (CS2AC), 10 Rambla Sant Nebridi, Terrassa, Spain
e-mail: jean.salazar@upc.edu

F. Nejjari
e-mail: fatiha.nejjari@upc.edu

R. Sarrate
e-mail: ramon.sarrate@upc.edu

P. Weber · D. Theilliol
Université de Lorraine, Centre de Recherche En Automatique de Nancy (CRAN), FST - B.P. 70239, 54506 Vandoeuvre-lès-Nancy, France
e-mail: philippe.weber@univ-lorraine.fr

D. Theilliol
e-mail: didier.theilliol@univ-lorraine.fr

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_12

1 Introduction

To improve the system reliability and minimize operational costs, component health monitoring should be considered in a controlled system [14]. After failure, the control effort can be redistributed among the available actuators to alleviate the work load and the stress factors on equipments with worst conditions avoiding in this manner their break down [2, 3]. For this purpose, an appropriate method should be developed to redistribute this control effort until maintenance actions can be taken. Over-actuated systems implies actuator redundancy, this means that the number of control inputs is larger than the number of inputs necessary to satisfy the control objective. This characteristic allows for several combinations of control inputs that produce the same desired output and provide the same system performance [13].

Model Predictive Control (MPC) is an efficient technique to manage actuator redundancy [18]. The MPC algorithm allows to include several criteria in the optimization problem. For example in [10] the authors present an application of MPC to a Drinking Water Network (DWN) taking into account economic, service level and degradation criteria. In [23], MPC formulation includes as a criteria the accumulated actuator usage. Its objective is to maintain the accumulated usage under a safe level at the end of the mission. This approach is exported to several applications, like in [24] where a MPC is applied to a two degree of freedom helicopter and the weighting parameters are used to maintain the degradation of actuators within a safe level.

Reliability is the ability of a system to operate successfully long enough to complete its assigned mission under stated conditions. It can be modelled as an exponential function [7, 30], a Weibull function [3, 12] or a Gamma function [15, 17, 20], among others.

System reliability can be expressed as a stochastic process [22]. For example it is common to use Markov Chains (MC) to model the reliability of components [29]. Unfortunately, in practice the complexity of the system leads to a combinatorial explosion of states resulting in a MC with a very large size.

The use of Bayesian Networks (BN) as a modelling method for reliability computing, taking into account observations (evidences) about the state of the components, has been recently considered in some works [1, 3, 24, 26, 27].

The application of BNs to reliability started at the end of 90's. In [25] the authors present the advantages of BNs in comparison with reliability block diagrams (RBD). In [5] the authors propose to model a fault tree using BN. A comparison between MC and Dynamic Bayesian Networks (DBN) application to reliability is presented in [28]. In this work BN is used to model the global system reliability. DBNs are interesting because they allow to model the system reliability with a factorization of the MC states leading to a compact model.

The main objective of this MPC framework is to preserve the system reliability providing control performance. Two approaches are proposed to achieve the goal. A local approach, which is focused on the component reliability, and a global approach, that is focused on the system reliability.

From an availability point of view, the objective is to deal with a closed-loop system combining a deterministic part related to the system dynamics and a stochastic part related to the actuators and system reliability. The effectiveness and benefits of the proposed control framework are shown by its application on a drinking water network.

The rest of the work is organized as follows: Sect. 2 deals with reliability and its modelling using a DBN. Section 3 describes the formulation used in the MPC problem. Section 4 presents the case study, and the proposed MPC formulation with reliability integration is given. In Sect. 5 some results of the control application are discussed. Finally, in Sect. 6 some conclusions are provided.

2 Reliability Modelling

2.1 Bayesian Network Framework

Basically, BNs compute the probability distribution in a set of variables according to the prior knowledge of some variables and the observation of others [11]. For instance, let **A** and **B** be two nodes with two possible states ($\mathbf{S}_1$ and $\mathbf{S}_2$) as is shown in Fig. 1. A probability is associated to each state of the node and this probability is defined *a priori* for a root node and computed by inference for the others. The *a priori* probabilities of node **A** are $P(\mathbf{A} = \mathbf{S}_{A1})$ and $P(\mathbf{A} = \mathbf{S}_{A2})$.

A Conditional Probability Table (CPT) is associated node **B** and defines the probability $P(\mathbf{B}|\mathbf{A})$ of each state of **B** given the states of **A**. Thus, the BN inference computes the marginal distribution $P(\mathbf{B} = \mathbf{S}_{B1})$:

$$\begin{aligned} P(\mathbf{B} = \mathbf{S}_{B1}) = P(\mathbf{B} = \mathbf{S}_{B1}|\mathbf{A} = \mathbf{S}_{A1})P(\mathbf{A} = \mathbf{S}_{A1}) \\ + P(\mathbf{B} = \mathbf{S}_{B1}|\mathbf{A} = S_{A2})P(\mathbf{A} = \mathbf{S}_{A2}) \end{aligned} \tag{1}$$

It is possible then to compute the probability distribution for each variable conditioned by the values of the other variables in the graph.

2.2 System Reliability

System reliability can be computed from its components reliability through a BN. For this purpose it is necessary to identify all the minimal success paths present in the system.

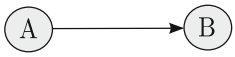


Fig. 1 Basic Bayesian network

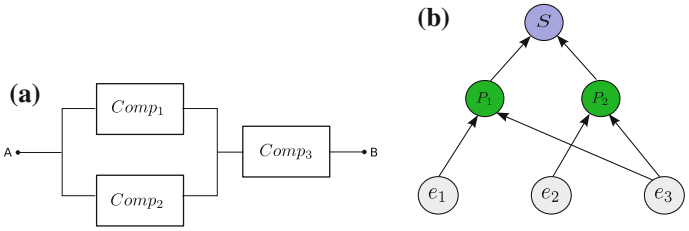


Fig. 2 Three components system example: (**a**) Minimal paths; (**b**) Bayesian network of system reliability

A minimal success path is a minimum set of components whose functioning (i.e. being up) ensures that the system is up (if all elements of the minimal success path are "Up" then the system is up). The minimal success path cannot be reduced.

For example, consider the system reliability block diagram shown in Fig. 2a, composed of three components. It is clear that with a minimum of two components (i.e. components 1 and 3 or components 2 and 3) it can perform its function satisfactorily.

With the minimal success paths identified, it is possible then to build the BN of the system. Figure 2b shows the BN that corresponds to the three components example. The components reliabilities are represented by nodes e_i. Those nodes are connected to their corresponding minimal success paths represented by nodes P_i. Finally, the system state (*Up*, *Down*) is represented by the binary random variable S [27].

Table 1 shows the CPT of node P_1. It depends on the states of the components e_1 and e_3 and its behavior corresponds to an AND gate, i.e. all the components in a success path should be available for the system to be available.

The CPT of node S is also shown in Table 2. It depends on the state of success path nodes P_1 and P_2 and has the behavior of an OR gate, i.e. if there is at least one success path available, then the system will be available. Consequently, the reliability of the system is expressed as:

$$R_s = \mathrm{P}(S = Up) \tag{2}$$

Table 1 CPT for node P_1

e_1	e_3	P_1	
		Up	*Dn*
Up	*Up*	1	0
Up	*Dn*	0	1
Dn	*Up*	0	1
Dn	*Dn*	0	1

Table 2 CPT for node S

P_2	P_3	S	
		Up	*Dn*
Up	*Up*	1	0
Up	*Dn*	1	0
Dn	*Up*	1	0
Dn	*Dn*	0	1

2.3 Importance Factors

In order to take into account the global reliability of the system, the use of an importance factor is proposed. Importance factors are criteria used as an evaluation of the components impact over the system. One of them involves determining the system reliability sensitivity against changes in the reliability of the *i*th component, also known as the Marginal Importance Factor (MIF) [4]. The sensitivity can be computed using the DBN [2, 3], as:

$$\text{MIF}_i = \frac{\partial R_s}{\partial R_i} = \text{P}(S = Up|e_i = Up) - \text{P}(S = Up|e_i = Dn) \tag{3}$$

where R_s is the system reliability, R_i is the component reliability.

The Diagnostic Importance Factor (DIF) [8] can also be used. This factor represents the probability that the functioning of component i contributes to the functioning of the system provided that the system is not faulty. DIF can be computed using the DBN [2, 3], as:

$$\text{DIF}_i = \text{P}(e_i = Up|S = Up) \tag{4}$$

Thus, if component i has a DIF equal to 1, it means that component i becomes critical for the functioning of the system, and if it fails, the system will fail. DIF_i criteria involves that components with more importance in the system structure are relieved to mitigate the system reliability decrease.

2.4 Component Reliability

Several mathematical models have been proposed to define the failure rate of a component [9]. In [6], the failure rate is modeled as:

$$\lambda_i = \lambda_i^0 \times g(\ell, \vartheta) \tag{5}$$

where λ_i^0 represents the baseline failure rate (nominal failure rate) for the *i*th component and $g(\ell, \vartheta)$ is a load function (independent of time) also known as covariate,

that represents the effect of stress on the component failure rate. ℓ represents an image of the load applied and ϑ is a component parameter.

The baseline failure rate can be modeled using a Weibull function [2] as:

$$\lambda_i^0 = \frac{\beta_i (k - \gamma_i)^{\beta_1 - 1}}{\eta_i^{\beta_i}} \tag{6}$$

where β is a shape parameter, γ is a location parameter, and η is a scale parameter. In this work, a constant baseline failure rate λ_i^0 is assumed.

Different definitions of function $g(\ell, \vartheta)$ exists in the literature. However, the exponential form is the most commonly used. In [13, 14] authors propose a load function based on the root-mean-square of the applied control input (u_i) until the end of the mission (t_M), and an actuator parameter defined from the upper and lower saturation bounds of u_i. This load function is used to distribute the control efforts between the redundant actuators, and the control action is calculated using a reliable state feedback.

In this work, it is assumed that the failure rate is provided by the following equation (7):

$$\lambda_i = \lambda_i^0 \, g_i(u_i) \tag{7}$$

where $g_i(u_i)$ corresponds to the following normalized control action:

$$g_i(u_i) = \frac{u_i(k) - \underline{u}_i}{\overline{u}_i - \underline{u}_i} \tag{8}$$

where $u_i(k)$ is the control effort at time k, $\underline{u}_i$ and $\overline{u}_i$ are the minimum and maximum control efforts allowed for the ith actuator and represents the amount of load on the actuator. The major actuator load corresponds to $u_i(k) = \overline{u}_i$, which leads to the worst failure rate $\lambda_i = \lambda_i^0$, through (7).

This behavior can be modeled using a DBN [26] taking advantage of the fact that the knowledge of the distribution probabilities and the CPT allows the computation of the distribution probabilities at time $k+1$. This computation is conditionally independent of the past given the present k, and can be exploited in an iterative process by using the information of time $k + 1$ to compute the distribution probabilities of time $k + 2$ and so on, as shown in Fig. 3.

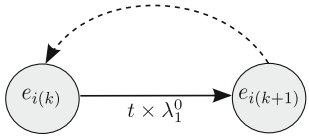


Fig. 3 DBN for component reliability

Table 3 CPT for node $e_{i,k+1}$

$e_{i,k}$	$e_{i,k+1}$	
	Up	Dn
Up	1-($\lambda_i^0 T_s\ g_i(u_i)$)	$\lambda_i^0 T_s\ g_i(u_i)$
Dn	0	1

Remark that with the inclusion of the amount of load in the failure rate, the DBN models becomes a 1/2 MC [1]. Table 3 shows the discretized CPT for the DBN model with sampling time T_s.

Therefore, the component reliability can be expressed as:

$$R_i(k+1) = \mathrm{P}(e_i(k+1) = \mathrm{Up}) \tag{9}$$

3 Reliability-Aware MPC

In this section a reliability-aware MPC design approach is proposed.

3.1 MPC Formulation

Consider the following linear discrete-time model described in state-space form of an over-actuated system with p actuators:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + E\varepsilon(k) \\ y(k) &= Cx(k) \end{aligned} \tag{10}$$

where $x(k) \in \mathbb{R}^n$ is the state vector, $u(k) \in \mathbb{R}^p$ is the control input vector with $u(k) \geq 0\ \forall\ k$, $y(k) \in \mathbb{R}^q$ is the measured output vector, $\varepsilon(k) \in \mathbb{R}^m$ is the disturbance vector, $A \in \mathbb{R}^{n\times n}$ is the state matrix, $B \in \mathbb{R}^{q\times p}$ input matrix, $E \in \mathbb{R}^{n\times m}$ is the disturbance matrix, and $C \in \mathbb{R}^{q\times n}$ is the output matrix.

MPC technique is considered to distribute the control effort among the actuators. In this work, the multiobjective optimization problem [21] is formulated as follows:

$$\begin{aligned} \min_{\substack{(\Delta\hat{u}(k|k),\ldots,\\ \Delta\hat{u}(k+Hc-1|k))}} \quad & J(k) = \sum_{j=0}^{H_p-1}\sum_{i=1}^{q} \alpha_i(k)[\hat{y}_i(k+j|k) - y_i^{ref}(k+j|k)]^2 \\ & + \sum_{j=0}^{H_c-1}\sum_{i=1}^{p} \Delta\hat{u}_i(k+j|k)^2 + \sum_{j=0}^{H_c-1}\sum_{i=1}^{p} \rho_i(k)\hat{u}_i(k+j|k)^2 \\ \text{subject to} \quad & \underline{u} \leq \hat{u}(k+j|k) \leq \overline{u} \qquad j = 0,\ldots,H_c-1 \\ & \underline{x} \leq \hat{x}(k+i|k) \leq \overline{x} \qquad i = 1,\ldots,H_p \end{aligned} \tag{11}$$

where $\hat{y}_i$ and y_i^{ref} are the predicted output and the set-point for a H_p horizon respectively, $\alpha_i(k)$ and $\rho_i(k)$ are weighting parameters, $\underline{u}$ and $\overline{u}$ denote the minimum and maximum actuator capacities, and $\underline{x}$ and $\overline{x}$ denote the minimum and maximum state values. The notation $k+i|k$ allows a future time instant $k+i$ to be referred at current time instant k, and the optimization problem consists in minimizing $\Delta\hat{u}_i(k)$ defined as $\hat{u}_i(k) - \hat{u}_i(k-1)$ over a control horizon H_c.

This multiobjective cost function considers three control objectives. The first term of the objective function aims to minimize the tracking error, which is weighted by $\alpha_i(k)$. The second aims a smooth pump operation and the third term penalizes pump operation according to $\rho_i(k)$. For instance, in [21], $\rho_i(k)$ represents the economic cost of pumping, which depends on the variable electric tariffs along a day.

3.2 Enhancing MPC with Reliability Information

In this work a reliability-aware MPC design approach which aim is to preserve as much the system reliability as possible is proposed. Such an objective is achieved by incorporating in the control design the importance factors seen in Sect. 2.3 to effectively manage the control inputs in order to reduce the loss of system reliability.

The control inputs managing is performed by the correct choice of the weights in the MPC cost function (11). The weights $\rho_i(k)$ are used then to build the weighting matrix D_u, that distributes the control effort among the actuators, this is:

$$Du_{i,k} = f(\boldsymbol{\Gamma}_{i:p,k}) := \text{diag}(\rho_1, \rho_2, \ldots, \rho_i) \tag{12}$$

where $\boldsymbol{\Gamma}$ is the criterion by which the control action is distributed among the available actuators. Different definition for $\boldsymbol{\Gamma}$ criteria are proposed.

The local approach focuses on component reliability $R_i(k)$ provided by the DBN using inference at each time instant. This definition tries to preserve system reliability through preserving components reliability, setting the weights as follows:

$$\rho_i(k) = 1 - R_i(k) \tag{13}$$

With this criteria the aim is to find the optimal control actions and distribute it among the available actuators in a way that actuators with lower reliability level are relieved. Hence, the use of highly reliable components is prioritized.

The local approach assumes an equivalent contribution of component reliability to system reliability. However, this is hardly ever true. In fact, the DBN reliability model can intrinsically explain this relation.

A global approach using MIF criteria is proposed. It is expected that components with a greater contribution to the system reliability are used less with respect to the others. Its aim is to preserve system reliability by setting the weights as follows:

$$\rho_i(k) = \text{MIF}_i(k) \tag{14}$$

Hence, the use of those components with a smaller sensitivity is prioritized. Components with bigger sensitivity are expected to greatly penalize system reliability, so they are assigned a higher cost.

Another possibility consists in combining MIF and DIF, to take advantage of both as follows:

$$\rho_i(k) = \text{MIF}_i(k) \times \text{DIF}_i(k) \tag{15}$$

This takes into account the sensitivity and criticality of the components, i.e. components with less impact in the system reliability and less critical to the system functioning will be charged more.

4 Case Study: Drinking Water Network

The proposed MPC framework is applied to a part of a Drinking Water Network (DWN) system (Fig. 4) [27].

4.1 DWN Description

A DWN is a network which is composed by sources (water supplies), sinks (water demand sectors) and pipelines that link sources to sinks. It also contains active elements like pumps and valves.

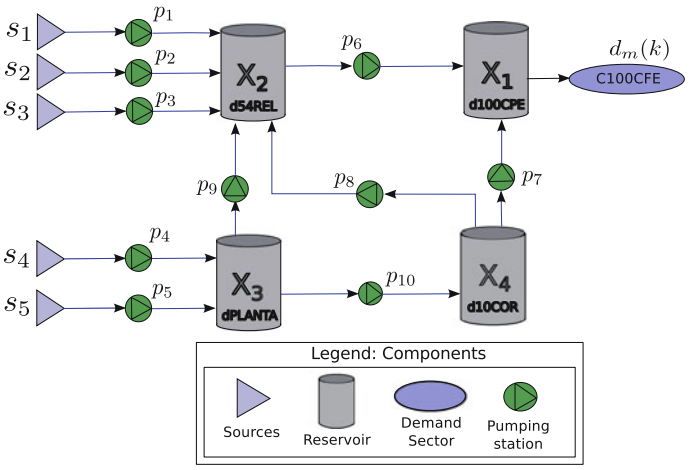


Fig. 4 Drinking water network diagram

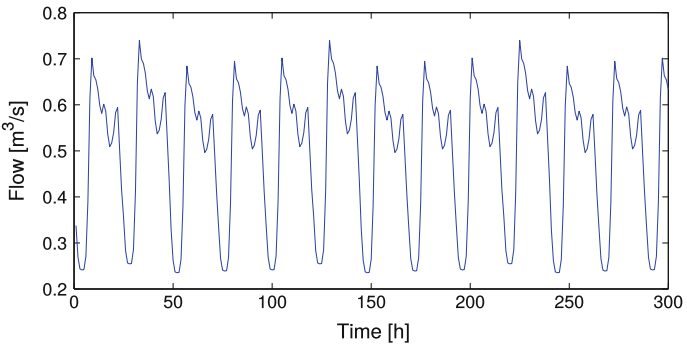


Fig. 5 Water demand

Concerning the DWN reliability study, sources, sinks, tanks and pipelines are considered perfectly reliable whereas active elements are not.

The water demand forecast can be computed using statistical data, in this case it is assumed known and presents a pattern that repeats every 4 days as Fig. 5 shows.

Remark that, it is assumed that with a single source it is possible to supply the required water demand.

4.2 System Reliability Using a Dynamic Bayesian Network Model

To compute the system and components reliability a DBN is used. The first step before modelling the DBN is to identify all the system components. The system has 10 pumps, 5 sources, 4 tanks and several pipes, Sources, tanks and pipes are considered perfectly reliable and are not subject to loss of reliability. Then, it is necessary to determine the quantity and composition of each minimal success path. In the case of the DWN system, it has 9 success paths. The availability of each success path depends on the reliability state of its components. So, the second step is to make a list of the components that are involved in each success path (Table 4).

The resulting DBN of the DWN is shown in Fig. 6 where nodes e_i and A_i are drawn for each component (i.e. pump p_i and source s_i, respectively). These nodes are interconnected to their corresponding minimal success path nodes P_i using arcs.

Each minimal success path is linked to the corresponding source availability node A'_i. This layer provides the availability of each source with respect to the availability of their corresponding minimal success paths considering that sources are assumed perfectly reliable, i.e. $\mathrm{P}(A_i = Up) = 1$. Finally, the source availability nodes are interconnected to the system reliability node S [27].

Initially, at instant $k = 0$, the pumps and the system are assumed to be fully reliable i.e. their reliability is 1. Then, the probability of each node is computed using

Table 4 Success paths components

	s_1	s_2	s_3	s_4	s_5	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
P_1	×					×					×				
P_2		×					×				×				
P_3			×					×			×				
P_4				×					×		×			×	
P_5				×					×			×			×
P_6				×					×		×		×		×
P_7					×					×	×			×	
P_8					×					×		×			×
P_9					×					×	×		×		×

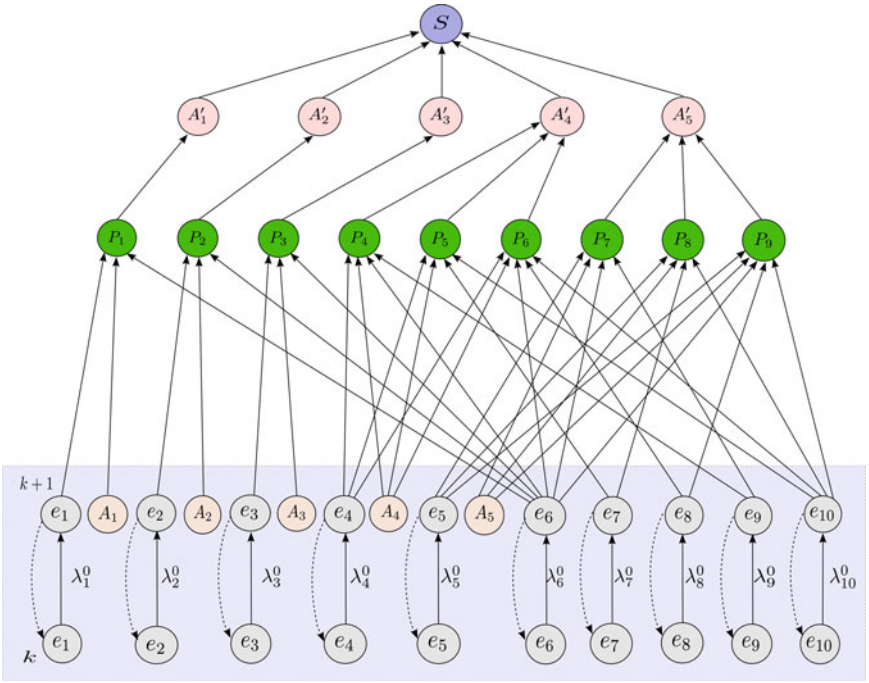


Fig. 6 Bayesian network model of the DWN

its CPT. In the case of minimal success paths nodes (P_i) their CPT follows an AND gate behavior, the availability nodes (A'_i) CPT behavior corresponds to an OR gate as well as the behavior of the system node S. At each sample time, the reliability R_i of each pump is computed according to its failure rate using the λ_i values of Table 5 in a MC (shaded blue layers in Fig. 6) using the BNT toolbox for Matlab [19] and simulated as discussed in Sect. 2.4.

Table 5 Pumps failure rate values

Failure rate λ^0 [$h^{-1} \times 10^{-4}$]									
9.85	10.70	10.50	1.40	0.85	0.80	11.70	0.60	0.74	0.78

The failure rate for each pump is assumed to be known and it could be obtained from statistical data concerning failures that have already occurred.

As the pumps failure rates depend on the control action through (5), then each time that the control action is reconfigured according to $\boldsymbol{\Gamma}$ their corresponding importance factor is updated.

5 Results

A hierarchical control structure is assumed, where the MPC formulation proposed in Sect. 3 produces at every sample time a set of set-points for the lower level flow controllers. Figure 5 displays the daily profile of the forecast water demand that has been taken into account. The initial tank volumes have been set to x_0. Table 6 provides the simulation parameters.

Figures 7 and 8 present the weights evolution under the studied criteria. As expected, under $1 - R_i$ criteria the pumps with a higher failure rate, i.e. 1, 2, 3 and 7 (see Table 5), are greatly penalized in order to decrease their reliability loss (see Fig. 7).

Figure 8 shows the MPC weights evolution under MIF_i criteria. It is clear that the weight of pump 6 is bigger than the weights corresponding to the others. This is due to the importance of pump 6 in the operation of the system. In the case of $\text{MIF}_i \times \text{DIF}_i$ criteria, the weights evolution presents a similar behavior as in the case of MIF_i.

Table 6 Simulation parameters

Parameter	Value									
H_p / H_c	24 / 8									
T_s / T_M [h]	1 / 2000									
ρ_i	$\{1, 1 - R_i, \text{MIF}_i, \text{MIF}_i \times \text{DIF}_i\}$									
α_i	0									
$\overline{u}$ [m^3/s]	0.75	0.75	0.75	1.20	0.85	1.60	1.70	0.85	1.70	1.60
$\underline{u}$ [m^3/s]	0	0	0	0	0	0	0	0	0	0
$\overline{x}$ [m^3]	65200		3100		14450			11745		
$\underline{x}$ [m^3]	25000		2200		5200			3500		
x_0 [m^3]	45100		2650		9825			7622		

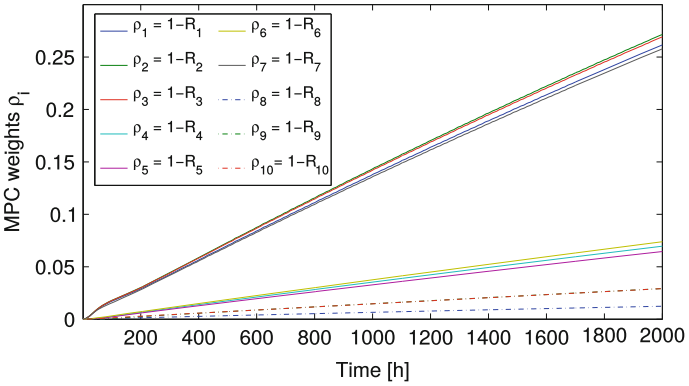


Fig. 7 MPC weights for the case $\rho_i = 1 - R_i$

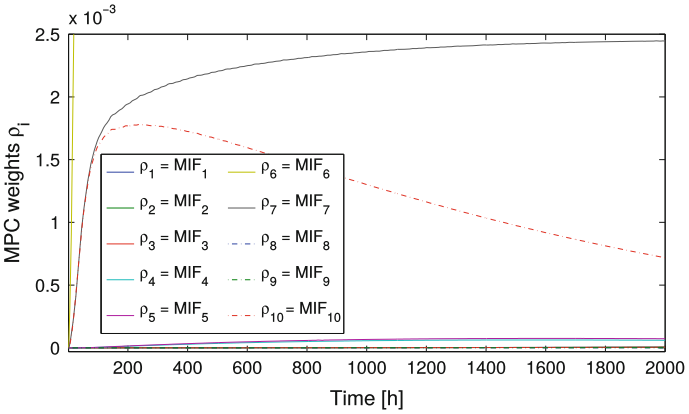


Fig. 8 MPC weights for the case $\rho_i = \text{MIF}_i$

Figure 9 shows the evolution of the pumps control effort corresponding to the scenarios where ρ_i criteria is 1, $1 - R_i$, MIF_i and $\text{MIF}_I \times \text{DIF}_i$. Notice that the plots corresponding to MIF_i and $\text{MIF}_i \times \text{DIF}_i$ are overlapped.

In order to compare the results, the cumulative pump usage U_{cum} index is defined. It is the measure of the pump energy consumption, expressed as:

$$U_{\text{cum}} = T_s \sum_{k=0}^{T_M/T_s} \left[u(k)^T u(k) \right] \tag{16}$$

The indices results are presented in Table 7.

Figure 10 shows system reliability (R_s) evolution for scenarios $\rho_i = 1$, $\rho_i = 1 - R_i$, $\rho_i = \text{MIF}_i$, and $\rho_i = \text{MIF} \times \text{DIF}$.

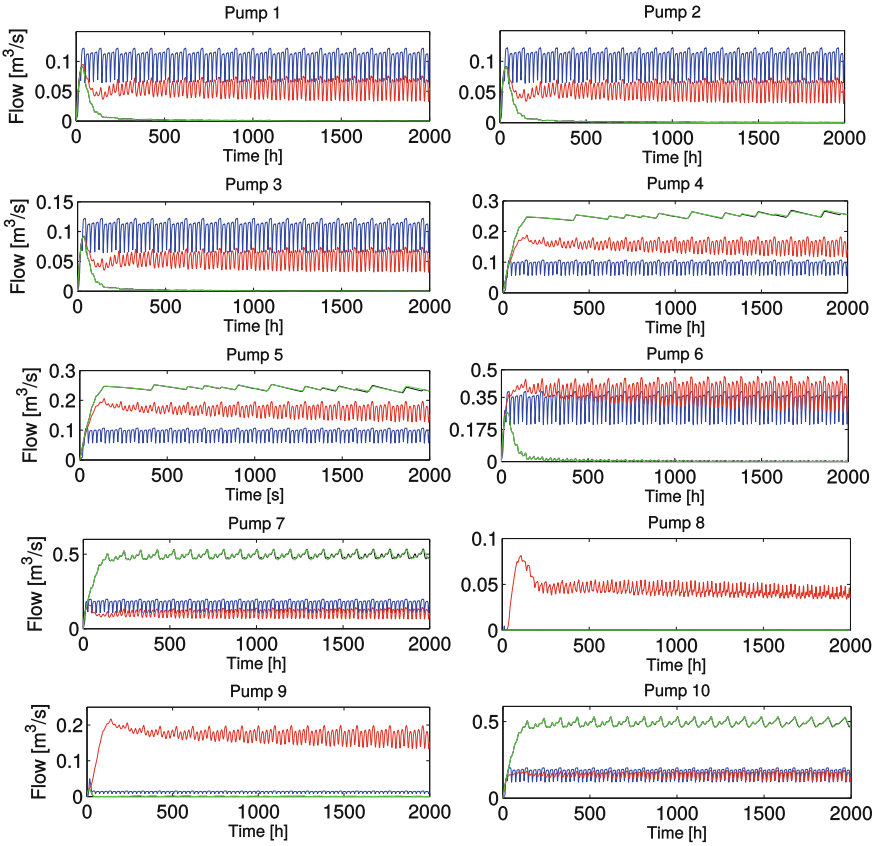


Fig. 9 Pump commands corresponding to $\rho_i = 1$ (*blue*), $\rho_i = 1 - R_i$ (*red*), and $\rho_i = \text{MIF}$ (*black*), $\rho_i = \text{MIF} \times \text{DIF}_i$ (*green*)

Table 7 Results summary

ρ_i	R_s at T_M	U_{cum}
1	0.97530	1.55685×10^6
$1 - R_i$	0.97903	2.02009×10^6
MIF_i	0.99794	4.20501×10^6
$\text{MIF}_i \times \text{DIF}_i$	0.99811	4.22680×10^6

For $\rho_i = \text{MIF}_i$ the system reliability is improved compared to the other criteria, and a small improvement is obtained using $\text{MIF}_i \times \text{DIF}_i$. Remark also that an improvement of system reliability in the scenarios MIF_i and $\text{MIF}_i \times \text{DIF}_i$ involves an increase of the energy consumption in the system. This also occurs in the $1 - R_i$ scenario, where the reliability improvement is not significant. This certifies that, improving system reliability can lead to increase of energy consumption and the fact that focusing on the reliability of components is not the best strategy to preserve system reliability.

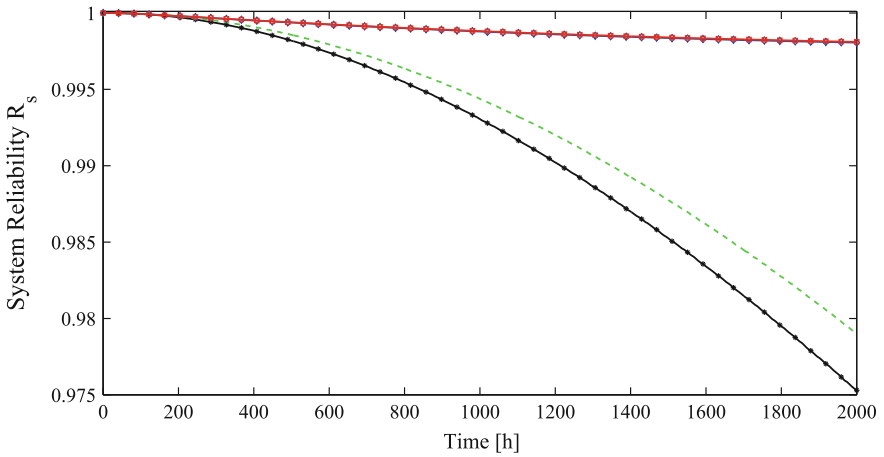


Fig. 10 System reliability: $\rho_i = 1$ (*asterisk black line*), $\rho_i = 1 - R_i$ (*dashed green line*), $\rho_i = \text{MIF}_i$ (*diamond blue line*), $\rho_i = \text{MIF} \times \text{DIF}$ (*circle red line*)

6 Conclusions

An MPC scheme was proposed using the reliability information of the system obtained in real-time from a DBN and tested in a DWN case study. A DBN model of the DWN is used to define the weights of the multiobjective cost function.

Three weights assignments have been proposed. In the first approach, component reliability is targeted, whereas in the second and third approach system reliability is focused using MIF and DIF criteria. It has been shown that focusing on the reliability of components does not guarantee the best system reliability, which was the ultimate goal.

In order to preserve system reliability as much as possible, its sensitivity to component reliability must be preferably accounted for, through MIF and DIF criteria. Analytical computation of this sensitivity is not trivial but a DBN model can provide it easily. The analytical computation needs a model of the system and the DBN offers the possibility to compute it through inference.

In this work the reliability is modelled using an MC. In the case of non-observable degradation it can be modelled using a Hidden Markov Model (HMM) or in the case of exogenous events (i.e. humidity, temperature), those variables can be modelled using Markov Switching Models (MSM) or using Input-Output HMM (IOHMM) [1].

In future research, it could be interesting to consider system availability instead of system reliability. Another issue would involve evaluating system unreliability through other importance factors, such as risk achievement worth and risk reduction worth [16].

References

1. Ben, A., Muller, A., Weber, P.: Dynamic Bayesian networks in system reliability analysis. In: Proceedings of the 6th IFAC Symposium on Fault Detection. Supervision and Safety of Technical Processes, Beijing, China, pp. 481–486 (2006)
2. Bicking, F., Weber, P., Theilliol, D.: Reliability importance measures for fault tolerant control allocation. In: Proceedings of the 2nd International Conference on Control and Fault-tolerant Systems, Nice, France, pp. 104–109 (2013)
3. Bicking, F., Weber, P., Theilliol, D., Aubrun, C.: Control allocation using reliability measures for over-actuated system. In: Intelligent Systems in Technical and Medical Diagnostics, vol. 230, pp. 487–497. Springer (2014)
4. Birnbaum, Z.W.: On the importance of different components in a multicomponent system. In: Krishnaiah, P.R. (ed.) Multivariate Analysis, vol. 2, pp. 581–592. Academic Press, New York (1969)
5. Bobio, A., Portinale, L., Minichino, M., Ciancarmela, E.: Improving the analysis of dependable systems by mapping fault trees into Bayesian networks. Reliab. Eng. Syst. Saf. **17**(3), 249–260 (2001)
6. Cox, D.R.: Regression models and life-tables. J. R. Stat. Soc. Ser. B (Methodological) **34**(2), 187–220 (1972)
7. Finkelstein, M.S.: A note on some aging properties of the accelerated life model. Reliab. Eng. Syst. Saf. **71**(1), 109–112 (2001)
8. Fussell, J.: How to hand-calculate system reliability and safety characteristics. IEEE Trans. Reliab. **R-24**(3), 169–174 (1975)
9. Gertsbakh, I.B.: Reliability Theory: With Applications to Preventive Maintenance. Springer, Berlin (2000)
10. Grosso, J.M., Ocampo-Martínez, C., Puig, V.: A service reliability model predictive control with dynamic safety stocks and actuators health monitoring for drinking water networks. In: Proceedings of the 51st IEEE Conference on Decision and Control, Hawaii, USA, pp. 4568–4573 (2012)
11. Jensen, F.: An Introduction to Bayesian Networks. Editions UCL Press, London (1996)
12. Jiang, R., Jardine, A.: Health state evaluation of an item: a general framework and graphical representation. Reliab. Eng. Syst. Saf. **93**(1), 89–99 (2008)
13. Khelassi, A., Theilliol, D., Weber, P.: Reconfigurability analysis for reliable fault-tolerant control design. Int. J. Appl. Math. Comput. Sci. **21**(3) (2011)
14. Khelassi, A., Theilliol, D., Weber, P., Ponsart, J.C.: Fault-tolerant control design with respect to actuator health degradation: an LMI approach. In: Proceedings of the IEEE International Conference on Control Applications, Denver, USA, pp. 983–988 (2011)
15. Lawless, J., Crowder, M.: Covariates and random effects in a gamma process model with application to degradation and failure. Lifetime Data Anal. **10**(3), 213–227 (2004)
16. Levitin, G., Podofillini, L., Zio, E.: Generalised importance measures for multi-state elements based on performance level restrictions. Reliab. Eng. Syst. Saf. **82**(3), 287–298 (2003)
17. Lu, C.J., Meeker, W.Q.: Using degradation measures to estimate a time-to-failure distribution. Technometrics **35**(2), 161–174 (1993)
18. Maciejowski, J.M.: Predictive Control: With Constraints. Prentice Hall, Harlow (2002)
19. Murphy, K.P.: The Bayes net toolbox for Matlab. Comput. Sci. Stat. **33**, 2001 (2001)
20. van Noortwijk, J.M.: A survey of the application of gamma processes in maintenance. Reliab. Eng. Syst. Saf. **94**(1), 2–21 (2009)
21. Ocampo-Martínez, C., Puig, V., Cembrano, G., Quevedo, J.: Application of predictive control strategies to the management of complex networks in the urban water cycle. IEEE Control Syst. Mag. **33**(1), 15–41 (2013)
22. Osaki, S., Nakagawa, T.: Bibliography for reliability and availability of stochastic systems. IEEE Trans. Reliab. **R-25**(4), 284–287 (1976)

23. Pereira, E., Galvao, R., Yoneyama, T.: Model predictive control using prognosis and health monitoring of actuators. In: Proceedings of the IEEE International Symposium on Industrial Electronics, Bari, Italy, pp. 237–243 (2010)
24. Salazar, J.C., Nejjari, F., Sarrate, R.: Reliable Control of a Twin Rotor MIMO System using Actuator Health Monitoring. In: Proceedings of the 22nd Mediterranean Conference on Control and Automation. pp. 481–486. Palermo, Italy (2014)
25. Torres-Toledano, J., Sucar, L.: Bayesian networks for reliability analysis of complex systems. In: Coelho, H. (ed.) Progress in Artificial Intelligence, pp. 195–206. Springer, Berlin (1998)
26. Weber, P., Jouffe, L.: Reliability modelling with dynamic Bayesian networks. IN: Proceedings of the 5th IFAC Symposium on Fault Detection. Supervision and Safety for Technical Processes, Washington D.C, USA, pp. 57–62 (2003)
27. Weber, P., Simon, C., Theilliol, D., Puig, V.: Fault-tolerant control design for over-actuated system conditioned by reliability: a drinking water network application. In: Proceedings of the 8th IFAC Symposium on Fault Detection. Supervision and Safety for Technical Processes, Mexico City, Mexico, pp. 558–563 (2012)
28. Welch, R., Thelen, T.: Dynamic reliability analysis in an operation context: the Bayesian network perspective. In: Smidts, C., Devooght, J., Labeau, P. (eds.) Dynamic Reliability: Future Directions, pp. 195–206. Maryland, USA (2000)
29. Wu, N.E.: Reliability of fault tolerant control systems: part I. In: Proceedings of the 40th IEEE Conference on Decision and Control, Orlando, USA, vol. 2, pp. 1460–1465 (2001)
30. Wu, N.E., Wang, X., Sampath, M., Kott, G.: An operational approach to budget-constrained reliability allocation. In: Proceedings of the 15th IFAC World Congress, Barcelona, Spain, pp. 113–118 (2002)

Towards Robust Predictive Control for Non-linear Discrete Time System

Mariusz Buciakowski, Marcin Witczak and Józef Korbicz

Abstract The paper is devoted to the issue of a robust predictive control for a class of non-linear discrete-time systems with an application of an ellipsoidal inner-bounding of a robust invariant set. The crucial issue is to maintain the state of the system inside the robust invariant feasible set, which is a set of states guaranteeing the stability of the proposed control strategy. The approach presented in this paper starts with a robust control design. In case the robust control does not provide expected results, which means that the current state does not belong to the robust invariant set, then a suitable predictive control action is performed in order to enhance the ellipsoidal invariant set. This appealing phenomenon makes it possible to enlarge the domain of attraction of the system that makes the proposed approach an efficient solution to the model predictive control problem.

Keywords Model predictive control · Robust control · Non-linear system · Robustness · Constraints

1 Introduction

Predictive control is the only advanced control technique—i.e., more advanced than standard PID control—to have had a significant and widespread impact on industrial process control and engineering [3, 7, 8, 11, 14]. The main reasons for its success in these applications are:

M. Buciakowski (✉) · M. Witczak · J. Korbicz
Institute of Control and Computation Engineering,
University of Zielona Góra, Ul. Pogórna 50, 65-246 Zielona Góra, Poland
e-mail: m.buciakowski@issi.uz.zgora.pl

M. Witczak
e-mail: m.witczak@issi.uz.zgora.pl

J. Korbicz
e-mail: j.korbicz@issi.uz.zgora.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_13

- it is the only generic control technique which can deal ordinarily with safety constraints [5, 14, 15]
- its basic formulation extends to multivariable plants with almost no modification
- it can take into account the actuator limitations
- it allows operating closer to constraints (compared to the traditional control), which frequently leads to more profitable operation, even with remarkably short periods of pay-backs
- control update rates are relatively low in these applications, so the on-line computations can be safely computed in allocated time
- its underlying idea is easy to understand
- it is more powerful than the PID control, even for single loops without constraints
- it is also relatively easy to tune, even on 'difficult' loops containing long time delays.

MPC was developed and used in the industry for nearly 20 years before attracting much serious attention from the academic control community. This community often neglected its potential for dealing with constraints, hence missing its main advantage. It was also often pointed out that, when constraints are ignored, predictive control is equivalent to the traditional one, though generally an advanced linear control. This is true but at the same time it misses the important point that issues, such as tunability and understandability, are crucial for the wider acceptance of a control technology in the industrial control environment. Fortunately, the academic community has for some years now accepted that the predictive control offers something new and has provided much analysis and new ideas, that has gone beyond current industrial applications and is preparing the ground for a much wider application of MPC. A constant increase in the computational power and speed also offers wider applications of the MPC for fast processes. On the other hand, the applicability to a wider class of systems can be achieved by developing suitable algorithms. For this purpose, the efficient predictive control scheme was introduced in [6]. The main idea behind it is to remove the necessity of using quadratic programming and replace it by simpler computation that is based on the so-called invariant set. Following the seminal work of Kouvaritakis, several works providing further efficiency extensions were proposed (see, e.g., [4]). However, most of them were proposed for linear systems only or the non-linear systems represented within the linear parameter-varying framework.

Thus, the main objective of this work is to provide an extension of the seminal approach proposed by Kouvaritakis for a class of non-linear systems. In particular, the paper is organized as follows. Section 2 presents preliminary information about the class of non-linear systems along with assumptions imposed upon the proposed design strategy. The practical way of realizing these assumptions is provided as well. In Sect. 3, a method for designing an unconstrained robust controller is provided. Subsequently, Sect. 4 extends the work of Kouvaritakis towards the class of non-linear system. Finally, the last section concludes the paper.

2 Preliminaries

Let us consider a non-linear system:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{W}\boldsymbol{w}_k \tag{1}$$

where $\boldsymbol{x}_k \in \mathbb{X} \subset \mathbb{R}^n$, $\boldsymbol{u}_k \in \mathbb{U} \subset \mathbb{R}^r$ denote the state and input, $\boldsymbol{w}_k \in l_2$ is an exogenous disturbance vector and $\boldsymbol{W} \in \mathbb{R}^{n\times n}$, stand for its distribution matrices while

$$l_2 = \left\{ \mathbf{w} \in \mathbb{R}^n |\ \|\mathbf{w}\|_{l_2} < +\infty \right\}, \|\mathbf{w}\|_{l_2} = \left(\sum_{k=0}^{\infty} \|\boldsymbol{w}_k\|^2 \right)^{\frac{1}{2}} \tag{2}$$

Moreover, the following set of assumptions are imposed:

Assumption 1.

$$\boldsymbol{g}(0) = \mathbf{0} \tag{3}$$

Assumption 2.

$$\boldsymbol{g}(\boldsymbol{x})^T \boldsymbol{x} \leq \boldsymbol{x}^T \boldsymbol{M}\boldsymbol{x}, \quad \boldsymbol{M} \in \mathbb{M} \tag{4}$$

Assumption 3.

$$\boldsymbol{g}(\boldsymbol{x})^T \boldsymbol{g}\left(\boldsymbol{x}\right) \leq \boldsymbol{x}^T \boldsymbol{M}^T \boldsymbol{M}\boldsymbol{x}, \quad \boldsymbol{M} \in \mathbb{M} \tag{5}$$

Assumption 4. The control limits shaping $\mathbb{U}$ are given by

$$-\bar{\boldsymbol{u}}_i \leq \boldsymbol{u}_{i,k} \leq \bar{\boldsymbol{u}}_i, \quad i = 1, \ldots, r \tag{6}$$

where $\bar{\boldsymbol{u}}_i > 0$ $(i = 1, \ldots, r)$ are given control bounds and $\boldsymbol{u}_{i,k}$ stands for the ith component of $\boldsymbol{u}_k$.

Assumption 5. The state limits shaping $\mathbb{X}$ are given by

$$-\bar{\boldsymbol{x}}_i \leq \boldsymbol{x}_{i,k} \leq \bar{\boldsymbol{x}}_i, \quad i = 1, \ldots, n \tag{7}$$

where $\bar{\boldsymbol{x}}_i > 0$ $(i = 1, \ldots, n)$ are given state bounds and $\boldsymbol{x}_{i,k}$ stands for the ith component of $\boldsymbol{u}_k$.

Assumption 6. $\boldsymbol{w}_k$ satisfies

$$\boldsymbol{w}_k \in \mathcal{E}_{\boldsymbol{w}_k}, \quad \mathcal{E}_{\boldsymbol{Q}} = \{\boldsymbol{w} \in \mathbb{R}^{r+n} |\ \boldsymbol{w}^T \boldsymbol{Q}\boldsymbol{w} \leq 1\}, \quad \boldsymbol{Q} \succ 0 \tag{8}$$

Assumption 7. $\boldsymbol{x}_k$ satisfies

$$\boldsymbol{x}_k \in \mathcal{E}_{\boldsymbol{x}_k}, \quad \mathcal{E}_{\boldsymbol{P}} = \{\boldsymbol{x}_k \in \mathbb{R}^n | \boldsymbol{x}_k^T \boldsymbol{P} \boldsymbol{x}_k \leq 1\}, \quad \boldsymbol{P} \succ 0 \tag{9}$$

For the purpose of further deliberations, let us remind the Differential Mean Value Theorem (DMVT) [16]:

$$\boldsymbol{g}(\boldsymbol{a}) - \boldsymbol{g}(\boldsymbol{b}) = \boldsymbol{M}_x(\boldsymbol{a} - \boldsymbol{b}) \tag{10}$$

with

$$\boldsymbol{M}_x = \begin{bmatrix} \dfrac{\partial g_1}{\partial x}(\boldsymbol{c_1}) \\ \vdots \\ \dfrac{\partial g_n}{\partial x}(\boldsymbol{c_n}) \end{bmatrix} \tag{11}$$

where $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n \in \mathrm{Co}(\boldsymbol{a}, \boldsymbol{b})$, $\boldsymbol{c}_i \neq \boldsymbol{a}$, $\boldsymbol{c}_i \neq \boldsymbol{b}$, $i = 1, \ldots, n$. Assuming that

$$\bar{a}_{i,j} \geq \frac{\partial g_i(\boldsymbol{x})}{\partial x_j} \geq \underline{a}_{i,j}, \quad i = 1, \ldots, n, \quad j = 1, \ldots, n \tag{12}$$

it is clear that there exists a matrix $\boldsymbol{M} \in \mathbb{M}$ such that:

$$\mathbb{M} = \left\{ \boldsymbol{M} \in \mathbb{R}^{n \times n} | \bar{a}_{i,j} \geq m_{i,j} \geq \underline{a}_{i,j}, \, i,j = 1, \ldots, n, \right\} \tag{13}$$

Thus, under *Assumption 1*, by substituting $\boldsymbol{b} = \boldsymbol{0}$ it easily can be shown that

$$\boldsymbol{g}(\boldsymbol{a}) = \boldsymbol{M}_x(\boldsymbol{a}) \tag{14}$$

It is thus evident that there $\boldsymbol{M} \in \mathbb{M}$ such that the inequality (4) is satisfied. From this constraint, it can be noticed that

$$\boldsymbol{g}(\boldsymbol{x})^T \boldsymbol{x} \leq \boldsymbol{x}^T \boldsymbol{M} \boldsymbol{x} = \frac{1}{2} \boldsymbol{x}^T (\boldsymbol{M} + \boldsymbol{M}^T) \boldsymbol{x} \leq \boldsymbol{x}^T \Theta \boldsymbol{x}, \quad \boldsymbol{M} \in \mathbb{M} \tag{15}$$

where $\Theta = \mathrm{diag}(\theta_1, \ldots, \theta_n)$. Thus, by noticing that $\frac{1}{2}(\boldsymbol{M} + \boldsymbol{M}^T)$ is a symmetric matrix, when the inequality

$$\boldsymbol{x}^T \left(\frac{1}{2} \boldsymbol{M} + \frac{1}{2} \boldsymbol{M}^T - \Theta \right) \boldsymbol{x} \leq 0, \quad \boldsymbol{M} \in \mathbb{M} \tag{16}$$

is satisfied then all diagonal entries of $\frac{1}{2}\boldsymbol{M} + \frac{1}{2}\boldsymbol{M}^T - \Theta$ must be non-positive. This property allows convenient computation of $\Theta = \mathrm{diag}(\theta_1, \ldots, \theta_n)$. This deliberation clearly exhibits a method of satisfying *Assumption 2*.

By a similar reasoning, it can be concluded that there exists $\boldsymbol{M}$ such that $\boldsymbol{M}_{x,u}^T \boldsymbol{M}_{x,u} \preceq \boldsymbol{M}^T \boldsymbol{M}$. In order to find the upper bound $\boldsymbol{M}^T \boldsymbol{M}$, the following evident inequality is used: $\boldsymbol{M}^T \boldsymbol{M} \preceq \lambda_{\max}(\boldsymbol{M}^T \boldsymbol{M}) \boldsymbol{I}_n$. Thus, the problem is

$$\boldsymbol{M}^* = \arg \max_{\boldsymbol{M} \in \mathbb{M}} \lambda_{\max}(\boldsymbol{M}^T \boldsymbol{M}) \tag{17}$$

Taking into account the fact that $\lambda_{\max}(\boldsymbol{M}^T \boldsymbol{M}) = \|\boldsymbol{M}^T \boldsymbol{M}\|_2 = \|\boldsymbol{M}\|_2^2$, the optimisation problem (17) can be replaced by

$$\boldsymbol{M}^* = \arg \max_{\boldsymbol{M} \in \mathbb{M}} \|\boldsymbol{M}\| \tag{18}$$

which can be perceived as a worst case norm analysis task. This easily can be solved, e.g., with MATLAB (cf. the `wcnorm` function).

It is worth noting that, if $\boldsymbol{M}^T \boldsymbol{M} = \gamma^2 \boldsymbol{I}$, then *Assumption 3* becomes a usual Lipschitz condition [1, 9, 12, 13], with γ being a Lipschitz constant. This appealing property makes the employed strategy more general than those presented in the literature [1, 9, 12, 13].

Finally, *Assumption 3* can be rewritten as

$$\boldsymbol{g}(\boldsymbol{x})^T \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{x}^T (\boldsymbol{M}^*)^T \boldsymbol{M}^* \boldsymbol{x} \tag{19}$$

3 Design of the State Feedback Controller

The main objective of this section is to present the design procedure of the robust controller for proposed system. The controller will be designed in such a way that for a predefined disturbance attenuation level with respect is the state of the system is achieved. To solve the above stated problem the following control scheme was proposed

$$\boldsymbol{u}_k = -\boldsymbol{K} \boldsymbol{x}_k \tag{20}$$

Substituting (20) into (1) gives

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_1 \boldsymbol{x}_k + \boldsymbol{g}(\boldsymbol{x}_k) + \boldsymbol{W} \boldsymbol{w}_k \tag{21}$$

where

$$\boldsymbol{A}_1 = \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} \tag{22}$$

Theorem 1 *For a prescribed disturbance attenuation level μ, the controller design problem for the system (21) is solvable if there exist $\boldsymbol{N}$, $\boldsymbol{U}$, $\boldsymbol{P} \succ \boldsymbol{0}$, $\alpha > 0$, $\beta > 0$ such that the following condition is satisfied:*

$$\begin{bmatrix} -\boldsymbol{P}+\alpha\Theta+\beta(\boldsymbol{M}^*)^T\boldsymbol{M}^* & -\frac{1}{2}\alpha\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{U}^T\boldsymbol{A}_1^T & \boldsymbol{U}^T \\ -\frac{1}{2}\alpha\boldsymbol{I} & -\beta\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & -\mu^2\boldsymbol{I} & \boldsymbol{W}^T & \boldsymbol{0} \\ \boldsymbol{A}_1\boldsymbol{U} & \boldsymbol{I} & \boldsymbol{W} & \boldsymbol{P}-\boldsymbol{U}-\boldsymbol{U}^T & \boldsymbol{0} \\ \boldsymbol{U} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{I} \end{bmatrix} \prec 0 \tag{23}$$

$$\text{with } \boldsymbol{A}_1\boldsymbol{U} = (\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K})\boldsymbol{U} = \boldsymbol{A}\boldsymbol{U}-\boldsymbol{B}\boldsymbol{K}\boldsymbol{U} = \boldsymbol{A}\boldsymbol{U}-\boldsymbol{B}\boldsymbol{N} \tag{24}$$

Proof The problem of $\mathcal{H}_\infty$ controller design is to determine the matrix $\boldsymbol{K}$ such that

$$\lim_{k\to\infty}\boldsymbol{x}_k = \boldsymbol{0} \quad \text{for } \boldsymbol{w}_k = \boldsymbol{0} \tag{25}$$

$$\|\boldsymbol{x}_k\|_{l_2} \le \mu\|\boldsymbol{v}_k\|_{l_2} \quad \text{for } \boldsymbol{w}_k \neq \boldsymbol{0},\ \boldsymbol{x}_0 = \boldsymbol{0} \tag{26}$$

In order to settle the above problem it suffices to find a Lyapunov function V_k such that:

$$\Delta V_k + \boldsymbol{x}_{f,k}^T\boldsymbol{x}_{f,k} - \mu^2\boldsymbol{w}_k^T\boldsymbol{w}_k < 0,\ k = 0, \dots \infty \tag{27}$$

where

$$\Delta V_k = V_{k+1} - V_k \tag{28}$$

$$V_k = \boldsymbol{x}_k^T\boldsymbol{P}_1\boldsymbol{x}_k \tag{29}$$

$$\Delta V_k = \boldsymbol{x}_{k+1}^T\boldsymbol{P}_1\boldsymbol{x}_{k+1} - \boldsymbol{x}_k^T\boldsymbol{P}_1\boldsymbol{x}_k \tag{30}$$

$$\boldsymbol{P}_1 = \boldsymbol{U}^{-T}\boldsymbol{P}\boldsymbol{U}^{-1} \tag{31}$$

Note that by Rayleigh quotient and the remark associated with *Lemma 1*:

$$\underline{\alpha} \le \lambda_i(\boldsymbol{U}^T\boldsymbol{U}) \le \overline{\alpha}, \quad \underline{\beta} \le \lambda_i(P) \le \overline{\beta} \quad i = 1, \dots, n$$

where $\lambda(\cdot)$ stands for an eigenvalue of its argument. This implies that

$$\underline{\alpha}\overline{\beta}\boldsymbol{x}_{f,k}^T\boldsymbol{x}_{f,k} \le V_k \le \overline{\alpha}\underline{\beta}\boldsymbol{x}_{f,k}^T\boldsymbol{x}_{f,k}$$

which clearly indicates that V_k is a proper Lyapunov candidate matrix.

Consequently, using (21)

$$\begin{aligned} &\Delta V_k + \boldsymbol{x}_{f,k}^T\boldsymbol{x}_{f,k} - \mu^2\boldsymbol{w}_k^T\boldsymbol{w}_k = \\ &\boldsymbol{x}_k^T\left(\boldsymbol{A}_1^T\boldsymbol{P}_1\boldsymbol{A}_1 - \boldsymbol{P}_1 + \boldsymbol{I}\right)\boldsymbol{x}_k + \boldsymbol{x}_k^T\left(\boldsymbol{A}_1^T\boldsymbol{P}_1\right)\boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{x}_k^T\left(\boldsymbol{A}_1^T\boldsymbol{P}_1\boldsymbol{W}\right)\boldsymbol{w}_k + \\ &\boldsymbol{g}\left(\boldsymbol{x}_k\right)^T\left(\boldsymbol{P}_1\boldsymbol{A}_1\right)\boldsymbol{x}_k + \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T\left(\boldsymbol{P}_1\right)\boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T\left(\boldsymbol{P}_1\boldsymbol{W}\right)\boldsymbol{w}_k + \\ &\boldsymbol{w}_k^T\left(\boldsymbol{W}^T\boldsymbol{P}_1\boldsymbol{A}_1\right)\boldsymbol{x}_k + \boldsymbol{w}_k^T\left(\boldsymbol{W}^T\boldsymbol{P}_1\right)\boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{w}_k^T\left(\boldsymbol{W}^T\boldsymbol{P}_1\boldsymbol{W} - \mu^2\boldsymbol{I}\right)\boldsymbol{w}_k < \boldsymbol{0} \end{aligned} \tag{32}$$

By defining

$$\boldsymbol{v}_k = [\boldsymbol{x}_k^T, \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T, \boldsymbol{w}_k^T]^T \tag{33}$$

it can be shown that (32) is equivalent to

$$\boldsymbol{v}_k^T \begin{bmatrix} \boldsymbol{A}_1^T\boldsymbol{P}_1\boldsymbol{A}_1 - \boldsymbol{P}_1 + \boldsymbol{I} & \boldsymbol{A}_1^T\boldsymbol{P}_1 & \boldsymbol{A}_1^T\boldsymbol{P}_1\boldsymbol{W} \\ \boldsymbol{P}_1\boldsymbol{A}_1 & \boldsymbol{P}_1 & \boldsymbol{P}_1\boldsymbol{W} \\ \boldsymbol{W}^T\boldsymbol{P}_1\boldsymbol{A}_1 & \boldsymbol{W}^T\boldsymbol{P}_1 & \boldsymbol{W}^T\boldsymbol{P}_1\boldsymbol{W} - \mu^2\boldsymbol{I} \end{bmatrix} \boldsymbol{v}_k \prec \boldsymbol{0} \tag{34}$$

Analyzing assumption of non-linear function, in particular (15), for some α inequality can be written as

$$\alpha \boldsymbol{x}_k^T \Theta \boldsymbol{x}_k - \alpha \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T \boldsymbol{x}_k \succeq 0, \quad \alpha > 0, \quad \boldsymbol{M} \in \mathbb{M} \tag{35}$$

which is equivalent to

$$\alpha \boldsymbol{v}_k^T \begin{bmatrix} \Theta & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{0} \\ -\frac{1}{2}\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{v}_k \succeq \boldsymbol{0}, \quad \alpha > 0, \quad \boldsymbol{M} \in \mathbb{M} \tag{36}$$

By congruence, i.e. by multiplying the matrix shaping the quadratic form (36), from left by diag$(\boldsymbol{U}^{-T}, \boldsymbol{I}, \boldsymbol{I})$ and from right by diag$(\boldsymbol{U}^{-1}, \boldsymbol{I}, \boldsymbol{I})$ it can be shown that (36) is equivalent to

$$\alpha \boldsymbol{v}_k^T \begin{bmatrix} \boldsymbol{U}^{-T}\Theta\boldsymbol{U}^{-1} & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{0} \\ -\frac{1}{2}\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{v}_k \succeq \boldsymbol{0} \tag{37}$$

Similarly, for some β inequality (19) can be written as

$$\beta \boldsymbol{x}_k^T (\boldsymbol{M}^*)^T \boldsymbol{M}^* \boldsymbol{x}_k - \beta \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T \boldsymbol{g}\left(\boldsymbol{x}_k\right) \succeq 0, \quad \beta > 0 \tag{38}$$

Using the same procedure as for (35), it is possible to show that (38) is equivalent to

$$\beta \boldsymbol{v}_k^T \begin{bmatrix} \boldsymbol{U}^{-T}(\boldsymbol{M}^*)^T\boldsymbol{M}^*\boldsymbol{U}^{-1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & -\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{v}_k \succeq \boldsymbol{0} \tag{39}$$

Combining (34), (37) and (39) together gives

$$\begin{bmatrix} \boldsymbol{A}_1^T\boldsymbol{P}_1\boldsymbol{A}_1 - \boldsymbol{P}_1 + \boldsymbol{I} + \alpha\boldsymbol{M}_1 + \beta\boldsymbol{M}_2 & \boldsymbol{A}_1^T\boldsymbol{P}_1 - \frac{1}{2}\alpha\boldsymbol{I} & \boldsymbol{A}_1^T\boldsymbol{P}_1\boldsymbol{W} \\ \boldsymbol{P}_1\boldsymbol{A}_1 - \frac{1}{2}\alpha\boldsymbol{I} & \boldsymbol{P}_1 - \beta\boldsymbol{I} & \boldsymbol{P}_1\boldsymbol{W} \\ \boldsymbol{W}^T\boldsymbol{P}_1\boldsymbol{A}_1 & \boldsymbol{W}^T\boldsymbol{P}_1 & \boldsymbol{W}^T\boldsymbol{P}_1\boldsymbol{W} - \mu^2\boldsymbol{I} \end{bmatrix} \prec \boldsymbol{0} \tag{40}$$

with

$$\boldsymbol{M}_1 = \boldsymbol{U}^{-T}\Theta\boldsymbol{U}^{-1}, \quad \boldsymbol{M}_2 = \boldsymbol{U}^{-T}(\boldsymbol{M}^*)^T\boldsymbol{M}^*\boldsymbol{U}^{-1} \tag{41}$$

Multiplying (40) from left by $\mathrm{diag}(\boldsymbol{U}^T, \boldsymbol{I}, \boldsymbol{I})$ and from right by $\mathrm{diag}(\boldsymbol{U}, \boldsymbol{I}, \boldsymbol{I})$, using Theorem 1 in [10] and then applying the Schur complements leads to (23), which completes the proof. □

4 Efficient Predictive Control

The objective of this section is to present the design procedure of the efficient predictive control for nonlinear system. For this purpose, the efficient predictive control scheme introduced in [6] is utilised. The control scheme has the following form

$$\boldsymbol{u}_k = -\boldsymbol{K}\boldsymbol{x}_k + \boldsymbol{c}_k \tag{42}$$

while the predicted input sequence is

$$\boldsymbol{u}_j = \begin{cases} -\boldsymbol{K}\boldsymbol{x}_j + \boldsymbol{c}_j, j = k, \ldots, k + n_c - 1 \\ -\boldsymbol{K}\boldsymbol{x}_j, \qquad\qquad j \geq k + n_c \end{cases} \tag{43}$$

where:

- n_c is the prediction horizon
- $\boldsymbol{K}$ is the $\mathcal{H}_\infty$ controller designed in previous section
- $\boldsymbol{c}_j$ is a vector introducing additional design freedom.

Note that beyond the control horizon n_c, $\boldsymbol{c}_j$ is set to zero. Within a prediction horizon a suitable control action can be provided which makes the robust control feasible.

Thus, the design of the proposed control strategy boils down to solving a set of problems:

- to design a robust controller $\boldsymbol{K}$ in such a way that a prescribed disturbance attenuation level is achieved with respect to $\boldsymbol{x}_k$ while guaranteeing its convergence to the origin
- to determine a set of states for which the robust controller (under the control and state constraints) is feasible
- to determine $\boldsymbol{c}_j$ so as to prevent actuator saturation and/or violation of state constraints.

To solve above problem let us start from formulating the predictions at time k, which (following [6]) can be given in the following extended form

$$z_{k+1} = \boldsymbol{Z} z_k + \boldsymbol{Y} \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{V} \boldsymbol{w}_k \tag{44}$$

where

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} & \boldsymbol{B}\boldsymbol{T} \\ \boldsymbol{0} & \boldsymbol{M}_1 \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{0}_{(n_c)r\times(n)} \end{bmatrix}$$
$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{W} \\ \boldsymbol{0}_{(n_c)r\times(n)} \end{bmatrix}, \quad \boldsymbol{M}_1 = \begin{bmatrix} \boldsymbol{0}_{(n_c-1)r\times r} & \boldsymbol{I} \\ \boldsymbol{0}_{r\times r} & \boldsymbol{0}_{r\times(n_c-1)r} \end{bmatrix}$$
$$z_k = \begin{bmatrix} \boldsymbol{x}_k \\ \omega_k \end{bmatrix}, \quad \boldsymbol{\omega}_k = \begin{bmatrix} \boldsymbol{c}_k \\ \boldsymbol{c}_{k+1} \\ \cdots \\ \boldsymbol{c}_{k+n_c-1} \end{bmatrix}$$
$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{I}_{r\times r} & \boldsymbol{0} & \ldots & \boldsymbol{0} \end{bmatrix}$$

The subsequent part of this section is devoted to investigate the stability of (44). For this purpose let use introduce the following definitions [2]: The system (44) is strictly quadratically bounded with $\boldsymbol{P} \succ \boldsymbol{0}$ for all allowable $\boldsymbol{w}_k \in \mathcal{E}_{\boldsymbol{Q}}$, $k \geq 0$, if $V_k > 1$ implies $V_{k+1} - V_k < 0$ for any $\boldsymbol{w}_k \in \mathcal{E}_{\boldsymbol{Q}}$, a set $\mathcal{E}_{\boldsymbol{P}}$ is a robust invariant set for the system (44) for all allowable $\boldsymbol{w}_k \in \mathcal{E}_{\boldsymbol{Q}}$ if $z_k \in \mathcal{E}_{\boldsymbol{P}}$ implies $z_{k+1} \in \mathcal{E}_{\boldsymbol{P}}$, for any $\boldsymbol{w}_k \in \mathcal{E}_{\boldsymbol{Q}}$. Following [2], strict quadratic boundedness guarantees that $V_{k+1} - V_k < 0$, and hence, it is employed for further stability analysis. Using the above definitions, the following theorem, which constitutes the main result of this section, is formulated:

Theorem 2 *The following statements are equivalent:*

1. *The system* (44) *is strictly quadratically bounded with* $\boldsymbol{P} \succ \boldsymbol{0}$ *for all allowable* $\boldsymbol{w}_k \in \mathcal{E}_{\boldsymbol{Q}}$ *and satisfies* (6) *and* (7).
2. *The ellipsoid* $\mathcal{E}_{\boldsymbol{P}}$ *is a robust invariant set for the system* (44) *satisfying* (6) *and* (7) *for all allowable* $\boldsymbol{w}_k \in \mathcal{E}_{\boldsymbol{Q}}$.
3. *There exist* $0 < \gamma < 1$, $\alpha > 0$, $\beta > 0$, $\boldsymbol{P} \succ \boldsymbol{0}$, $\boldsymbol{F}$, $\boldsymbol{L}$ *such that*

$$\begin{bmatrix} -\boldsymbol{P} + \alpha\Theta + \beta(\boldsymbol{M}^*)^T\boldsymbol{M}^* & -\frac{1}{2}\alpha\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{Z}^T\boldsymbol{P} \\ -\frac{1}{2}\alpha\boldsymbol{I} & -\beta\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{Y}^T\boldsymbol{P} \\ \boldsymbol{0} & \boldsymbol{0} & -\gamma\boldsymbol{Q} & \boldsymbol{V}^T\boldsymbol{P} \\ \boldsymbol{P}\boldsymbol{Z} & \boldsymbol{P}\boldsymbol{Y} & \boldsymbol{P}\boldsymbol{V} & -\boldsymbol{P} \end{bmatrix} \prec 0 \tag{45}$$

$$\begin{bmatrix} -\boldsymbol{F} & \boldsymbol{T}_z \\ \boldsymbol{T}_z^T & -\boldsymbol{P} \end{bmatrix} \prec \boldsymbol{0}, \quad \boldsymbol{F}_{i,i} \leq \bar{\boldsymbol{x}}_{Si}^2, \quad i = 1, \ldots, n \tag{46}$$

$$\begin{bmatrix} -\boldsymbol{L} & \boldsymbol{R} \\ \boldsymbol{R}^T & -\boldsymbol{P} \end{bmatrix} \prec \boldsymbol{0}, \quad \boldsymbol{L}_{i,i} \leq \bar{u}_i^2, \quad i = 1, \ldots, r \tag{47}$$

Proof Definition 1 implies that

$$\boldsymbol{w}_k^T \boldsymbol{Q} \boldsymbol{w}_k < \boldsymbol{z}_k^T \boldsymbol{P} \boldsymbol{z}_k, \Rightarrow \boldsymbol{z}_{k+1}^T \boldsymbol{P} \boldsymbol{z}_{k+1} - \boldsymbol{z}_k \boldsymbol{P} \boldsymbol{z}_k < 0 \tag{48}$$

As a consequence, using (44) it can be shown that

$$\begin{aligned} &\boldsymbol{z}_{k+1}^T \boldsymbol{P} \boldsymbol{z}_{k+1} - \boldsymbol{z}_k \boldsymbol{P} \boldsymbol{z}_k = \qquad (49)\\ &\boldsymbol{z}_k^T \left(\boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Z} - \boldsymbol{P}\right) \boldsymbol{z}_k + \boldsymbol{z}_k^T \left(\boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Y}\right) \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{z}_k^T \left(\boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{V}\right) \boldsymbol{w}_k + \\ &\boldsymbol{g}\left(\boldsymbol{x}_k\right)^T \left(\boldsymbol{Y}^T \boldsymbol{P} \boldsymbol{Z}\right) \boldsymbol{z}_k + \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T \left(\boldsymbol{Y}^T \boldsymbol{P} \boldsymbol{Y}\right) \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T \left(\boldsymbol{Y}^T \boldsymbol{P} \boldsymbol{V}\right) \boldsymbol{w}_k + \\ &\boldsymbol{w}_k^T \left(\boldsymbol{V}^T \boldsymbol{P} \boldsymbol{Z}\right) \boldsymbol{z}_k + \boldsymbol{w}_k^T \left(\boldsymbol{W}^T \boldsymbol{P} \boldsymbol{Y}\right) \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{w}_k^T \left(\boldsymbol{W}^T \boldsymbol{P} \boldsymbol{V}\right) \boldsymbol{w}_k < \boldsymbol{0} \qquad (50) \end{aligned}$$

By defining

$$\bar{\boldsymbol{v}}_k = [\boldsymbol{z}_k^T, \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T, \boldsymbol{w}_k^T]^T$$

it can be shown that (50) is equivalent to

$$\bar{\boldsymbol{v}}_k^T \begin{bmatrix} \boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Z} - \boldsymbol{P} & \boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Y} & \boldsymbol{Z}^T \boldsymbol{P} \boldsymbol{Y} \\ \boldsymbol{Y}^T \boldsymbol{P} \boldsymbol{Z} & \boldsymbol{Y}^T \boldsymbol{P} \boldsymbol{Y} & \boldsymbol{Y}^T \boldsymbol{P} \boldsymbol{V} \\ \boldsymbol{V}^T \boldsymbol{P} \boldsymbol{Z} & \boldsymbol{W}^T \boldsymbol{P} \boldsymbol{Y} & \boldsymbol{W}^T \boldsymbol{P} \boldsymbol{V} \end{bmatrix} \bar{\boldsymbol{v}}_k \prec \boldsymbol{0} \tag{51}$$

Similarly for Eqs. (35) and (38) and for some $\alpha > 0$ and $\beta > 0$ it is

$$\alpha \boldsymbol{z}_k^T \boldsymbol{Y} \Theta \boldsymbol{Y}^T \boldsymbol{z}_k - \alpha \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T \boldsymbol{Y}^T \boldsymbol{z}_k \succeq 0, \quad \alpha > 0 \tag{52}$$

which is equivalent to

$$\alpha \bar{\boldsymbol{v}}_k^T \begin{bmatrix} \boldsymbol{Y} \Theta \boldsymbol{Y}^T & -\frac{1}{2}\boldsymbol{Y} & \boldsymbol{0} \\ -\frac{1}{2}\boldsymbol{Y} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \bar{\boldsymbol{v}}_k \succeq \boldsymbol{0}, \quad \alpha > 0, \quad \boldsymbol{M} \in \mathbb{M} \tag{53}$$

and

$$\beta \boldsymbol{z}_k^T \boldsymbol{Y} (\boldsymbol{M}^*)^T \boldsymbol{M}^* \boldsymbol{Y} \boldsymbol{z}_k - \beta \boldsymbol{g}\left(\boldsymbol{x}_k\right)^T \boldsymbol{g}\left(\boldsymbol{x}_k\right) \succeq 0, \quad \beta > 0, \quad \boldsymbol{M} \in \mathbb{M} \tag{54}$$

which is equivalent to

$$\beta \bar{\boldsymbol{v}}_k^T \begin{bmatrix} \boldsymbol{Y} (\boldsymbol{M}^*)^T \boldsymbol{M}^* \boldsymbol{Y}^T & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & -\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \bar{\boldsymbol{v}}_k \succeq \boldsymbol{0} \tag{55}$$

From (48), in particular from $\boldsymbol{w}_k^T \boldsymbol{Q} \boldsymbol{w}_k < \boldsymbol{z}_k^T \boldsymbol{P} \boldsymbol{z}_k$ it is evident that for $\gamma > 0$

$$\gamma \bar{\boldsymbol{v}}_k^T \begin{bmatrix} -\boldsymbol{P} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{Q} \end{bmatrix} \bar{\boldsymbol{v}}_k \prec \boldsymbol{0} \tag{56}$$

Combining (51), (53), (55) and (56) together and then applying Schur complement leads to (45).

To incorporate the input constraints, let us define

$$\boldsymbol{R} = [-\boldsymbol{K} \quad \boldsymbol{T}] \tag{57}$$

and hence

$$\boldsymbol{u}_k = \boldsymbol{R}\boldsymbol{z}_k \tag{58}$$

Subsequently, it can be observed that for $\boldsymbol{z}_k \in \mathcal{E}_{\boldsymbol{P}}$ the above inequality implies

$$\begin{aligned} &|\boldsymbol{e}_i^T \boldsymbol{u}_k|^2 = |\boldsymbol{e}_i^T \boldsymbol{R}\boldsymbol{z}_k|^2 = \\ &|\boldsymbol{e}_i^T \boldsymbol{R}\boldsymbol{P}^{-1/2}\boldsymbol{P}^{1/2}\boldsymbol{z}_k|^2 \leq ||\boldsymbol{e}_i^T \boldsymbol{R}\boldsymbol{P}^{-1/2}||^2 ||\boldsymbol{P}^{1/2}\boldsymbol{z}_k||^2 \leq ||\boldsymbol{e}_i^T \boldsymbol{R}\boldsymbol{P}^{-1/2}||^2 \end{aligned} \tag{59}$$

and if there exists a symmetric matrix $\boldsymbol{F}$ such that

$$\boldsymbol{R}\boldsymbol{P}^{-1}\boldsymbol{R}^T - \boldsymbol{F} \prec \boldsymbol{0}, \quad \boldsymbol{F}_{i,i} \leq \bar{u}_i^2, \quad i = 1, \ldots, r \tag{60}$$

then $|u_{i,k}| \leq \bar{u}_i$, $(i = 1, \ldots, r)$. Note that using Schur complements leads to (47).

The state constraints can be written in a similar fashion. Let $\boldsymbol{e}_i$ denote ith column of the n-order identity matrix, which makes it possible to rewrite the state constraints as follows:

$$|\boldsymbol{e}_i^T \boldsymbol{x}_k| \leq \bar{\boldsymbol{x}}_i, \quad i = 1, \ldots, n \tag{61}$$

Let

$$\boldsymbol{x}_k = \boldsymbol{T}_z \boldsymbol{z}_k \tag{62}$$

Using the similar approach as the one for the input constraints, the state limits can be rewritten as

$$|\boldsymbol{e}_i^T \boldsymbol{x}_k|^2 \leq \|\boldsymbol{e}_i^T \boldsymbol{T}_z \boldsymbol{P}^{-1/2}\|^2 \tag{63}$$

which can be written in the LMI form (46), which completes the proof. □

If the robust invariant set along with input ans state constraints are described in a form of LMIs, then it is possible to solve them and simultaneously maximize the invariant set. For this purpose, the determinant criterion is selected to maximize the size of the $\mathcal{E}_{\boldsymbol{P}}$:

$$\max \det(\boldsymbol{P}) \tag{64}$$

under the constraints formed with (45), (46), (47).

The algorithm for computing $\boldsymbol{c}_k$ in (44) is also inspired by [6] and boils down to perform, at each sampling time, the following minimization

$$\boldsymbol{\omega}_k^* = \min_{\boldsymbol{\omega}_k} \boldsymbol{\omega}_k^T \boldsymbol{\omega}_k, \qquad s.t.\ \boldsymbol{z}_k^T \boldsymbol{P} \boldsymbol{z}_k \leq 1 \tag{65}$$

which can be equivalently written as:

$$\begin{aligned} \boldsymbol{\omega}_k^* = \min_{\boldsymbol{\omega}_k} \boldsymbol{\omega}_k^T \boldsymbol{\omega}_k, \qquad & s.t.\ \boldsymbol{x}_k^T \boldsymbol{P}_{1,1} \boldsymbol{x}_k + \\ & 2\boldsymbol{x}_k^T \boldsymbol{P}_{1,2} \boldsymbol{\omega}_k + \\ & \boldsymbol{\omega}_k^T \boldsymbol{P}_{2,2} \boldsymbol{\omega}_k \leq 1 \end{aligned} \tag{66}$$

where $\boldsymbol{P}_{1,1}$, $\boldsymbol{P}_{1,2}$ and $\boldsymbol{P}_{2,2}$ are block partitions of $\boldsymbol{P}$ conformal to the partition of $\boldsymbol{z}_k = [\boldsymbol{x}_k^T \ \boldsymbol{\omega}_k^T]^T$.

Thus, if the $\mathcal{H}_\infty$ control is feasible then $\boldsymbol{\omega} = \boldsymbol{0}$, otherwise the solution lies on the boundary of $\mathcal{E}_z$ described by (66). This means that when $\boldsymbol{\omega} = \boldsymbol{0}$ is contained in $\mathcal{E}_z$ described by (66), then there is no need for optimization and the optimal solution is $\boldsymbol{\omega} = \boldsymbol{0}$. Otherwise, as indicated in [6], the above optimization problem has a unique solution and can be very efficiently solved with, e.g., the Newton-Raphson algorithm [4]. Thus, the structure of the whole robust predictive control can be summarized as follows:

Off-line computation:

1. for a predefined disturbance attenuation level $\mu > 0$, design a robust controller $\boldsymbol{K}$ by solving (23)
2. determine the robust invariant set by solving (64) under the constraints (45), (46), (47).

On-line computation: for each k,

1. solve the optimization problem (66)
2. implement the first element of $\boldsymbol{\omega}_k$, i.e., $\boldsymbol{c}_k$.

5 Conclusions

The main objective of this paper was to deal with the issue of a robust predictive control for a class of non-linear discrete-time systems. The approach begins with a robust controller, which design procedure is provided in a convenient LMI form. Predictive control approaches for nonlinear system with ellipsoidal invariant set was proposed. In the paper two-step procedure was provided. In the off-line phase a maximum ellipsoidal robust invariant set is designed. The on-line computation boils down to a simple one-variable constrained non-linear optimization.

Acknowledgments The authors would like to express their sincere gratitude to the referees, whose constructive comments contributed significantly toward the current shape of the paper. The work was supported by the National Science Center of Poland under grant no. 2013/11/B/ST7/01110.

References

1. Abbaszadeh, M., Marquez, H.: LMI optimization approach to robust h_∞ observer design and static output feedback stabilization for non-linear uncertain systems. Int. J. Robust Nonlinear Control **19**(3), 313–340 (2008)
2. Alessandri, A., Baglietto, M., Battistelli, G.: Design of state estimators for uncertain linear systems using quadratic boundedness. Automatica **42**(3), 497–502 (2006)
3. Aicha, B.F., Bouani, F., Ksouri, M.: A multivariable multiobjective predictive controller. Int. J. Appl. Math. Comput. Sci. **23**(1), 35–45 (2013)
4. Imsland, L., Bar, N., Foss, B.: More efficient predictive control. Automatica **41**(8), 1395–1403 (2005)
5. Korbicz, J., Kościelny, J., Kowalczuk, Z., Cholewa, W. (eds.): Fault Diagnosis. Models, Artificial Intelligence, Applications. Springer, Berlin (2004)
6. Kouvaritakis, B., Cannon, M., Rossiter, J.: Who needs QP for linear MPC anyway? Automatica **38**(5), 879–884 (2002)
7. Ławryńczuk, M., Tatjewski, P.: Nonlinear predictive control based on neural multi-models. Int. J. Appl. Math. Comput. Sci. **20**(1), 7–21 (2010)
8. Maciejowski, J.: Predictive Control With Constraints. Prentice Hall, Upper Saddle River (2002)
9. Marquez, H.: Nonlinear Control Systems. Analysis and Design. Wiley, New Jersey (2003)
10. de Oliveira, M., Bernussou, J., Geromel, J.: A new discrete-time robust stability condition. Syst. Control Lett. **37**(4), 261–265 (1999)
11. Patan, K., Korbicz, J.: Nonlinear model predictive control of a Boiler unit: a fault tolerant control study. Int. J. Appl. Math. Comput. Sci. **22**(1), 225–237 (2012)
12. Pertew, A.M., Marquez, H.J., Zhao, Q.: H_∞ synthesis of unknown input observers for non-linear Lipschitz systems. Int. J. Control **78**(15), 1155–1165 (2005)
13. Rajamani, R.: Observers for lipschitz non-linear systems. IEEE Trans. Autom. Control **43**(3), 397–401 (1998)
14. Tatjewski, P.: Advanced Control of Industrial Processes: Structures and Algorithms. Advances in Industrial Control. Springer, London (2007)
15. Witczak, M.: Fault diagnosis and fault-tolerant control strategies for non-linear systems. Lecture Notes in Electrical Engineering, vol. 266. Springer International Publishing Switzerland, Cham (2014)
16. Zemouche, A., Boutayeb, M., Iulia Bara, G.: Observer design for nonlinear systems: an approach based on the differential mean value theorem. In: Proc. 44th IEEE Conference on Decision and Control, CDC (2005)

Self-healing Control Against Actuator Stuck Failures Under Constraints: Application to Unmanned Helicopters

Xin Qi, Didier Theilliol, Juntong Qi, Youmin Zhang and Jianda Han

Abstract This paper investigates the problem of actuator stuck failures under constraints. In order to guarantee the post-failure system stability and acceptable performance, self-healing control framework is proposed which includes self-healing management module, fault-tolerant controller, reference redesigner and anti-windup compensator. Because of the existence of actuator constraints, the post-failure system may be unstable and the reference may be unreachable. Hence, fault-tolerant controller with anti-windup compensator was used to guarantee stability which was proved by introducing H_∞ performance. Reachability of reference was analyzed by self-healing management module and a new reference could be calculated by reference redesigner. At last, the proposed self-healing framework was applied to a linear unmanned helicopter model for velocities and yaw tracking control.

Keywords Fault-tolerant systems · Actuators · Stuck · Saturation · Autonomous vehicles

X. Qi (✉) · J. Qi · J. Han
Shenyang Institute of Automation, Chinese Academy of Sciences,
Shenyang 110016, China
e-mail: qixin@sia.cn

J. Qi
e-mail: qijt@sia.cn

J. Han
e-mail: jdhan@sia.cn

D. Theilliol
University of Lorraine, CRAN-CNRS, UMR 7039,
BP 70239, 54506 Vandoeuvre Cedex, France
e-mail: didier.theilliol@univ-lorraine.fr

Y. Zhang
Concordia University, Montreal, QC H3G 1M8, Canada
e-mail: ymzhang@encs.concordia.ca

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_14

1 Introduction

Traditional control techniques cannot guarantee correct and safe operation of equipment in the event of malfunctions in actuators [10]. Abundant approaches have been proposed against actuator faults, using hardware redundancy and fault-tolerant control (FTC) techniques [17]. Generally, actuator malfunctions are divided into two categories: faults and failures. Actuator failures signify all efficiency is lost and actuator cannot respond control signal completely. One typical actuator failure is actuator stuck malfunction.

Compared to actuator faults, few researches focused on actuator stuck failures. In [12], an adaptive state feedback method against actuator failures was proposed. Actuator stuck failures can be compensated by remaining fault-free actuators adaptively and the system can keep tracking the original reference. In [15], a stuck failure was modeled as a bounded input and its effect on the closed-loop system is described by a peak-to-peak gain. The FTC controller is designed based on H_∞. In [4], an iterative learning observer (ILO) for actuator stuck failures was presented which can provide both the estimates of the system states and information on the transient of failure compensation. The drawback of these approaches is lack of consideration the constraints of actuators. In this paper, a FTC method against actuator stuck failures under actuator constraints is investigated.

System inputs of stuck actuators have to be compensated by remaining fault-free actuators. Taking into account actuator constraints, the remaining actuators margin is degraded after stuck-failure compensation. Thus, post-failure system may not track the original reference without offset. For the sake of system stability and preventing failure deterioration, reference redesign is necessary. In [6], a new reference of post-failure system was generated according to system remaining performance. The distance between the new reference and the original one before failure is minimum. In [13], a model predictive control strategy was proposed to redesign the new reference on-line which is achieved by solving an optimization problem. The drawback of this method is that the new reference needs to be calculated in real-time. So it is impossible to obtain the steady-state reference at the beginning of failure occurrence. In [16], a control input management approach was investigated to compute a new steady-state reference which is based on the open-loop gain of post-failure system in steady-state case. The method is based on experience so that the new reference may not be optimal. The disadvantage of these methods is the shortage of reachability analysis of original references after actuator stuck occurrence. In this paper, a reference admissible set is computed for analyzing reachability.

Similar researches against actuator failures and constraints are limited such as [3] where flatness technique was used for quadrotors specially. However, actuator constraints and only actuator faults were considered and analysis of reference reachability was absent.

On the other hand, actuator constraints also affect dynamic performance of both fault-free and post-failure systems especially at the moment when controller is switched from fault-free one to fault-tolerant one. One way to improve dynamic

performance is to consider anti-windup techniques. Two categories of anti-windup methods have been proposed. The first one is to take into account anti-windup technology when the system controller is designed [9]. The second one is to design a nominal controller without anti-windup first and then add into an anti-windup compensator. The second method is more popular. In [14], an anti-windup compensator design method was proposed by introducing sector theory. The difference between the controller output and saturated actuator output is assumed bounded and the compensator design problem is recast into a robust control paradigm. In [5], based on invariant theory, both anti-windup compensator and stability domain can be achieved by solving a group of linear matrix inequalities (LMIs). Besides anti-windup techniques, command governor was used as an added primal compensator to modify the reference inputs so as to avoid violation of the constraints [1].

The main contribution of this paper is to present a self-healing control framework against actuator stuck failures under actuator constraints. The self-healing framework includes self-healing management module, reconfigurable controller with anti-windup compensator, reference redesigner and fault diagnosis and identification (FDI) module, as shown in Fig. 1, where *ref* represents the original reference while ref_{new} is the redesigned new reference, f represents actuator faults/failures and ψ is difference between actuator output with/without saturation. The self-healing management module is used to analyze the reachability of the original reference by calculating reference admissible set. It allows to pre-evaluate the reachability of post-failure system before it is in motion. The reconfigurable controller is designed to guarantee the closed-loop stability and tracking performance while an anti-windup module is used to improve dynamic performance of the system when actuators are in saturation. The function of reference redesigner is to compute a new reachable reference which satisfies actuator constraints and conditions of the post-failure system. FDI module is used to detect, isolate and identify the stuck failures. Furthermore, with the information provided by FDI module, the proposed framework can be designed online. In the following discussion, stuck-failure magnitude is assumed to be provided by FDI module correctly with no time delay. The investigation of FDI methods is not included in this paper.

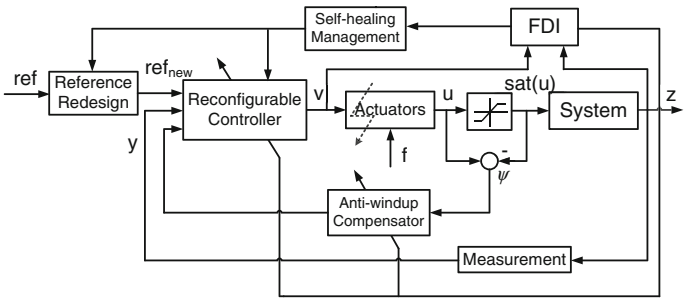


Fig. 1 The structure of self-healing framework

The paper is organized as follows: Problem statement is dedicated in Sect. 2. Section 3 is devoted to the development of the proposed approach. Fault-tolerant control method is proposed with related stability certification. Then, reachability of reference and reference redesign are investigated based on achieved admissible set of reference. In Sect. 4, a linear unmanned helicopter (UH) model is considered to illustrate the proposed method. Section 5 ends the paper.

2 Problem Statement

Consider an open-loop linear time-invariant (LTI) system which is stabilizable and detectable as following:

$$\begin{cases} \dot{x}(t) = A_p x(t) + B_p sat(u(t)) \\ z(t) = C_{p1} x(t) \\ y(t) = C_{p2} x(t) \end{cases} \tag{1}$$

where $x \in R^n$ is the system state vector. $u \in R^m$ is the system input vector neglecting actuator constraints. $z \in R^{p1}$ is the system controlled output vector, and $y \in R^{p2}$ is the system measurement output vector. $A_p,\ B_p,\ C_{p1},\ C_{p2}$ are constant matrices with appropriate dimensions. Furthermore, $B_p = [b_1\ b_2\ \dots\ b_m]$, where b_i represents the ith column of matrix B_p. $sat(\cdot)$ represents a vector function defined by

$$sat(u_i(t)) = \begin{cases} u_i^{max} & if \quad u_i(t) > u_i^{max} \\ u_i(t) & if \quad u_i^{min} \le u_i(t) \le u_i^{max} \\ u_i^{min} & if \quad u_i(t) < u_i^{min} \end{cases} \tag{2}$$

for $i = 1, 2, ..., m$. u^{min}, u^{max} are actuator constraints, and $sat(u(t))$ is the control input vector with actuator constraints. Assume that $u^{min} = -u^{max} = \tilde{u}$.

Actuator stuck failures can be modeled as following:

$$u(t) = \Phi v(t) + (I - \Phi)\bar{u} \tag{3}$$

where $v(t)$ represents controller output vector, and $\bar{u}$ is a constant vector representing the magnitude of stuck failure. $\Phi = diag(\phi_1,\ \phi_2, \dots,\ \phi_m)$, and $\phi_i = 1\ or\ 0$ where $\phi_i = 1$ represents that the ith actuator is fault-free and $\phi_i = 0$ represents that the ith actuator is lock-in-place because the actuator cannot respond the control signal completely.

Hence, the control input vector neglecting actuator constraints can be divided into two parts $u(t) = [u_0^T(t)\ \bar{u}_f^T]^T$ in stuck-failure case, where $u_0(t) \in R^{m_0}$ is the fault-free control input, $\bar{u}_f \in R^{m_f}$ is the stuck control input which is constant and $m_0 + m_f = m$.

The control matrix B_p can be decomposed into $[B_{p0}\ B_{pf}]$ with $B_{p0} \in R^{n\times m_0}$ and $B_{pf} \in R^{n\times m_f}$ correspondingly.

Then system (1) in stuck-failure case can be described as:

$$\begin{cases} \dot{x}(t) = A_p x(t) + B_{p0} sat(u_0(t)) + B_{pf} sat(\bar{u}_f) \\ z(t) = C_{p1} x(t) \\ y(t) = C_{p2} x(t) \end{cases} \tag{4}$$

Compared with the fault-free system defined by (1), the post-failure system with stuck actuators has degraded input matrix and an additional constant item. The degraded input matrix will affect state and output controllability so that the following assumption is necessary:

Assumption The post-failure system $(Ap,\ B_{p0})$ is stabilizable.

The additional constant item will affect state value in steady case. Taking into account actuator constraints, the problem will be more troublesome. Actuator constraints affect the post-failure system in two ways:

- *Global stability may not be guaranteed.* Because of actuator constraints, actuator output is limited so that the controllable domain may not be guaranteed as the global space. Thus, regional stability is considered instead [8].
- *Reference may be unreachable.* For a set-point tracking problem, the following relationship is established in steady case:

$$ref = z(\infty) = \begin{cases} H(\infty) sat(u(\infty)) & Normal \\ H_0(\infty) sat(u_0(\infty)) + H_f(\infty)\bar{u}_f & Stuck \end{cases} \tag{5}$$

  where ref is reference, $H(\infty) = [H_0(\infty)\ H_f(\infty)]$ is open-loop gain in steady case, $H_0(\infty)$ and $H_f(\infty)$ are related to fault-free and post-failure actuators respectively. Because actuator allowance is consumed by stuck-failure compensation, the second equation might not be valid. In other words, the reference may be unreachable.

Hence, the proposed self-healing control framework should solve the following two problems:

Problem 1 Design a fault-tolerant controller to guarantee stability of the post-failure system with acceptable set-point tracking under actuator stuck failures and saturation.

Problem 2 Analyze reference reachability subject to closed-loop post-failure system and compute a new one if required.

3 Main Results

3.1 Fault-Tolerant Control Method

In addition to guarantee the post-failure system being stable, the task of fault-tolerant controller also includes compensating stuck failures with capability of tracking set-points with offset-free and anti-windup. The three functions are achieved by three additional items of the dynamic output feedback controller. Hence, the dynamic fault-tolerant controller against post-failure system (4) is proposed as following.

$$\begin{cases} \dot{x}_c(t) = A_c x_c(t) + B_c y(t) + E_c(sat(u_0(t)) - u_0(t)) + K_f \bar{u}_f + K_r ref \\ u_0(t) = C_c x_c(t) + D_c y(t) \end{cases} \tag{6}$$

where $x_c \in R^{n_c}$ is controller state vector which has the same dimension as the open-loop system $n_c = n$, $u_0 \in R^{m_0}$ is controller output vector, ref is reference vector, E_c is anti-windup compensator, K_f is stuck-failure compensator, and K_r is feedforward gain matrices. A_c, B_c, C_c, and D_c are constant feedback controller matrices. Note that, controller (6) can be divided into four parts: (1) Matrices A_c, B_c, C_c, and D_c consist a classical dynamic output feedback controller which can be designed according to (1) without considering actuator saturation [5]. The design methods of A_c, B_c, C_c, D_c are classical and out of research scope of this paper; (2) Anti-windup compensator E_c which can be designed based on a nominal closed-loop system; (3) Stuck-failure compensator K_f is used to reduce the impact of actuator stuck and (4) Feedforward controller K_r is used to guarantee tracking performance. The design method of matrices E_c, K_f and K_r will be introduced in the following.

Define extended state vector $\xi(t) = \left[x^T(t) \;\; x_c^T(t)\right]^T \in R^{n+n_c}$, exogenous input vector $\omega = \left[\bar{u}_f^T \;\; ref^T\right]^T$, and function $\psi(u_0(t)) = u_0(t) - sat(u_0(t))$. Then, the post-failure closed-loop system can be described by:

$$\begin{cases} \dot{\xi}(t) = A\xi(t) - (B_0 + RE_c)\psi(K\xi(t)) + D\omega \\ z(t) = C\xi(t) \end{cases} \tag{7}$$

where

$$A = \begin{bmatrix} A_p + B_{p0} D_c C_{p2} & B_{p0} C_c \\ B_c C_{p2} & A_c \end{bmatrix}, \; R = \begin{bmatrix} 0 \\ I_{n_c} \end{bmatrix}$$

$$B_0 = \begin{bmatrix} B_{p0} \\ 0 \end{bmatrix}, \; B_f = \begin{bmatrix} B_{pf} \\ K_f \end{bmatrix}, \; L = \begin{bmatrix} 0 \\ K_r \end{bmatrix}$$

$$D = \begin{bmatrix} B_f & L \end{bmatrix}, \; C = \begin{bmatrix} C_{p1} & 0 \end{bmatrix} \;\text{ and }\; K = \begin{bmatrix} D_c C_{p2} & C_c \end{bmatrix}$$

Considering steady-state case of system (7), if all of the fault-free actuators are not saturated, which means that $\psi(u_0(t)) = 0$, the controlled output will be

$$\begin{aligned} z(\infty) = &- C_{p1}\left[A_p + B_{p0}(-C_c A_c^{-1} B_c + D_c)C_{p2}\right]^{-1} \\ &\left[(-B_{p0}C_c A_c^{-1}K_f + B_{pf})\bar{u}_f - B_{p0}C_c A_c^{-1}K_r ref\right] \end{aligned} \tag{8}$$

In order to compensate actuator stuck failures, the coefficient of $\bar{u}_f$ should be equal to zero. Thus, the stuck-failure compensator can be calculated by:

$$K_f = \left(C_{p1}MB_{p0}C_c A_c^{-1}\right)^{+} C_{p1}MB_{pf} \tag{9}$$

where $M = \left[A_p + B_{p0}(-C_c A_c^{-1}B_c + D_c)C_{p2}\right]^{-1}$ and $(\cdot)^{+}$ represents pseudo inverse.

For set-point tracking problem, $z(\infty) = ref$ should be satisfied. Thus, the tracking matrix can be computed by:

$$K_r = (C_{p1}MB_{p0}C_c A_c^{-1})^{+} \tag{10}$$

Remark Considering columns of matrix B_p, if $\alpha_1 b_{i_1} + \alpha_2 b_{i_2} + \cdots + \alpha_q b_{i_q} = 0,\ \alpha_i \neq 0$ is satisfied, the column rank of B_p will be less than m. In other words, assume $Rank(B_p) = Rank(B_{p0}) = q < m$, then $B_{pf} = B_{p0}Q$ where Q is a constant matrix. So the control inputs of stuck actuators can be compensated by the remaining actuators directly such as $u_0 = Q\bar{u}_f$. Most of the research works against actuator stuck failures are under the above condition [4, 12]. Obviously, these results will be useless when $Rank(B_p) < m$ is not satisfied such as $Rank(B_p) = m$. Compared to these method, the proposed fault-tolerant controller design method in this paper can work under both $Rank(B_p) < m$ and $Rank(B_p) = m$.

Before anti-windup compensator design and stability certification, the following lemma [5] is recalled which is required for the coming theorem.

Lemma 1 *Consider a matrix $G \in R^{m\times(n+n_c)}$ and define the following polyhedral set:*

$$\Omega = \left\{\xi(t)\mid\ |(K_i - G_i)\xi(t)| \leq \tilde{u}_i,\ i = 1, \ldots, m\right\} \tag{11}$$

where i represents the ith row of matrix K and G.

If $\xi(t) \in \Omega$, then the relation

$$\psi(K\xi(t))^T T\left[\psi(K\xi(t)) - G\xi(t)\right] \leq 0 \tag{12}$$

is verified for any positive-definite matrix $T \in R^{m\times m}$.

Clearly, Lemma 1 defines a set of system states related to actuator saturation and the relation based on the set is useful for the coming theorem.

Theorem 1 *Given $\gamma > 0$ and a symmetric positive-definite matrix $R \in R^{(n+n_c)\times(n+n_c)}$, if there exist a symmetric positive-definite matrix $W \in R^{(n+n_c)\times(n+n_c)}$, matrices $Y \in R^{m\times(n+n_c)}$, $Z \in R^{n_c\times m}$, and a diagonal positive-definite matrix $S \in R^{m\times m}$ satisfying*

$$\inf_{W,Y,Z,S} \quad \lambda$$

$$\begin{bmatrix} W & WK_i^T - Y_i^T \\ * & \tilde{u}_i^2 \end{bmatrix} \geq 0, \quad i = 1, \ldots, m \tag{13}$$

$$\begin{bmatrix} \frac{1}{\gamma}(WA^T + AW) & \frac{1}{\gamma}(B_0 S + RZ - Y^T) & \frac{1}{\gamma}B_f & \frac{1}{\gamma}L & -WC^T \\ * & -\frac{2}{\gamma}S & 0 & 0 & 0 \\ * & * & -\gamma I & 0 & 0 \\ * & * & * & -\gamma I & I \\ * & * & * & * & -\gamma I \end{bmatrix} < 0 \tag{14}$$

$$\begin{bmatrix} \lambda R & I \\ * & W \end{bmatrix} \geq 0 \tag{15}$$

then the anti-windup compensator is $E_c = ZS^{-1}$, and the stability domain is $\varepsilon(P) = \left\{\xi(t) \,|\xi^T(t)P\xi(t) \leq 1\right\}$ with $P = W^{-1}$.

Proof If relations in (13) are valid, $\varepsilon(P) \subset \Omega$ will be satisfied with $G = YP$ [2]. Thus, $\forall \xi(t) \in \varepsilon(P)$, $\psi(K\xi(t))$ satisfies sector condition (12) [5]. Taking into account H_∞ performance $\|ref - C\xi(t)\|_2 \leq \gamma\|\omega\|_2$, it can be written as

$$J = \int_0^T \left[(ref - C\xi(t))^T(ref - C\xi(t)) - \gamma^2\omega^T\omega\right] dt < 0$$

Considering zero initial condition and Lyapunov function $V(\xi) = \xi(t)^T P\xi(t)$,

$$J = \int_0^T J_1 dt - V(\xi(T))$$

where $J_1 = (ref - C\xi(t))^T(ref - C\xi(t)) - \gamma^2\omega^T\omega + \frac{d}{dt}V(\xi(t))$. Clearly, if $J_1 < 0$ is satisfied, H_∞ performance will be guaranteed. In the following analysis, symbol t will be ignored and $\psi(K\xi(t))$ will be replaced by ψ for simplicity. According to Lemma 1,

$$
\begin{aligned}
J_1 &\leq (ref - C\xi)^T (ref - C\xi) - \gamma\omega^T\omega + \dot{V}(\xi) - 2\psi(K\xi)^T T\,[\psi(K\xi) - G\xi(t)] \\
&= \begin{bmatrix} \xi \\ -\psi \\ \bar{u}_f \\ ref \end{bmatrix}^T \left(\begin{bmatrix} -C^T \\ 0 \\ 0 \\ I \end{bmatrix} [\,-C\ 0\ 0\ I\,] \right. \\
&\quad \left. + \begin{bmatrix} A^TP+PA & P(B_0+RE_c)+G^TT & PB_f & PH \\ * & -2T & 0 & 0 \\ * & * & -\gamma^2 I & 0 \\ * & * & * & -\gamma^2 I \end{bmatrix} \right) \begin{bmatrix} \xi \\ -\psi \\ \bar{u}_f \\ ref \end{bmatrix} \\
&\leq 0
\end{aligned}
$$

Considering Schur complement, the following inequality is achieved:

$$
\begin{bmatrix} A^TP+PA & P(B_0+RE_c)+G^TT & PB_f & PH & -C^T \\ * & -2T & 0 & 0 & 0 \\ * & * & -\gamma^2 I & 0 & 0 \\ * & * & * & -\gamma^2 I & I \\ * & * & * & * & -I \end{bmatrix} \leq 0
$$

Then pre- and post-multiplying the above inequality by $diag\left[\gamma^{-1/2}P^{-1}\ \gamma^{-1/2}T^{-1}\ \gamma^{-1/2}I\ \gamma^{-1/2}I\ \gamma^{1/2}I\right]$ and considering $W = P^{-1}$, $S = T^{-1}$, $Y = GP^{-1}$, and $Z = E_C T^{-1}$, relation (14) is achieved. If relation (14) is valid, $\dot{V}(\xi) \leq 0$ will be satisfied. Thus, $\forall\xi(t) \in \varepsilon(P)$, $\varepsilon(P)$ is a positively invariant and contractive region. In other words, $\varepsilon(P)$ is the stability domain for system (7).

Finally, the objectives to be optimized $inf\ \lambda$ and relation (15) are used to enlarge the domain [7].

Thus, a solution of Problem 1 has been found.

3.2 Self-healing Management

The main target of self-healing management is to analyze the remaining capability of post-failure system and select suitable strategy to guarantee system stabilization with acceptable performance. In order to guarantee post-failure system stability, system states in steady case should be inside stability domain achieved by Theorem 1 such as

$$\xi(\infty)^T P \xi(\infty) \leq 1 \tag{16}$$

Note that, $\xi(\infty) \in \varepsilon(P) \subset \Omega$ can guarantee $|(K-G)\xi(\infty)| \leq \tilde{u}$ but not $|K\xi(\infty)| \leq \tilde{u}$. In other words, actuators may be saturated. In order to guarantee offset-free tracking performance, control input should never be saturated in steady-state case. Thus, system inputs of post-failure system should satisfy $|u_0(\infty)| = |K\xi(\infty)| \leq \tilde{u}$ where $u_0(\infty)$ can be described as function of $\bar{u}_f$, and *ref* such as:

$$u_0(\infty) = M_1\bar{u}_f + M_2 ref \tag{17}$$

where

$$M_1 = -C_c A_c^{-1} K_f$$
$$M_2 = \left[(-C_c A_c^{-1} B_c + D_c) C_{p2} M B_{p0} - I\right] C_c A_c^{-1} K_r$$

Because M_1, M_2, and $\bar{u}_f$ can be obtained based on information of system and FDI module, the system inputs in steady case can be achieved under reference *ref*.

Thus, reference admissible set can be defined as:

$$Y = \left\{ ref \middle| \, \xi(\infty) \in \varepsilon(P), |u_0(\infty)| \leq \tilde{u} \right\} \tag{18}$$

If $ref \in Y$ is not valid, the original reference *ref* is recognised unreachable for the post-failure system. Hence, a new reference is required instead of the original one and Problem 2 is solved.

3.3 Reference Redesign

The target of reference redesign is to find a new optimal reference to guarantee the post-failure system stable with acceptable performance. If the new reference ref_{new} is expected to be as close as possible to the original one *ref*, the following optimization problem can be defined.

$$\min_{ref_{new}} \|N(ref_{new} - ref)\|_2$$

subject to:

$$ref_{new} \in Y$$

where $N \in R^{p1 \times p1}$ is a diagonal weighting matrix. Thus, new optimal reference can be achieved and reachability is guaranteed.

4 Application to Unmanned Helicopters

A linear model of Unmanned helicopter, Fig. 2, including swashplate configuration and rotor speed control [11] is considered in this paper.

4.1 Unmanned Helicopter Model

The state vector is $x = [u\ v\ w\ \varphi\ \theta\ \psi\ p\ q\ r\ a_{1s}\ b_{1s}]^T$ where u, v, w are triaxial velocities, $\varphi\ \theta\ \psi$ are attitudes, $p\ q\ r$ are triaxial angular velocities, and $a_{1s}\ b_{1s}$ are flapping angle of main rotor. The control input vector is $u = [\theta_{M1}\ \theta_{M2}\ \theta_{M3}\ \theta_T\ \Omega]^T$,

Fig. 2 The unmanned helicopter

where the first four variables are output positions of servos for main rotor and tail rotor and the last one is rotor speed. The UH output vector is $z = [u\ v\ w\ \psi]^T$. Note, that actuator constraints are normalized in this paper, such as $\tilde{u} = [1\ 1\ 1\ 1\ 1]$, by multiplying an index matrix behind matrix B_p of [11] and the eigenvalues of the system matrix A_p are [0 −28.9183 10.0359 −5.9322+9.4855i −5.9322−9.4855i 1.5746 − 1.2855 − 0.0085 + 0.3257i − 0.0085 − 0.3257i − 0.0001 − 0.0128] which means it is an open-loop unstable system.

4.2 Simulation Results

Assume the first actuator is stuck at 10 s with $\bar{u}_f = 0.3$ so that $B_{pf} = [b_1]$, and $B_{p0} = [b_2\ b_3\ b_4\ b_5]$. The stuck-failure compensator K_f and tracking matrix K_r can be calculated by (9) and (10). Anti-windup compensator E_c and related stability domain $\varepsilon(P)$ are achieved by the proposed Theorem 1. Stuck-failure information is assumed to be provided by FDI module without time delay.

Stabilization problem is illustrated firstly. Assume the initial sates of u, v, w are [2 2 −1] and the others are zero. Simulation results of control inputs and system outputs are shown in Figs. 3 and 4 respectively. Dash lines represent inputs and outputs of post-failure system with fault-free controller. As shown in Fig. 3, after the first actuator being stuck, the other four fault-free control inputs oscillate between the upper and lower bounds of constraints. Clearly, stability of the post-failure system cannot be guaranteed and outputs are out of order as shown in Fig. 4. In other words, after actuator stuck failures occurrence, the post-failure system will be out of order if controller is not reconfigured. Post-failure system with fault-tolerant controller is represented by solid lines. As shown in the figures, system stability is guaranteed and actuator outputs are not saturated. On the other hand, due to the existence of

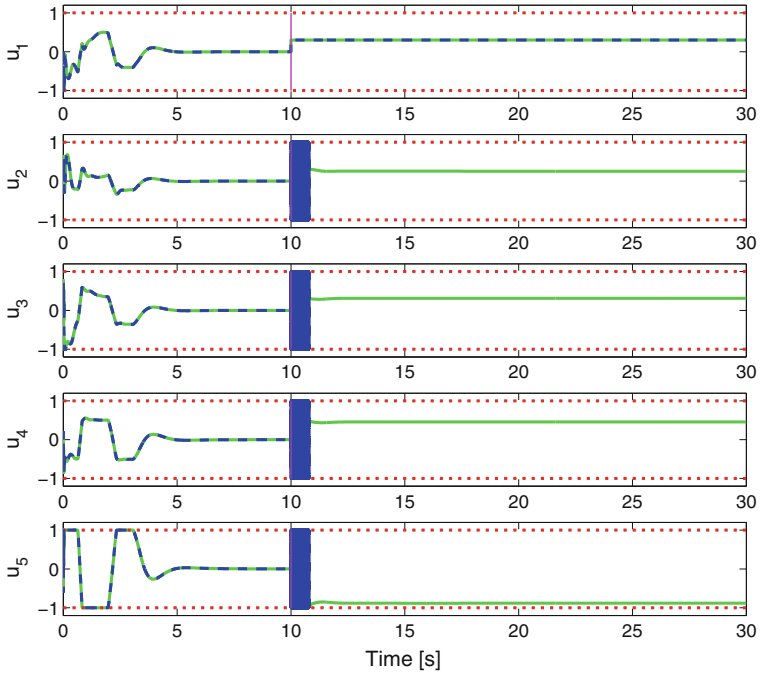


Fig. 3 Control inputs of stabilization problem

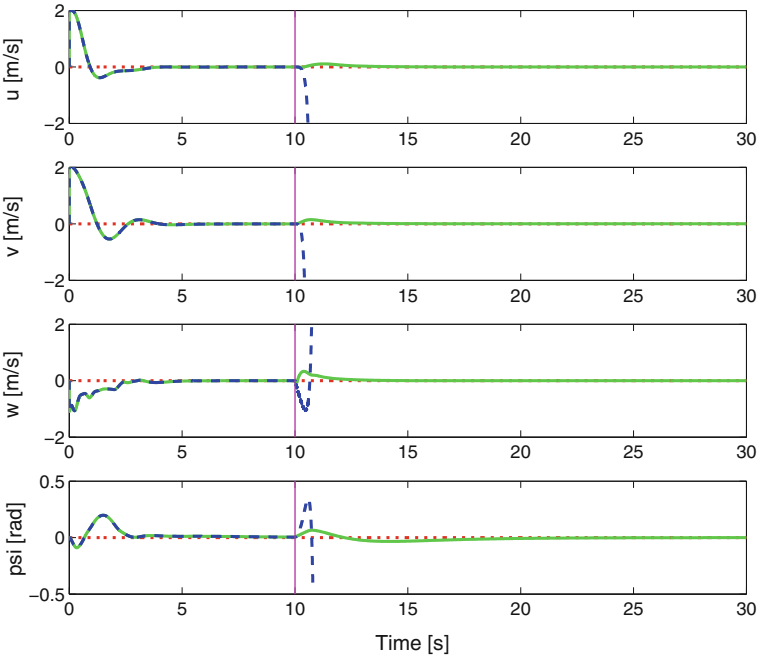


Fig. 4 System outputs of stabilization problem

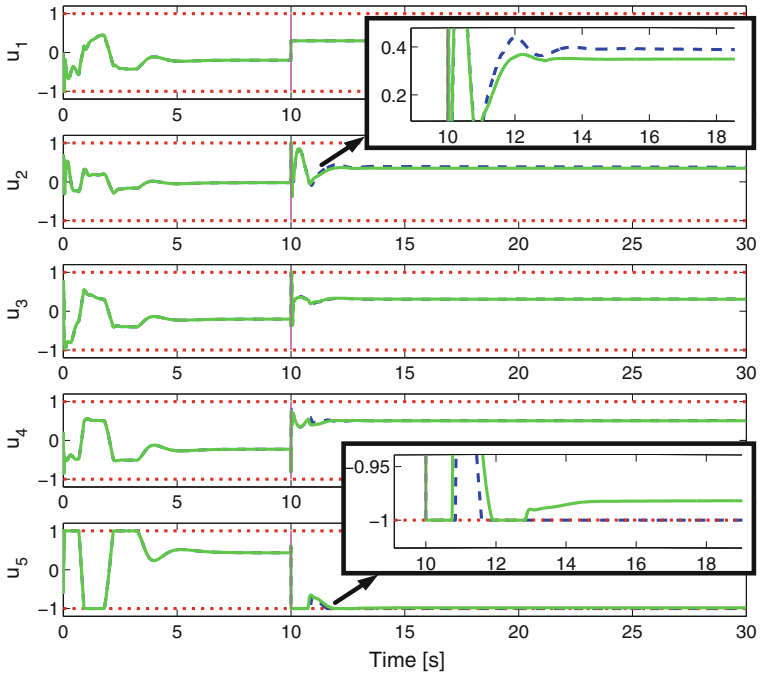


Fig. 5 Control inputs of set-point tracking

anti-windup compensator, actuator saturation is allowed during dynamic process as shown in Fig. 3.

For set-point tracking problem, the original reference for fault-free system is $ref = [1\ 1\ 1\ 0]^T$. Simulation results are shown in Figs. 5 and 6. Post-failure system with fault-free controller is represented by dash lines. As shown in the figures, the post-failure system is stable at last. However, because the tracked reference is the original one and, as analyzed before, it is not inside the admissible set of reference, the fifth actuator is saturated and the outputs cannot track the reference. Thus, analyzing reference reachability before system motion is necessary.

Based on self-healing management, the remaining control inputs of post-failure system in steady-state case can be obtained by (17) such as $u_0(\infty) = [0.3974\ \ 0.3079\ \ 0.5308\ \ -1.0301]$ which is outside the actuator constraints $\tilde{u}$. At the same time, $\xi(\infty)^T P \xi(\infty) = 2.4032 > 1$. In other words, the reference is unreachable for the post-failure system because of $ref \notin Y$. Thus, new reference is required. Based on the proposed reference redesign method, the optimal new reference is $ref_{new} = [0.61\ 0.67\ 0.4\ 0.05]^T$ with $N = [1\ 1\ 0.1\ 1]$. Post-failure system with self-healing control framework, including both fault-tolerant controller and reference redesign, is drawn by solid lines. As shown in Fig. 5, all of fault-free actuators are not saturated in steady case and Fig. 6 shows that system outputs can track the new reference without offset.

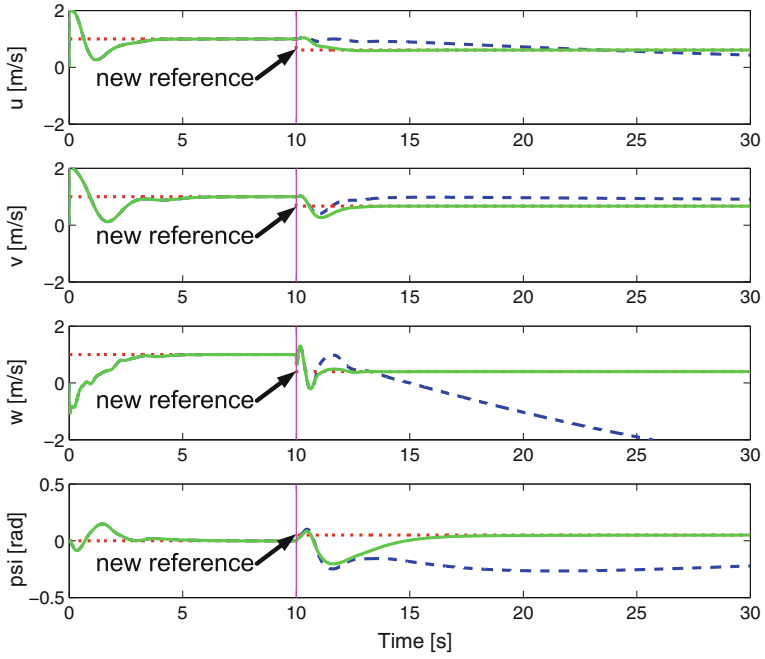


Fig. 6 System outputs of set-point tracking

5 Conclusions

Self-healing control framework against actuator stuck failures under constraints is proposed in this paper to guarantee the post-failure system stability and achieve acceptable performance. The proposed framework mainly includes fault-tolerant controller, reference redesign module, self-healing management module and anti-windup compensator. According to related information, self-healing management module can analyze reference reachability before system motion and reference redesign module can calculate an optimal new one if it is required. At last, the self-healing control method is illustrated by an unmanned helicopter model with velocities and yaw set-point tracking. Based on the proposed framework, stability and tracking performance of the post-failure system can be guaranteed. However, the fault-tolerant control, self-healing management and reference redesign methods are interconnected but the three parts are considered separately in the proposed framework. In the future, the three parts will be integrated together to reduce conservative.

Acknowledgments This work was supported by National Key Technology Research and Development Program of China under Grant: 2013BAK03B02; The Chinese Academy of Sciences Visiting Professorship for Senior International Scientists under Grant 2012T1G0007.

References

1. Bemporad, A., Casavola, A., Mosca, E.: Nonlinear control of constrained linear systems via predictive reference management. IEEE Trans. Autom. Control **42**(3), 340–349 (1997)
2. Boyd, S., Ghaoui, L.E., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in Systems and Control Theory. SIAM, Philadelphia (1994)
3. Chamseddine, A., Zhang, Y., Rabbath, C.A., Join, C., Theilliol, D.: Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle. IEEE Trans. Aerosp. Electron. Syst. **48**(4), 2832–2848 (2012)
4. Chen, W., Jiang, J.: Fault-tolerant control against stuck actuator faults. IEEE Proc. Control Theory Appl. **152**(2), 138–146 (2005)
5. da Silva, J.M.G., Tarbouriech, S.: Antiwindup design with guaranteed regions of stability: an lmi-based approach. IEEE Trans. Autom. Control **50**(1), 106–111 (2005)
6. Dardinier-Maron, V., Hamelin, F., Noura, H.: A fault-tolerant control design against major actuator failures: application to a three-tank system. In: Proceedings of the 38th IEEE Conference on Decision and Control. vol. 4, pp. 3569–3574. Arizona, USA (1999)
7. Hu, T., Lin, Z., Chen, B.M.: Analysis and design for discrete-time linear systems subject to actuator saturation. Syst. Control Lett. **45**(2), 97–112 (2002)
8. Hu, T., Miller, D.E., Qiu, L.: Null controllable region of LTI discrete-time systems with input saturation. Automatica **38**(11), 2009–2013 (2002)
9. Kapila, V., Grigoriadis, K.: Actuator Saturation Control. Marcel Dekker Inc, New York (2002)
10. Noura, H., Theilliol, D., Ponsart, J.C., Chamseddine, A.: Fault-tolerant Control Systems: Design and Practical Applications. Advances in Industrial Control. Springer, London (2009)
11. Qi, X., Theilliol, D., Qi, J., Zhang, Y., Wang, L., Han, J.: Self healing control method against unmanned helicopter actuator stuck faults. In: International Conference on Unmanned Aircraft Systems. Orlando, Fl, USA (2014)
12. Tao, G., Joshi, S.M., Ma, X.: Adaptive state feedback and tracking control of systems with actuator failures. IEEE Trans. Autom. Control **46**(1), 78–95 (2001)
13. Theilliol, D., Join, C., Zhang, Y.: Actuator fault tolerant control design based on a reconfigurable reference input. Int. J. Appl. Math. Comput. Sci. **18**(4), 553–560 (2008)
14. Wu, F., Lu, B.: Anti-windup control design for exponentially unstable lti systems with actuator saturation. Syst. Control Lett. **52**(3–4), 305–322 (2004)
15. Yang, G., Lum, K.Y.: Fault-tolerant flight tracking control with stuck faults. In: Proceedings of the 2003 American Control Conference. vol. 1, pp. 521–526 (2003)
16. Zhang, Y., Jiang, J.: Fault tolerant control system design with explicit consideration of performance degradation. IEEE Trans. Aerosp. Electron. Syst. **39**(3), 838–848 (2003)
17. Zhang, Y., Jiang, J.: Bibliographical review on reconfigurable fault-tolerant control systems. Annu. Rev. Control **32**(2), 229–252 (2008)

H_∞ Approach to Virtual Actuators Design

Dušan Krokavec, Anna Filasová, Vladimír Serbák and Pavol Liščinský

Abstract The H_∞ approach to virtual actuators design, intended for linear continuous-time systems, is presented in the paper. Using the H_∞ principle, new conditions for virtual actuators design in P and PI structures are formulated in terms of linear matrix inequalities. Related to the static output control under influence of single actuator faults, an example is presented to highlight the benefit of the proposed framework.

Keywords Linear systems · Virtual actuators · Fault tolerant control

1 Introduction

To increase the reliability of systems, fault tolerant control (FTC) usually fix a system with faults so that it can continue its mission with certain limitations of functionality and quality. Considering this, the different approaches were studied in FTC design (see, e.g., [1, 3, 10, 15] and the references therein). The standard approach to control reconfiguration discards the nominal controller from the control loop and replace it with a new one so that its parameters are re-tuned in occurred fault conditions and, in dependency on the remaining set of sensors and actuators, to recover in a certain extent the performance of the fault-free control system [6, 9, 14, 16]. By contrast,

D. Krokavec (✉) · A. Filasová · V. Serbák · P. Liščinský
Faculty of Electrical Engineering and Informatics, Department of Cybernetics and Artificial Intelligence, Technical University of Košice, Letná 9, 042 00 Košice, Slovakia
e-mail: dusan.krokavec@tuke.sk
URL: http://web.tuke.sk/kkui

A. Filasová
e-mail: anna.filasova@tuke.sk

V. Serbák
e-mail: vladimir.serbak@tuke.sk

P. Liščinský
e-mail: pavol.liscinsky@tuke.sk

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_15

instead of adapting the controller to the faulty plant, the virtual approach keeps the nominal controller in the reconfigured closed-loop system and virtually adapt the faulty plant to the nominal controller in such a way that the virtual reconfiguration block together with the faulty plant imitates the fault-free plant [7, 12]. The reconfiguration block is chosen so as to hide the fault from the controller point of view (the fault-hiding paradigm) and offers a way for the minimum invasive control reconfiguration. Designated to sensor faults the reconfiguration block is termed virtual sensor (VS), while in the case of actuator faults is named virtual actuator (VA).

The technique proposed in the paper is inspired by the virtual manner used in [2, 8, 13] so that VA is designed using linear matrix inequality (LMI) concept and, extending the bounded real lemma approach, H_∞ norm principle is proposed for design of VA in proportional (P) as well as proportional-integral (PI) structure. To achieve the desired nominal control objective, the static output controller design method is considered [5].

The paper is organized as follows. Continuing with basic preliminaries presentation in Sect. 2, the proposed design methods exploiting H_∞ approach are given in Sects. 3 and 4, stating there desired specifications as well discussing the LMI forms of the design conditions. In response, Sect. 5 shows the performance of the proposed approach using an application example and Sect. 5 gives some concluding remarks.

Throughout the paper, the following notations are used: $\boldsymbol{x}^T$, $\boldsymbol{X}^T$ denotes the transpose of the vector $\boldsymbol{x}$ and the matrix $\boldsymbol{X}$, respectively, for a square matrix $\boldsymbol{X} < 0$ means that $\boldsymbol{X}$ is symmetric negative definite matrix, the symbol $\boldsymbol{I}_n$ indicates the n-th order unit matrix, $\mathbb{R}$ denotes the set of real numbers and $\mathbb{R}^n$, $\mathbb{R}^{n\times r}$ refers to the set of all n-dimensional real vectors and $n \times r$ real matrices, respectively.

2 Basic Preliminaries

In the paper, there are taken into account the continuous-time linear dynamic systems described in the fault-free conditions as

$$\dot{\boldsymbol{q}}(t) = \boldsymbol{A}\boldsymbol{q}(t) + \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{V}\boldsymbol{d}(t) \tag{1}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{q}(t) \tag{2}$$

where $\boldsymbol{q}(t) \in \mathbb{R}^n$ stands up for the system state, $\boldsymbol{u}_c(t) \in \mathbb{R}^r$ denotes the control input, $\boldsymbol{y}(t) \in \mathbb{R}^m$ is the measurable output, $\boldsymbol{d}(t) \in \mathbb{R}^p$ is the vector of unknown disturbance, the matrices $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times r}$, $\boldsymbol{C} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{V} \in \mathbb{R}^{n\times p}$ are finite valued and $r = m$.

If the pair $(\boldsymbol{A}, \boldsymbol{B})$ is stabilizable (all its hidden modes are stable) and the pair $(\boldsymbol{A}, \boldsymbol{C})$ is detectable [4] then the square system (1), (2) can be stabilized by the static output feedback

$$\boldsymbol{u}_c(t) = -\boldsymbol{K}\boldsymbol{y}(t) = -\boldsymbol{K}\boldsymbol{C}\boldsymbol{q}(t) \tag{3}$$

where $\boldsymbol{K} \in \mathbb{R}^{r\times m}$ is the control gain matrix. Therefore, the nominal autonomous closed-loop system with unknown disturbance is described as

$$\dot{\boldsymbol{q}}(t) = (\boldsymbol{A} - \boldsymbol{BKC})\boldsymbol{q}(t) + \boldsymbol{Vd}(t) \tag{4}$$

$$\boldsymbol{y}(t) = \boldsymbol{Cq}(t) \tag{5}$$

The state-space description of the system with a single actuator fault is considered as follows

$$\dot{\boldsymbol{q}}_{fa}(t) = \boldsymbol{A}\boldsymbol{q}_{fa}(t) + \boldsymbol{B}_f\boldsymbol{u}_{fa}(t) + \boldsymbol{Vd}(t) \tag{6}$$

$$\boldsymbol{y}_{fa}(t) = \boldsymbol{C}\boldsymbol{q}_{fa}(t) \tag{7}$$

where $\boldsymbol{q}_{fa}(t) \in \mathbb{R}^n$, $\boldsymbol{u}_{fa}(t) \in \mathbb{R}^{r_f}$, $\boldsymbol{y}_{fa}(t) \in \mathbb{R}^m$ denote the system state variables vector, the vector of the acting control input variables and the vector of output variables, respectively, and the matrix $\boldsymbol{B}_f \in \mathbb{R}^{n\times r_f}$ is finite valued, where $rank(\boldsymbol{B}_f) < rank(\boldsymbol{B})$. Moreover, it is supposed that the pair $(\boldsymbol{A}, \boldsymbol{B}_f)$ is controllable and the input vector $\boldsymbol{u}_{fa}(t)$ is available for reconfiguration (all inputs to the plant are available as they use the nominal controller, but one associated with the faulty actuator is broken).

The used structure of the static output control is given in (3) and the stability condition has to reflect the closed-loop system (4), (5). The design conditions are then given by the following lemma. The dynamic controllers design methodology for linear systems with virtual actuators is presented in [7].

Lemma 1 [5] *The nominal static output feedback control (3) to the system (1), (2) exists if there exist a symmetric positive definite matrix* $\boldsymbol{T} \in \mathbb{R}^{n\times n}$ *and matrices* $\boldsymbol{W} \in \mathbb{R}^{r\times m}$, $\boldsymbol{H} \in \mathbb{R}^{m\times m}$ *such that*

$$\boldsymbol{T} = \boldsymbol{T}^T > 0 \tag{8}$$

$$\boldsymbol{AT} + \boldsymbol{TA}^T - \boldsymbol{BWC} - \boldsymbol{C}^T\boldsymbol{W}^T\boldsymbol{B}^T < 0 \tag{9}$$

$$\boldsymbol{CT} = \boldsymbol{HC} \tag{10}$$

When the above conditions hold, the gain matrix $\boldsymbol{K}$ *is given by the relation*

$$\boldsymbol{K} = \boldsymbol{WH}^{-1} \tag{11}$$

3 Virtual Actuator

In this case the plant with a faulty actuator is modified by adding a VA block that masks the actuator fault, and allows the controller to perceive the system as it was before the fault, i.e., the nominal controller may still be used without it being necessary readjusted.

Writing (1) and (6) compactly as

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{q}}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{q}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{fa}(t) \\ \boldsymbol{u}_c(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{V} \end{bmatrix} \boldsymbol{d}(t) \tag{12}$$

the extended system model behavior can be described using $\boldsymbol{q}_{fa}(t)$ and the equation for the error vector

$$\boldsymbol{e}_{fa}(t) = \boldsymbol{q}_{fa}(t) - \boldsymbol{q}(t) \tag{13}$$

Therefore, to perform the coordinate change, the transform matrix $\boldsymbol{T}$ is defined with respect to (13) as follows

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix}, \qquad \boldsymbol{T}^{-1} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix} \tag{14}$$

where $\boldsymbol{I} \in \mathbb{R}^{n \times n}$ is identity matrix, then it yields

$$\boldsymbol{T} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{q}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{q}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \end{bmatrix}, \quad \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{V} \end{bmatrix} = \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \end{bmatrix} \tag{15}$$

$$\begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} \end{bmatrix}, \quad \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{B}_f & -\boldsymbol{B} \end{bmatrix} \tag{16}$$

and (12) can be rewritten into the following form

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{e}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{B}_f & -\boldsymbol{B} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{fa}(t) \\ \boldsymbol{u}_c(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{d}(t) \tag{17}$$

Introducing the faulty input estimation as follows

$$\boldsymbol{u}_{fa}(t) = -\boldsymbol{G}\boldsymbol{e}_{fa}(t) \tag{18}$$

where $\boldsymbol{G} \in \mathbb{R}^{r_f \times n}$ and substituting (18) in (17) then

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{e}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \end{bmatrix} - \begin{bmatrix} \boldsymbol{B}_f\boldsymbol{G} \\ \boldsymbol{B}_f\boldsymbol{G} \end{bmatrix} \boldsymbol{e}_{fa}(t) - \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{B} \end{bmatrix} \boldsymbol{u}_c(t) + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{d}(t) \tag{19}$$

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{e}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & -\boldsymbol{B}_f\boldsymbol{G} \\ \boldsymbol{0} & \boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{G} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \end{bmatrix} - \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{B} \end{bmatrix} \boldsymbol{u}_c(t) + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{d}(t) \tag{20}$$

respectively. Note, (20) is an alternative description of the same connected system and contains the same information as (12), since $\boldsymbol{q}_{fa}(t)$ and $\boldsymbol{e}_{fa}(t)$ uniquely determine $\boldsymbol{q}_{efa}(t)$. Because of the block structure of the connected system matrix in (20) the

separation property holds, i.e., the gain matrix $\boldsymbol{G}$ can be designed independently on the fault system description if $(\boldsymbol{A}, \boldsymbol{B}_f)$ is controllable.

Thus, VA is given by (18) and the second part of (20) [13], i.e.,

$$\dot{\boldsymbol{e}}_{fa}(t) = (\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{G})\boldsymbol{e}_{fa}(t) - \boldsymbol{B}\boldsymbol{u}_c(t) \tag{21}$$

$$\boldsymbol{u}_{fa}(t) = -\boldsymbol{G}\boldsymbol{e}_{fa}(t) \tag{22}$$

Since one can write

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{q}(t) = \boldsymbol{C}(\boldsymbol{q}_{fa}(t) - (\boldsymbol{q}_{fa}(t) - \boldsymbol{q}(t))) \tag{23}$$

then with (7) and (13) it yields

$$\boldsymbol{y}(t) = \boldsymbol{y}_{fa}(t) - \boldsymbol{C}\boldsymbol{e}_{fa}(t) \tag{24}$$

and the control law acting after the reconfiguration is

$$\boldsymbol{u}_c(t) = -\boldsymbol{K}\boldsymbol{y}(t) = -\boldsymbol{K}\boldsymbol{y}_{fa}(t) + \boldsymbol{K}\boldsymbol{C}\boldsymbol{e}_{fa}(t) \tag{25}$$

while $\boldsymbol{e}_{fa}(t)$ is obtained from VA and $\boldsymbol{y}_{fa}(t)$ is the measured output of the plant with the actuator fault.

Moreover, the first part of (20) implies

$$\dot{\boldsymbol{q}}_{fa}(t) = \boldsymbol{A}\boldsymbol{q}_{fa}(t) - \boldsymbol{B}_f\boldsymbol{G}\boldsymbol{e}_{fa}(t) + \boldsymbol{V}\boldsymbol{d}(t) \tag{26}$$

which can be rewritten using (13) as

$$\dot{\boldsymbol{q}}_{fa}(t) = (\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{G})\boldsymbol{q}_{fa}(t) + \begin{bmatrix}\boldsymbol{B}_f & \boldsymbol{V}\end{bmatrix}\begin{bmatrix}\boldsymbol{G}\boldsymbol{q}(t) \\ \boldsymbol{d}(t)\end{bmatrix} \tag{27}$$

It is obvious that the system with a single actuator fault and VA is working after reconfiguration in the mode with an unknown extended external input disturbance $\boldsymbol{d}_{fa}(t)$, where

$$\boldsymbol{d}_{fa}^T(t) = \begin{bmatrix}\boldsymbol{q}^T(t)\boldsymbol{G}^T & \boldsymbol{d}^T(t)\end{bmatrix}, \quad \boldsymbol{V}_{fa} = \begin{bmatrix}\boldsymbol{B}_f & \boldsymbol{V}\end{bmatrix} \tag{28}$$

and $\boldsymbol{V}_{fa} \in I\!R^{n \times r_{fa}}$, $r_{fa} = r_f + p$.

The comparison of (21) and (27) implies that after reconfiguration the dynamic of the closed loop system with VA and the dynamic of VA are same. Thus, the proposed novelty is to reckon in design conditions the structure of (27) and to reflect the disturbance (28). In such a way, given the structure (21), (22) of VA, the H_∞ norm-based design conditions are provided in the following theorem.

Theorem 1 *The virtual actuator (21), (22) is asymptotically stable if there exist a positive definite symmetric matrix* $\boldsymbol{X} \in \mathbb{R}^{n\times n}$, *a matrix* $\boldsymbol{Z} \in \mathbb{R}^{r_f\times n}$ *and a positive scalar* $\gamma \in \mathbb{R}$ *such that*

$$\boldsymbol{X} = \boldsymbol{X}^T > 0, \qquad \gamma > 0 \tag{29}$$

$$\begin{bmatrix} \boldsymbol{A}\boldsymbol{X} + \boldsymbol{X}\boldsymbol{A}^T - \boldsymbol{B}_f\boldsymbol{Z} - \boldsymbol{Z}^T\boldsymbol{B}_f^T & * & * \\ \boldsymbol{V}_{fa}^T & -\gamma\boldsymbol{I}_{r_{fa}} & * \\ \boldsymbol{C}\boldsymbol{X} & \boldsymbol{0} & -\gamma\boldsymbol{I}_m \end{bmatrix} < 0 \tag{30}$$

where $\boldsymbol{I}_{r_{fa}} \in \mathbb{R}^{r_{fa}\times r_{fa}}$, $\boldsymbol{I}_m \in \mathbb{R}^{m\times m}$ *are identity matrices.*

Then, if the above conditions hold, the gain matrix $\boldsymbol{G}$ *of VA is computed as*

$$\boldsymbol{G} = \boldsymbol{Z}\boldsymbol{X}^{-1} \tag{31}$$

Proof Defining the Lyapunov function candidate as follows [6]

$$\begin{aligned} &v(\boldsymbol{q}_{fa}(t)) = \\ &= \boldsymbol{q}_{fa}^T(t)\boldsymbol{P}\boldsymbol{q}_{fa}(t) + \gamma^{-1}\int_0^t \left(\boldsymbol{y}_{fa}^T(v)\boldsymbol{y}_{fa}(v) - \gamma^2\boldsymbol{d}_{fa}^T(v)\boldsymbol{d}_{fa}(v)\right)\mathrm{d}v > 0 \end{aligned} \tag{32}$$

where $\boldsymbol{P} = \boldsymbol{P}^T > 0$, $\boldsymbol{P} \in \mathbb{R}^{n\times n}$, $\gamma \in \mathbb{R}$ and $\gamma > 0$ is the H_∞ norm of the disturbance transfer function matrix then, evaluating the derivative of $v(\boldsymbol{q}_{fa}(t))$ with respect to t along the faulty system trajectory, it yields

$$\begin{aligned} &\dot{v}(\boldsymbol{q}_{fa}(t)) = \\ &= \dot{\boldsymbol{q}}_{fa}^T(t)\boldsymbol{P}\boldsymbol{q}_{fa}(t) + \boldsymbol{q}_{fa}^T(t)\boldsymbol{P}\dot{\boldsymbol{q}}_{fa}(t) \ + \gamma^{-1}\boldsymbol{y}_{fa}^T(t)\boldsymbol{y}_{fa}(t) - \gamma\boldsymbol{d}_{fa}^T(t)\boldsymbol{d}_{fa}(t) < 0 \end{aligned} \tag{33}$$

Then, substituting (27) into (33), it yields

$$\begin{aligned} &\dot{v}(\boldsymbol{q}_{fa}(t)) = \\ &= \boldsymbol{q}_{fa}^T(t)\boldsymbol{A}_{caf}^T\boldsymbol{P}\boldsymbol{q}_{fa}(t) + \boldsymbol{q}_{fa}^T(t)\boldsymbol{P}\boldsymbol{A}_{caf}\boldsymbol{q}_{fa}(t) + \boldsymbol{d}_{fa}^T(t)\boldsymbol{V}_{fa}^T\boldsymbol{P}\boldsymbol{q}_{fa}(t) + \\ &+ \boldsymbol{q}_{fa}^T(t)\boldsymbol{P}\boldsymbol{V}_{fa}\boldsymbol{d}_{fa}(t) + \gamma^{-1}\boldsymbol{q}_{fa}^T(t)\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{q}_{fa}(t) - \gamma\boldsymbol{d}_{fa}^T(t)\boldsymbol{d}_{fa}(t) < 0 \end{aligned} \tag{34}$$

where

$$\boldsymbol{A}_{caf} = \boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{G} \tag{35}$$

Thus, with the notation

$$\boldsymbol{q}_{fa}^{\diamond T}(t) = \left[\boldsymbol{q}_{fa}^T(t)\ \boldsymbol{d}_{fa}^T(t)\right] \tag{36}$$

it is obtained

$$\dot{v}(\boldsymbol{q}_{fa}^{\diamond}(t)) = \boldsymbol{q}_{fa}^{\diamond T}(t)\boldsymbol{P}_{fa}\boldsymbol{q}_{fa}^{\diamond}(t) < 0 \tag{37}$$

while

$$\boldsymbol{P}_{fa} = \begin{bmatrix} \gamma^{-1}\boldsymbol{C}^T\boldsymbol{C} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} + \begin{bmatrix} (\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{G})^T\boldsymbol{P}+\boldsymbol{P}(\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{G}) & \boldsymbol{P}\boldsymbol{V}_{fa} \\ \boldsymbol{V}_{fa}^T\boldsymbol{P} & -\gamma\boldsymbol{I}_{r_{fa}} \end{bmatrix} < 0 \tag{38}$$

Since

$$\begin{bmatrix} \gamma^{-1}\boldsymbol{C}^T\boldsymbol{C} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{C}^T \\ \boldsymbol{0} \end{bmatrix} \gamma^{-1}\boldsymbol{I}_m \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \tag{39}$$

the Schur complement property implies

$$\begin{bmatrix} (\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{G})^T\boldsymbol{P}+\boldsymbol{P}(\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{G}) & \boldsymbol{P}\boldsymbol{V}_{fa} & \boldsymbol{C}^T \\ \boldsymbol{V}_{fa}^T\boldsymbol{P} & -\gamma\boldsymbol{I}_{r_{fa}} & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{0} & -\gamma\boldsymbol{I}_m \end{bmatrix} < 0 \tag{40}$$

Defining the transform matrix

$$\boldsymbol{T}_f = \text{diag}\begin{bmatrix} \boldsymbol{X} & \boldsymbol{I}_{r_{fa}} & \boldsymbol{I}_m \end{bmatrix}, \quad \boldsymbol{X} = \boldsymbol{P}^{-1} \tag{41}$$

pre-multiplying the left and right sides of (41) by $\boldsymbol{T}_f$, it yields

$$\begin{bmatrix} \boldsymbol{X}(\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{G})^T+(\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{G})\boldsymbol{X} & \boldsymbol{V}_{fa} & \boldsymbol{X}\boldsymbol{C}^T \\ \boldsymbol{V}_{fa}^T & -\gamma\boldsymbol{I}_{r_{fa}} & \boldsymbol{0} \\ \boldsymbol{C}\boldsymbol{X} & \boldsymbol{0} & -\gamma\boldsymbol{I}_m \end{bmatrix} < 0 \tag{42}$$

Thus, with the notation

$$\boldsymbol{Z} = \boldsymbol{G}\boldsymbol{X} \tag{43}$$

the condition (42) implies (30). This concludes the proof. ■

4 PI Virtual Actuator

If in the reconfiguration structure an integrator is a part of the virtual actuator, then a new vector variable $z_{fa}(t)$, satisfying the following differential equation

$$\dot{z}_{fa}(t) = \boldsymbol{C}\boldsymbol{e}_{fa}(t) = \boldsymbol{C}(\boldsymbol{q}_{fa}(t) - \boldsymbol{q}(t)) \tag{44}$$

has to be introduced. Therefore, (12) and (44) are reformulated as

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{q}}(t) \\ \dot{\boldsymbol{z}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{C} & -\boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{q}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{fa}(t) \\ \boldsymbol{u}_c(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{V} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{q}(t) \tag{45}$$

Defining the transform matrix $\boldsymbol{T}$ with respect to (45) as follows

$$\boldsymbol{T} = \boldsymbol{T}^{-1} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{I} & -\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I}_m \end{bmatrix} \tag{46}$$

where $\boldsymbol{I}_m \in \mathbb{R}^{m \times m}$ is identity matrix, then it yields

$$\boldsymbol{T} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{q}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix}, \quad \boldsymbol{T} \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{B}_f & -\boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \tag{47}$$

$$\boldsymbol{T} \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{C} & -\boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \boldsymbol{T}^{-1} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C} & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{T} \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{V} \\ \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \tag{48}$$

and (45) can be rewritten into the following form

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{e}}_{fa}(t) \\ \dot{\boldsymbol{z}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{0} \\ \boldsymbol{B}_f & -\boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{fa}(t) \\ \boldsymbol{u}_c(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{q}(t) \tag{49}$$

Introducing the faulty input estimation as follows

$$\boldsymbol{u}_{fa}(t) = -\boldsymbol{G}_P \boldsymbol{e}_{fa}(t) - \boldsymbol{G}_I \boldsymbol{z}_{fa}(t) \tag{50}$$

where $\boldsymbol{G}_P \in \mathbb{R}^{r_f \times n}$, $\boldsymbol{G}_I \in \mathbb{R}^{r_f \times m}$ and, by substituting (50) in (49), then

$$\begin{aligned} \begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{e}}_{fa}(t) \\ \dot{\boldsymbol{z}}_{fa}(t) \end{bmatrix} &= \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} - \begin{bmatrix} \boldsymbol{B}_f \boldsymbol{G}_P \\ \boldsymbol{B}_f \boldsymbol{G}_P \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{e}_{fa}(t) \\ &\quad - \begin{bmatrix} \boldsymbol{B}_f \boldsymbol{G}_I \\ \boldsymbol{B}_f \boldsymbol{G}_I \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{z}_{fa}(t) - \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{u}_c(t) + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{d}(t) \end{aligned} \tag{51}$$

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{e}}_{fa}(t) \\ \dot{\boldsymbol{z}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & -\boldsymbol{B}_f \boldsymbol{G}_P & -\boldsymbol{B}_f \boldsymbol{G}_I \\ \boldsymbol{0} & \boldsymbol{A} - \boldsymbol{B}_f \boldsymbol{G}_P & -\boldsymbol{B}_f \boldsymbol{G}_I \\ \boldsymbol{0} & \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{e}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} - \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{u}_c(t) + \begin{bmatrix} \boldsymbol{V} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{d}(t) \tag{52}$$

respectively. Because of the block structure of the connected system (52) the separation property holds, i.e., the gain matrices $\boldsymbol{G}_P$, $\boldsymbol{G}_I$ can be designed independently on the fault system description if $(\boldsymbol{A}, \boldsymbol{B}_f)$ is controllable.

Therefore, VA is given by (50) and the last parts of (52), i.e.,

$$\begin{bmatrix} \dot{\boldsymbol{e}}_{fa}(t) \\ \dot{\boldsymbol{z}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} - \boldsymbol{B}_f \boldsymbol{G}_P & -\boldsymbol{B}_f \boldsymbol{G}_I \\ \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} - \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{u}_c(t) \tag{53}$$

$$\boldsymbol{u}_{fa}(t) = -\begin{bmatrix} \boldsymbol{G}_P & \boldsymbol{G}_I \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} \tag{54}$$

where (23)–(25) implies

$$\boldsymbol{u}_c(t) = -\boldsymbol{K}\boldsymbol{y}(t) = -\boldsymbol{K}\boldsymbol{y}_{fa}(t) + \boldsymbol{K} \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} \tag{55}$$

while $\boldsymbol{e}_{fa}(t)$, $\boldsymbol{z}_{fa}(t)$ are obtained from VA and $\boldsymbol{y}_{fa}(t)$ is the measured output of the plant with the actuator fault.

Moreover, the first row of (52) implies

$$\dot{\boldsymbol{q}}_{fa}(t) = \boldsymbol{A}\boldsymbol{q}_{fa}(t) - \boldsymbol{B}_f \boldsymbol{G}_P \boldsymbol{e}_{fa}(t) - \boldsymbol{B}_f \boldsymbol{G}_I \boldsymbol{z}_{fa}(t) + \boldsymbol{V}\boldsymbol{d}(t) \tag{56}$$

and using (13), then (56) can be written together with the last row of (52) as

$$\begin{bmatrix} \dot{\boldsymbol{q}}_{fa}(t) \\ \dot{\boldsymbol{z}}_{fa}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} - \boldsymbol{B}_f \boldsymbol{G}_P & -\boldsymbol{B}_f \boldsymbol{G}_I \\ \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_{fa}(t) \\ \boldsymbol{z}_{fa}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{V} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{C} \end{bmatrix} \begin{bmatrix} \boldsymbol{G}\boldsymbol{q}(t) \\ \boldsymbol{d}(t) \\ \boldsymbol{q}(t) \end{bmatrix} \tag{57}$$

Introducing the notations

$$\boldsymbol{A}^\circ = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{B}_f^\circ = \begin{bmatrix} \boldsymbol{B}_f \\ \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{V}_{fa}^\circ = \begin{bmatrix} \boldsymbol{B}_f & \boldsymbol{V} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{C} \end{bmatrix} \tag{58}$$

$$\boldsymbol{C}^\circ = \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{G}^\circ = \begin{bmatrix} \boldsymbol{G}_P & \boldsymbol{G}_I \end{bmatrix} \tag{59}$$

$$\boldsymbol{q}_{fa}^{\circ T}(t) = \begin{bmatrix} \boldsymbol{q}_{fa}^T(t) & \boldsymbol{z}_{fa}^T(t) \end{bmatrix}, \quad \boldsymbol{d}_{fa}^{\circ T} = \begin{bmatrix} \boldsymbol{q}^T(t)\boldsymbol{G}_P^T & \boldsymbol{d}^T(t) & \boldsymbol{q}^T(t) \end{bmatrix} \tag{60}$$

where $\boldsymbol{A}^\circ \in I\!R^{(n+m)\times(n+m)}$, $\boldsymbol{B}_f^\circ \in I\!R^{(n+m)\times r_f}$, $\boldsymbol{V}_{fa}^\circ \in I\!R^{(n+m)\times r_{fa}^\circ}$, $\boldsymbol{C}^\circ \in I\!R^{m\times(n+m)}$, $\boldsymbol{G}^\circ \in I\!R^{r_f\times(n+m)}$, $r_{fa}^\circ = r_f + p + n$, then it is possible to write (57) as

$$\dot{\boldsymbol{q}}_{fa}^\circ(t) = (\boldsymbol{A}^\circ - \boldsymbol{B}_f^\circ \boldsymbol{G}^\circ)\boldsymbol{q}_{fa}^\circ(t) + \boldsymbol{V}_{fa}^\circ \boldsymbol{d}_{fa}^\circ(t) \tag{61}$$

while the output of the faulty system (7) takes the form

$$\boldsymbol{y}_{fa}(t) = \boldsymbol{C}^\circ \boldsymbol{q}_{fa}^\circ(t) \tag{62}$$

Considering (58), (59), it is evident that the system matrices of (53) and (61) are the same and reflecting the unknown disturbance properties (60), the H_∞ norm-based design conditions are provided in the following theorem.

Theorem 2 *The virtual actuator (53), (54) is asymptotically stable if there exist a positive definite symmetric matrix* $\boldsymbol{X}^\circ \in \mathbb{R}^{(n+m)\times(n+m)}$, *a matrix* $\boldsymbol{Z}^\circ \in \mathbb{R}^{r_f\times(n+m)}$ *and a positive scalar* $\gamma^\circ \in \mathbb{R}$ *such that*

$$\boldsymbol{X}^\circ = \boldsymbol{X}^{\circ T} > 0, \qquad \gamma^\circ > 0 \tag{63}$$

$$\begin{bmatrix} \boldsymbol{A}^\circ\boldsymbol{X}^\circ + \boldsymbol{X}^\circ\boldsymbol{A}^{\circ T} - \boldsymbol{B}_f^\circ\boldsymbol{Z}^\circ - \boldsymbol{Z}^{\circ T}\boldsymbol{B}_f^{\circ T} & * & * \\ \boldsymbol{V}_{fa}^{\circ T} & -\gamma^\circ\boldsymbol{I}_{r_{fa}^\circ} & * \\ \boldsymbol{C}^\circ\boldsymbol{X}^\circ & \boldsymbol{0} & -\gamma^\circ\boldsymbol{I}_m \end{bmatrix} < 0 \tag{64}$$

where $\boldsymbol{I}_{r_{fa}^\circ} \in \mathbb{R}^{r_{fa}^\circ\times r_{fa}^\circ}$, $\boldsymbol{I}_m \in \mathbb{R}^{m\times m}$ *are identity matrices.*

If the above conditions hold, the gain matrix $\boldsymbol{G}^\circ$ *of VA can be computed as*

$$\boldsymbol{G}^\circ = \boldsymbol{Z}^\circ(\boldsymbol{X}^\circ)^{-1} \tag{65}$$

Proof Defining the Lyapunov function candidate

$$\begin{aligned} &v(\boldsymbol{q}_{fa}^\circ(t)) = \\ &= \boldsymbol{q}_{fa}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{q}_{fa}^\circ(t) + \gamma^{\circ-1}\int_0^t \left(\boldsymbol{y}_{fa}^T(v)\boldsymbol{y}_{fa}(v) - \gamma^{\circ 2}\boldsymbol{d}_{fa}^{\circ T}(v)\boldsymbol{d}_{fa}^\circ(v)\right)\mathrm{d}v > 0 \end{aligned} \tag{66}$$

where $\boldsymbol{P}^\circ = \boldsymbol{P}^{\circ T} > 0$, $\boldsymbol{P}^\circ \in \mathbb{R}^{(n+m)\times(n+m)}$, $\gamma^\circ \in \mathbb{R}$ and $\gamma^\circ > 0$ is the H_∞ norm of the generalized disturbance transfer matrix, then

$$\begin{aligned} &\dot v(\boldsymbol{q}_{fa}^\circ(t)) = \\ &= \dot{\boldsymbol{q}}_{fa}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{q}_{fa}^\circ(t) + \boldsymbol{q}_{fa}^{\circ T}(t)\boldsymbol{P}^\circ\dot{\boldsymbol{q}}_{fa}^\circ(t) + \gamma^{\circ-1}\boldsymbol{y}_{fa}^T(t)\boldsymbol{y}_{fa}(t) - \gamma^\circ\boldsymbol{d}_{fa}^{\circ T}(t)\boldsymbol{d}_{fa}^\circ(t) < 0 \end{aligned} \tag{67}$$

and, substituting (61), (62) into (67), it yields

$$\begin{aligned} &\dot v(\boldsymbol{q}_{fa}^\circ(t)) = \\ &= \boldsymbol{q}_{fa}^{\circ T}(t)\boldsymbol{A}_{caf}^{\circ T}\boldsymbol{P}^\circ\boldsymbol{q}_{fa}^\circ(t) + \boldsymbol{q}_{fa}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{A}_{caf}^\circ\boldsymbol{q}_{fa}^\circ(t) + \boldsymbol{d}_{fa}^{\circ T}(t)\boldsymbol{V}_{fa}^{\circ T}\boldsymbol{P}^\circ\boldsymbol{q}_{fa}^\circ(t) + \\ &+ \boldsymbol{q}_{fa}^{\circ T}(t)\boldsymbol{P}^\circ\boldsymbol{V}_{fa}^\circ\boldsymbol{d}_{fa}^\circ(t) + \gamma^{\circ-1}\boldsymbol{q}_{fa}^{\circ T}(t)\boldsymbol{C}^{\circ T}\boldsymbol{C}^\circ\boldsymbol{q}_{fa}^\circ(t) - \gamma^\circ\boldsymbol{d}_{fa}^{\circ T}(t)\boldsymbol{d}_{fa}^\circ(t) < 0 \end{aligned} \tag{68}$$

where

$$\boldsymbol{A}_{caf}^\circ = \boldsymbol{A}^\circ - \boldsymbol{B}_f^\circ\boldsymbol{G}^\circ \tag{69}$$

Thus, with the notation

$$\boldsymbol{q}_{fa}^{\bullet T}(t) = \begin{bmatrix} \boldsymbol{q}_{fa}^{\circ T}(t) & \boldsymbol{d}_{fa}^{\circ T}(t) \end{bmatrix} \tag{70}$$

it is obtained

$$\dot{v}(\boldsymbol{q}^{\bullet}_{fa}(t)) = \boldsymbol{q}^{\bullet T}_{fa}(t)\boldsymbol{P}^{\bullet}_{fa}\boldsymbol{q}^{\bullet}_{fa}(t) < 0 \tag{71}$$

$$\boldsymbol{P}^{\bullet}_{fa} = \begin{bmatrix} (\boldsymbol{A}^{\circ} - \boldsymbol{B}^{\circ}_{f}\boldsymbol{G}^{\circ})^{T}\boldsymbol{P}^{\circ} + \boldsymbol{P}^{\circ}(\boldsymbol{A}^{\circ} - \boldsymbol{B}^{\circ}_{f}\boldsymbol{G}^{\circ}) + \gamma^{\circ -1}\boldsymbol{C}^{\circ T}\boldsymbol{C}^{\circ} & \boldsymbol{P}^{\circ}\boldsymbol{V}^{\circ}_{fa} \\ \boldsymbol{V}^{\circ T}_{fa}\boldsymbol{P}^{\circ} & -\gamma^{\circ}\boldsymbol{I}_{r^{\circ}_{fa}} \end{bmatrix} < 0 \tag{72}$$

and so the Schur complement property implies

$$\begin{bmatrix} (\boldsymbol{A}^{\circ} - \boldsymbol{B}^{\circ}_{f}\boldsymbol{G}^{\circ})^{T}\boldsymbol{P}^{\circ} + \boldsymbol{P}^{\circ}(\boldsymbol{A}^{\circ} - \boldsymbol{B}^{\circ}_{f}\boldsymbol{G}^{\circ}) & \boldsymbol{P}^{\circ}\boldsymbol{V}^{\circ}_{fa} & \boldsymbol{C}^{\circ T} \\ \boldsymbol{V}^{\circ T}_{fa}\boldsymbol{P}^{\circ} & -\gamma^{\circ}\boldsymbol{I}_{r^{\circ}_{fa}} & \boldsymbol{0} \\ \boldsymbol{C}^{\circ} & \boldsymbol{0} & -\gamma^{\circ}\boldsymbol{I}_{m} \end{bmatrix} < 0 \tag{73}$$

Defining the transform matrix

$$\boldsymbol{T}^{\circ}_{f} = \text{diag}\left[\boldsymbol{X}^{\circ}\ \boldsymbol{I}_{r^{\circ}_{fa}}\ \boldsymbol{I}_{m}\right], \quad \boldsymbol{X}^{\circ} = (\boldsymbol{P}^{\circ})^{-1} \tag{74}$$

and, pre-multiplying the left and right sides of (74) by $\boldsymbol{T}^{\circ}_{f}$, it yields

$$\begin{bmatrix} \boldsymbol{X}^{\circ}(\boldsymbol{A}^{\circ} - \boldsymbol{B}^{\circ}_{f}\boldsymbol{G}^{\circ})^{T} + (\boldsymbol{A}^{\circ} - \boldsymbol{B}^{\circ}_{f}\boldsymbol{G}^{\circ})\boldsymbol{X}^{\circ} & \boldsymbol{V}^{\circ}_{fa} & \boldsymbol{X}^{\circ}\boldsymbol{C}^{\circ T} \\ \boldsymbol{V}^{\circ T}_{fa} & -\gamma^{\circ}\boldsymbol{I}_{r^{\circ}_{fa}} & \boldsymbol{0} \\ \boldsymbol{C}^{\circ}\boldsymbol{X}^{\circ} & \boldsymbol{0} & -\gamma^{\circ}\boldsymbol{I}_{m} \end{bmatrix} < 0 \tag{75}$$

Thus, with the notation

$$\boldsymbol{Z}^{\circ} = \boldsymbol{G}^{\circ}\boldsymbol{X}^{\circ} \tag{76}$$

the condition (75) implies (64). This concludes the proof. ■

Note, the PI VA can be designed only for a loss of gain in the single actuator.

5 Illustrative Example

The state space representation (1), (2) consists of the following matrices

$$\boldsymbol{A} = \begin{bmatrix} -1.0522 & -1.8666 & 0.5102 \\ -0.4380 & -5.4335 & 0.9205 \\ -0.5522 & 0.1334 & -0.4898 \end{bmatrix}, \ \boldsymbol{B} = \begin{bmatrix} 3 & 1 \\ 1 & 1 \\ 3 & 0 \end{bmatrix}, \ \boldsymbol{V} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}, \ \boldsymbol{C} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

and this system, in closed loop structure under the state feedback (3), was used in the presented simulations.

Firstly, the gain matrix $\boldsymbol{K}$ was synthesized solving (8)–(10) using the SeDuMi package [11], where the following LMI matrix variables and the controller gain matrix were produced

$$\boldsymbol{T} = \begin{bmatrix} 0.4340 & -0.0206 & -0.1356 \\ -0.0206 & 0.3220 & 0.0236 \\ -0.1356 & 0.0236 & 0.4782 \end{bmatrix}, \quad \boldsymbol{H} = \begin{bmatrix} 0.4134 & -0.1120 \\ -0.1562 & 0.5017 \end{bmatrix}$$

$$\boldsymbol{W} = \begin{bmatrix} 0.2007 & 0.1111 \\ -0.6103 & -0.7253 \end{bmatrix}, \; \boldsymbol{K} = \begin{bmatrix} 0.6215 & 0.3602 \\ -2.2086 & -1.9387 \end{bmatrix}$$

Note, the solution of LMIs set was feasible and the design condition results a stable set of the the closed loop system matrix eigenvalues

$$\varrho(\boldsymbol{A} - \boldsymbol{BKC}) = \{-1.2534, \; -1.6465 \pm 3.2462\,\mathrm{i}\}$$

Then, considering the second actuator fault (the second column of $\boldsymbol{B}$ is zero column), the gain matrix $\boldsymbol{G}$ was designed solving (29), (30), where

$$\gamma = 11.7633$$

$$\boldsymbol{X} = \begin{bmatrix} 5.3954 & -0.4419 & -0.2043 \\ -0.4419 & 1.8258 & 0.2550 \\ -0.2043 & 0.2550 & 5.0056 \end{bmatrix}, \quad \boldsymbol{Z} = \begin{bmatrix} -0.0244 & -0.0493 & 1.2043 \\ 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

Using the obtained virtual actuator gain matrix

$$\boldsymbol{G} = \begin{bmatrix} -0.0003 & -0.0611 & 0.2437 \\ 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

the virtual actuator system matrix eigenvalue spectrum is

$$\varrho(\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{G}) = \{-0.5064, \; -1.5853, \; -5.5529\}$$

Finally, by solving (63), (64) for the loss of gain of the second actuator, the obtained results for PI VA were

$$\boldsymbol{B}_f = \begin{bmatrix} 3 & 0.001 \\ 1 & 0.001 \\ 3 & 0.001 \end{bmatrix}, \; \boldsymbol{X}^\circ = \begin{bmatrix} 6.8459 & 0.0821 & 3.7837 & -5.5746 & 2.0216 \\ 0.0821 & 3.2836 & -0.3647 & 0.1555 & -1.4111 \\ 3.7837 & -0.3647 & 7.6188 & -0.6284 & -5.0182 \\ -5.5746 & 0.1555 & -0.6284 & 21.8810 & -6.0957 \\ 2.0216 & -1.4111 & -5.0182 & -6.0957 & 25.5394 \end{bmatrix}$$

$$\gamma = 22.0902\,, \quad \boldsymbol{Z}^\circ = \begin{bmatrix} 0.7706 & -0.1467 & 1.3666 & 2.7283 & 1.3662 \\ 21.0645 & -8.3697 & -18.2744 & 8.4805 & -12.1680 \end{bmatrix}$$

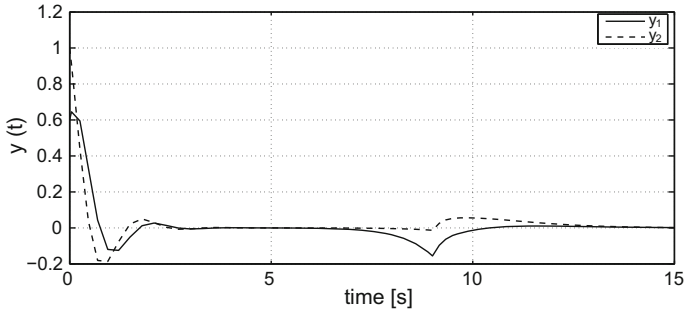


Fig. 1 The reconstructed output of the system with VA

$$\boldsymbol{G}^\circ = \left[\begin{array}{ccc|cc} 0.0942 & 0.0320 & 0.2439 & 0.1951 & 0.1423 \\ 11.5923 & -5.4705 & -10.3495 & 2.1891 & -3.2074 \end{array}\right]$$

$$\varrho(\boldsymbol{A}^\circ - \boldsymbol{B}_f^\circ \boldsymbol{G}^\circ) = \{-0.0033,\ -0.4795,\ -5.5949,\ -0.9752 \pm 0.4932\,\mathrm{i}\}$$

In the simulation, $\boldsymbol{q}(0) = [\,0.2\;0.4\;0.6\,]$ and the autonomous mode was established. As the results, Fig. 1 presents the system outputs response to the second actuator fault, starting and continuing from the time instant $t = 5$ s, where the virtual actuator (21), (22) was being applied in the time instant $t = 9$ s (reflecting the fault detection and isolation time delay lasted approximately 4 s). Note, after the fault of the second actuator occurs, the system controlled by the nominal regulator is unstable. If the static error in forced mode with static output controller would be unacceptable, it is necessary to use PI VAs. Due to the limited extent of the contribution, other simulations are not included in the paper.

6 Concluding Remarks

Using presented VAs, the virtual elimination of an actuator fault influence on the system output is analyzed to obtain the minimum invasive control reconfiguration, adapting the faulty plant to the nominal controller by hiding faults from the controller input point of view without redesign of the nominal controller. The proposed H_∞ based methods present new design features where it was emphasized that the advantage offered by such approach is a collection of feasible algorithms with enough robustness to disturbances. The design conditions of both VA types are accounted in terms of LMIs and use the standard numerical optimization operations. The virtual parts were formulated as autonomous algorithms that may be performed online starting with dependence on the FDIR detection subsystem fault localization time.

Acknowledgments The work presented in this paper was supported by VEGA, the Grant Agency of the Ministry of Education and the Academy of Science of Slovak Republic under Grant No. 1/0348/14. This support is very gratefully acknowledged.

References

1. Alwi, H., Edwards, C., Tan, C.P.: Fault Detection and Fault-Tolerant Control Using Sliding Modes. Springer, London (2011)
2. Amani, A.M., Afshar, A., Menhaj, M.B.: Fault tolerant networked control systems subject to actuator failure using virtual actuator technique. In: Prep. 18th IFAC World Congress, pp. 5465–5470. Milano, Italy (2011)
3. Blanke, M., Kinnaert, M., Lunze, J., Staroswiecki, M.: Diagnosis and Fault-Tolerant Control. Springer, Berlin (2006)
4. Crusius, C.A.R., Trofino, A.: Sufficient LMI conditions for output feedback control problems. IEEE Trans. Autom. Control **44**(5), 1053–1057 (1999)
5. Krokavec, D., Filasová, A.: Design of fault residual functions for systems stabilized by static output feedback. In: Proceedings of the 2nd International Conference Control and Fault-Tolerant Systems SysTol'13, pp. 596–600. Nice, France (2013)
6. Krokavec, D., Filasová, A., Serbák, V., Liščinský, P.: An enhanced approach to actuator fault estimation design for linear continuous-time systems. J. Physics: Conference Series, 570, ID 072002, 10p (2014)
7. Krokavec, D., Filasová, A., Serbák, V.: FTC Structures with virtual actuators and dynamic output controllers. In: Proceedings of the 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS'15, Paris, France (2015). (in press)
8. Lunze, J., Steffen, T.: Control reconfiguration after actuator failures using disturbance decoupling methods. IEEE Trans. Autom. Control **51**(10), 1590–1601 (2006)
9. Mahmoud, M.S., Xia, Y.: Analysis and Synthesis of Fault-Tolerant Control Systems. Wiley, Chichester (2014)
10. Noura, H., Theilliol, D., Ponsart, J.C., Chamseddine, A.: Fault-Tolerant Control Systems: Design and Practical Applications. Springer, London (2009)
11. Peaucelle, D., Henrion, D., Labit, Y., Taitz, K.: User's Guide for SeDuMi Interface 1.04. LAAS-CNRS, Toulouse (2002)
12. Richter, J.H.: Reconfigurable Control of Nonlinear Dynamical System: A Fault-Hiding Approach. Springer, Berlin (2011)
13. Steffen, T.: Control Reconfiguration of Dynamical Systems. Linear Approaches and Structural Tests. Springer, Berlin (2005)
14. Zhang, Y., Jiang, J.: Bibliographical review on reconfigurable fault-tolerant control systems. In: Proceedings of the 5th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, pp. 265–276. Washington, DC, USA (2003)
15. Zhang, K., Jiang, B., Shi, P.: Observer-Based Fault Estimation and Accomodation for Dynamic Systems. Springer, Berlin (2013)
16. Zolghadri, A., Henry, D., Cieslak, J., Efimov, D., Goupil, P.: Fault Diagnosis and Fault-Tolerant Control and Guidance for Aerospace Vehicles: From Theory to Application. Springer, London (2014)

Design of a Predictive Fault-Tolerant Control for the Battery Assembly Station

Pawel Majdzik, Anna Akielaszek-Witczak and Lothar Seybold

Abstract The paper deals with modeling and fault-tolerant control of a real battery assembly system, which is under implementation at RAFI GmbH Company. For that purpose a unified max-plus algebra and model predictive control framework is introduced. Subsequently, the control strategy is enhanced with the fault-tolerance features that enhance the overall performance of the production system. As a result, a novel predictive fault-tolerant strategy is developed that is applied to the battery assembly system. Finally, the last part of the paper shows an illustrative example, which clearly exhibits the performance of the proposed approach.

Keywords Max-plus algebra · Battery assembly · Model predictive control · Fault-tolerant control

1 Introduction

Nowadays, manufacturing systems proceed towards the agile manufacturing, which increases production demands significantly [7]. These increasing demands require efficient and reliable control strategies that ensure required performance under normal conditions and guarantee that the performance will be minimally degraded in the presence of faults. As demonstrated mainly for continuous systems, Model

P. Majdzik (✉)
Institute of Control and Computation Engineering,
University of Zielona Góra, ul. Podgórna 50, 65-246 Zielona Góra, Poland
e-mail: p.majdzik@issi.uz.zgora.pl

A. Akielaszek-Witczak
The State Higher Vocational School in Glogow,
ul. P. Skargi 5, 67-200 Glogow, Poland
e-mail: a.akielaszek@pwsz.glogow.pl

L. Seybold
RAFI GmbH Co. KG, Ravensburger Straße,
128-134, D-88276 Berg/Ravensburg, Germany
e-mail: lothar.seybold@rafi.de

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_16

Predictive Control (MPC) is able to meet these demands in many practical production system [8, 11, 12]. Indeed, MPC can be employed for structural changes, such as sensor and/or actuator faults [4] and variations in the system parameters (it has special importance in the flexible manufacturing systems [6]). Such a strategy is called Fault-Tolerant Control (FTC) [2, 9, 13, 14], which is moderate combination of fault diagnosis and control [15]. However, most of the existing works treat the FDI and FTC problems separately. Unfortunately, a perfect FDI and fault identification are impossible, and hence, there always is inaccuracy related to this process.

The paper deals with the FTC design and implementation for a battery assembly system, which is described using the discrete event max-plus algebra framework [1, 3]. The investigated battery assembly system is under construction at RAFI company, which is one of the leading electronic manufacturing service provider in Germany. With a total of 2500 employees and manufacturing sites in Germany, Hungary, Italy, the USA and China it is providing its services on a global scale.

Thus, the contribution of the paper is the design and implementation of FTC framework for the battery assembly station, which makes it possible to minimize the energy consumption of the autonomous robots while satisfying all production process-related constraints. The proposed strategy has also an appealing property that it can deal with faults regarding mobile robots as well processing and the transportation.

The paper is organized as follows. Section 2 introduces elementary definitions and concepts. The battery assembly system is carefully described in Sect. 3. Subsequently, Sect. 4 introduces the MPC algorithm along with its implementation details. Section 5 presents the details of the FTC algorithm. The subsequent section presents the performance of the proposed approach.

2 Preliminaries

The main objective of this section is to provide essential definitions and concepts that will be exploited in further deliberations.

Definition 1: A fault is an unpermitted deviation of at least one characteristic performance time of the system from the nominal condition.

Definition 2: A failure is a permanent interruption of the system ability to perform a required mission under specified operating conditions.

After providing elementary definitions, it is possible to explain the main mathematical concepts related to the max-plus formalism as well as the max-plus linear system framework.

2.1 Max-plus Algebra and Max-plus Linear Systems

The (max,+) algebraic structure ($\mathbb{R}_{\max}$,$\oplus$,$\otimes$) is defined as follows:

- $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$, where $\mathbb{R}$ is the field of real numbers
- $\forall a, b \in \mathbb{R}_{\max}, a \oplus b = \max(a, b)$
- $\forall a, b \in \mathbb{R}_{\max}, a \otimes b = a + b$.

The operation $\oplus$ and $\otimes$ are called the max-plus algebraic addition and max-plus algebraic multiplication, respectively. Let $k \in \mathbb{N}$, then the kth max-plus algebraic power of $a \in \mathbb{R}$ is denoted by $a^{\otimes^k}$. For $k > 0$, let us define ε such that $\varepsilon^{\otimes^k} = \varepsilon$ and $\varepsilon^{\otimes^0} = 0$. The rules related to the order of evaluation of max-plus algebraic operators are the same as those of the conventional algebra. Thus, the max-plus algebraic power has the highest priority, while the max-plus algebraic multiplication has higher priority than the max-plus algebraic addition. The main properties are:

$$\begin{array}{l} \forall a \in \mathbb{R}_{\max} : a \oplus \varepsilon = a \text{ and } a \otimes \varepsilon = \varepsilon \\ \forall a \in \mathbb{R}_{\max} : a \otimes e = a \end{array} \tag{1}$$

where $\varepsilon = -\infty$ and $e = 0$ are the neutral elements for the max-plus-algebraic addition and max-plus-algebraic multiplication operations respectively.
For matrices $X, Y \in \mathbb{R}_{\max}^{m \times n}$ and $Z \in \mathbb{R}_{\max}^{n \times p}$

$$\begin{array}{l} (X \oplus Y)_{ij} = x_{ij} \oplus y_{ij} = \max(x_{ij}, y_{ij}) \\ (X \otimes Z)_{ij} = \bigoplus_{k=1}^{n} x_{ik} \otimes z_{kj} = \max_{k=1,\ldots,n}(x_{ik} + z_{kj}) \end{array} \tag{2}$$

for all i, j. The matrix E_n is the $n \times n$ max-plus algebraic identity matrix—$(E_n)_{ii} = 0$ and $(E_n)_{ij} = \varepsilon$ for $i \neq j, i, j = 1, \ldots, n$. Thus, the matrix power of $A \in R_{\max}^{m \times n}$ is defined as follows:

$$A^{\otimes^0} = E_n \quad \text{and} \quad A^{\otimes^k} = A \otimes A^{\otimes^{k-1}}, \text{for} \quad k = 1, 2, 3, \ldots$$

Further definitions and details related to the max-plus algebra formalism can be found in [1, 3].

If the max-plus algebra framework is provided, then it is possible to introduce discrete-event systems that can be described by a model in the following form:

$$x(k+1) = A \otimes x(k) \oplus B \otimes u(k) \tag{3}$$

$$y(k) = C \otimes x(k) \tag{4}$$

where the index k is the event counter, while:

- $x(k) \in \mathbb{R}^n_{\max}$ represents the state typically containing the time instants at which the internal events occur for the kth time
- $u(k) \in \mathbb{R}^r_{\max}$ is the input vector containing the time instants at which the input events occur for the kth time
- $y(k) \in \mathbb{R}^m_{\max}$ states for the output vector containing the time instants at which the output events occur for the kth time

and the system matrices are $A \in \mathbb{R}^{n\times n}_{\max}$, $B \in \mathbb{R}^{n\times r}_{\max}$, and $C \in \mathbb{R}^{m\times n}_{\max}$.

3 Battery Assembly Station

Right now the RAFI Company is assembling a small number of battery systems, which are realized by hand mostly. Further regulations and predicted numbers of high performance batteries force change of this procedure in future [10]. Therefore, a flexible battery assembly system with autonomous robots will be introduced for high volume serial production. This new production system is based on transport and manipulation robots, with additional hand assembly stations. The RAFI company goals are to set up a battery assembly system providing a maximal flexibility for upcoming variants of further battery system products and a maximum quality level and protection for product and staff [5].

At the first stage of expansion, only the transport robots are implemented. The actual product of assembly is a high performance battery system for domestic and private usage to buffer renewable energy sources and provide independent energy supply. An overview of the components of the battery system is shown in Fig. 1.

To date, two different main systems with two different voltage ratings are built. The two main formats are a rack based and box based system. These systems are built by either a rack or box housing, two battery modules and a main battery management system. For the rack based system two voltage ratings (1000 and 400 V) are available. Due to the space constraints, let us consider the battery assembly system depicted in Fig. 2. The processing times and transportation times are defined in Fig. 2. To shorten the description of the battery system it has been defined the sequences of transportation times, where $t_{i,j}$ is the sequence of times between t_i and t_j time (including t_i and t_j times). For example the notation $t_{1,7}$ denotes the following sum: $t_1 + t_2 + t_4 + t_6 + t_7$ (Fig. 2).

It is defined that:

- $u_i(k)$ denotes the time instant at which ith robot reaches the individual assembly station
- $x_i(k)$ denotes the time instant at which the ith processing unit starts performing a desired task
- $y(k)$ stands for the time of delivering the final product.

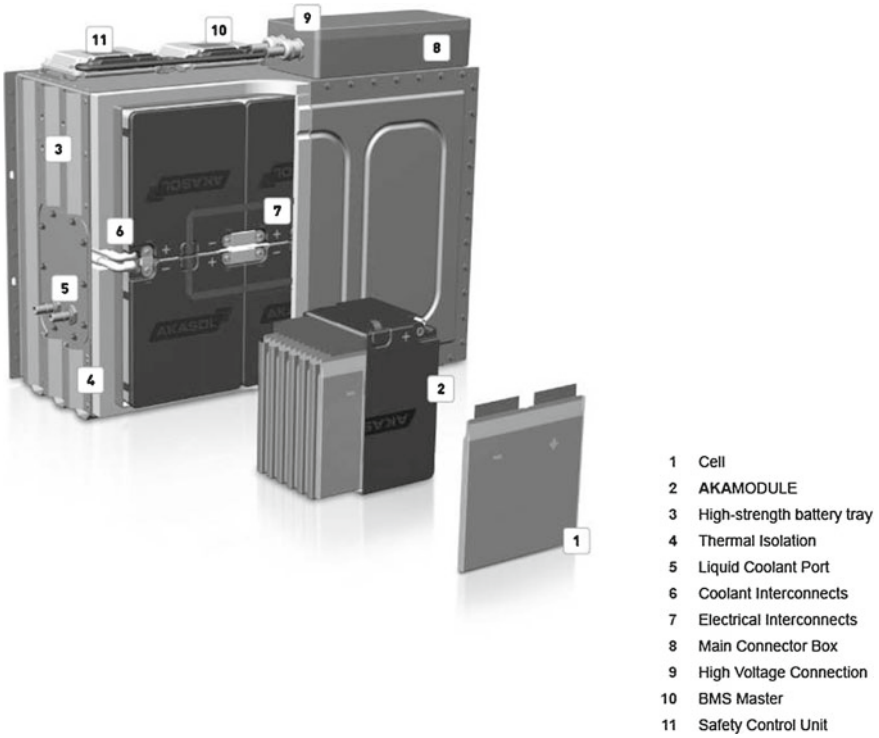


Fig. 1 Overview of the battery assembly system

Note that a processing unit starts performing its operation on a new product (battery) if it has finished performing the previous one. Given a general system structure it is possible to develop its max-plus linear model.

3.1 Max-plus Linear Model

As it was already mentioned, the max plus linear model has the following form:

$$x(k+1) = A \otimes x(k) \oplus B \otimes u(k) \tag{5}$$

$$y(k) = C \otimes x(k) \tag{6}$$

Using the max plus modeling strategy (cf. [6] for a comprehensive explanation), the model matrices depicted in (7) were obtained.

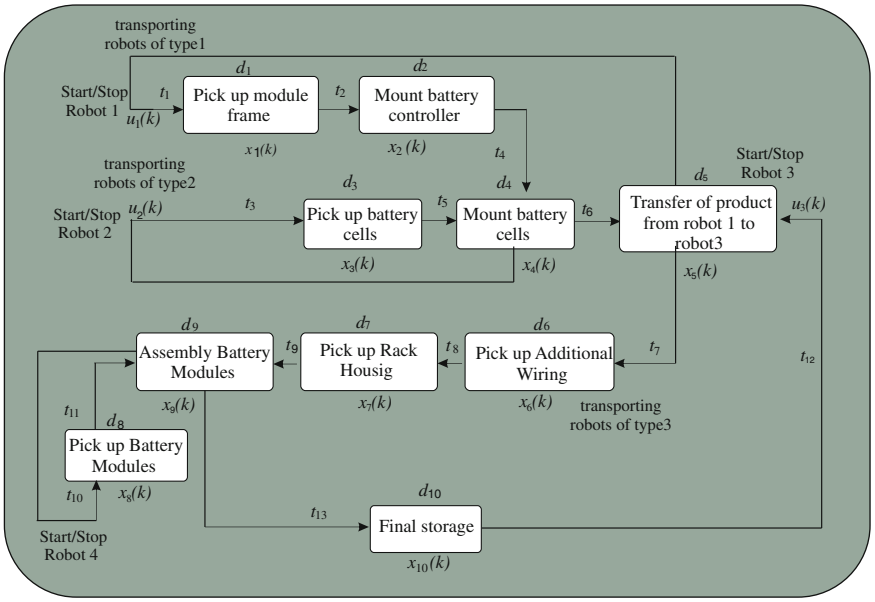


Fig. 2 Details of the assembly process

$$A = \begin{bmatrix} d_1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 2d_1 + t_3 & d_2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & d_3 & \varepsilon & \varepsilon \\ 2d_1 + t_3 + d_2 + t_4 & 2d_2 + t_4 & 2d_3 + t_5 & d_4 & \varepsilon \\ 2d_1 + t_3 + d_2 + t_4 + d_4 + t_6 & 2d_2 + t_4 + d_4 + t_6 & 2d_3 + t_5 + d_4 + t_6 & 2d_4 + t_6 & d_5 \end{bmatrix}$$

$$B = \begin{bmatrix} t_1 & \varepsilon \\ t_1 + d_1 + t_3 & \varepsilon \\ \varepsilon & t_2 \\ t_1 + d_1 + t_3 + d_2 + t_4 & t_2 + d_3 + t_5 \\ t_1 + d_1 + t_3 + d_2 + t_4 + d_4 + t_6 & t_2 + d_3 + t_5 + d_4 + t_6 \end{bmatrix}$$

$$C = [\varepsilon, \ \varepsilon, \ \varepsilon, \ \varepsilon, \ d_5].$$

Since the analytical description of the system is given, it is possible to introduce the system constraints.

3.2 Handling Constraints

The system constraints are as follows:

- first of all, the designed system has to follow some predefined time trajectory that can be defined as a scheduling constraints of the form:

$$x_j(k) \leq t_{ref,j}(k), \quad j = 1, \ldots, n \tag{7}$$

where $t_{ref,j}(k)$ is the upper bound of $x_j(k)$ at time k.

- the second constraint is related with the robot performance,

$$\underline{u}_i \leq u_i(k) \leq \bar{u}_i, \quad i = 1, \ldots, r \tag{8}$$

The lower bound $\underline{u}_i$ corresponds to the maximum speed of the robot. The upper bound $\bar{u}_i$ corresponds to the minimum speed of the robot. Crossing this limit means that its energy consumption rises drastically.

- the last constraint concerns the change rate:

$$u_j(k+1) - u_j(k) \geq z_j, \quad j = 1, \ldots, r \tag{9}$$

where $z_j > 0$ is upper bound of the change rate.

Since the system is described within the max-plus algebra along with suitable constraints, it is possible to develop a control strategy that will enable its optimal performance.

4 Constrained Model Predictive Control

Independently of the system type (continuous or discrete) constraints and control quality measures are inevitable in modern industrial systems. As it was mentioned in the introductory part of this paper, MPC is a perfect candidate to settle this challenging problem. Indeed, one of the core advantages of MPC is its natural ability of handling constraints. The proposed framework is based on the general idea of MPC for max-plus linear systems described in [6]. Note that, according to *Definition 1*, a violation of a scheduling constraint (7) means a faulty behaviour of the system, which will be analyzed within the subsequent sections of this paper. In this section, it is assumed that the system is fault-free, and hence, all imposed constraints (7)–(9) can be satisfied.

Thus, within the proposed framework, MPC along with max-plus algebra are to be used to minimize the robot's energy consumption. This can be perceived as a kind of economic MPC for which the energy consumption is the most important goal. Finally, the problem boils down to finding the input sequence $u(k), \ldots, u(k+N_p-1)$ that minimizes the cost function $J(u)$

$$J(u) = -\sum_{j=0}^{N_p-1} \sum_{i=1}^{r} q_i u_i(k+j) \tag{10}$$

where $q_i > 0$, $i = 1, \ldots, r$ is a positive weighting constant corresponding to the relative importance of the energy consumption of ith robot, while N_p stands for the prediction horizon. The main advantage of (10) over the quadratic criteria employed in the case of continuous systems is that there is no need for using the relatively

time consuming quadratic programming. Instead, taking into account the linear constraints (7)–(9), an efficient linear programming framework can be used. The first task towards the computational framework is to eliminate a direct influence of $x(k+1), \ldots, x(k+N_p-1)$ to the scheduling constraints (7). For this purpose, let:

$$\tilde{x}(k+N_p-1) = M \otimes x(k) \oplus H \otimes \tilde{u}(k) \tag{11}$$

where

$$\begin{aligned} \tilde{u}(k) &= \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+N_p-1) \end{bmatrix} \\ \tilde{x}(k+N_p-1) &= \begin{bmatrix} x(k+1) \\ \vdots \\ x(k+N_p-1) \end{bmatrix} \end{aligned} \tag{12}$$

Using (3), it can be shown that:

$$H = \begin{bmatrix} B & \varepsilon & \cdots & \varepsilon \\ A \otimes B & B & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ A^{\otimes k+N_p-2} \otimes B & A^{\otimes k+N_p-3} \otimes B & \cdots & B \end{bmatrix}$$

$$M = \begin{bmatrix} A \\ A^{\otimes 2} \\ \vdots \\ A^{\otimes k+N_p-1} \end{bmatrix}$$

Thus, substituting (11) into scheduling constraints (7) allows to formulate a linear optimization problem of the form:
Given an initial condition $x(k)$, obtain the optimal input sequence $\tilde{u}(k)^*$ by solving:

$$\tilde{u}(k)^* = \arg\min_{\tilde{u}(k)} J(u) \tag{13}$$

under the constraints (7)–(9).
To summarize, the control algorithm has the structure outlined by Algorithm 1.

Algorithm 1. *Max-plus MPC*

Step 0. Set $k = 0$.
Step 1. Measure the state $x(k)$ and obtain $\tilde{u}(k)^*$ by solving the constrained optimization problem (13).
Step 2. Use the first vector element of $\tilde{u}(k)^*$ (i.e., $u(k)^*$) and feed it into the system (3)–(4).
Step 3. Set $k = k+1$ and go to *Step 1*.

5 Fault-Tolerant Control of the Battery Assembly Station

The main objective of this section is to provide tools that are useful while handling the faults that can appear in the battery assembly station. These fault are divided into two groups:

- mobile robot faults
- process faults.

Fig. 3 provides an outline of the FTC system regarding the above mentioned faults. In particular, the fault diagnosis block is responsible for providing an information about processing and transportation time, whose actual values are fed to this block. When the fault is indicated, then system matrices are suitably recalculated. Finally, the new system has the following form:

$$x(k+1) = A_f \otimes x(k) \oplus B_f \otimes u(k) \tag{14}$$

$$y(k) = C_f \otimes x(k) \tag{15}$$

In case of production and/or transportation faults, the recalculation is very straightforward and requires updating suitable d_i and/or t_i in matrices depicted in Fig. 3, which are provided by Manufacturing Execution System (MES). In case of a mobile robot fault, MES provides also the actual time at which the robot reaches individual assembly station that is denoted by $u_f(k)$. Note that the faulty behavior should be perceived as a delayed reaction of the robot comparing to the calculated transportation time $u(i,k)^*$. Having this information, the FTC system decides about the faulty or fault-free status of the robots, which is realized by a simple residual-based decision threshold:

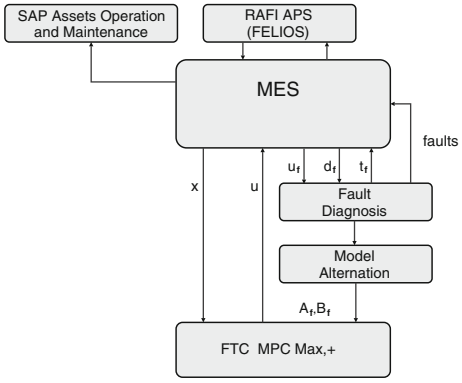


Fig. 3 Detailed FTC MPC Max+

$$\text{if } s_i > \delta_i, \text{ then the } i\text{th mobile robot is faulty} \tag{16}$$

where the residual is:

$$s_i = u_f(i,k) - u(i,k)^* \tag{17}$$

for all $i = 1, \ldots, r$ and $\delta_i > 0$ being a small positive constant that is robot-depended and should be set by the designer. If a mobile robot fault is detected then the matrix B should be replaced by B_f, which in case of an ith fault is defined as:

$$b_{f,j,i} = b_{j,i} \otimes s_i, \quad j = 1, \ldots, m \tag{18}$$

Moreover, it is evident that the faults may have an influence on the scheduling constraints. Indeed, the faulty behavior may cause that the optimization problem (13) will be infeasible. Thus, it is proposed to relax the scheduling constraints as follows:

$$x_j(k) \leq t_{ref,j}(k) + \alpha_j, \quad j = 1, \ldots, n \tag{19}$$

where $\alpha_j \geq 0, j = 1, \ldots, n$ should be as small as possible in order to exhibit a minor divergence from the desired time schedule. In order to obtain the optimal values of α_j, a new cost function is proposed:

$$J(\alpha) = \sum_{i=1}^{n} \alpha_i \tag{20}$$

and hence, a new optimization framework can be described:

$$J(u,\alpha) = (1-\beta)J(u) + \beta J(\alpha) \tag{21}$$

where $1 \leq \beta \leq 0$ is a constant set by the designer, which can be adjusted to reflect a higher importance of either $J(u)$ or $J(\alpha)$, respectively.

Given an initial condition $x(k)$, obtain the optimal input sequence $\tilde{u}(k)^*$ by solving:

$$\tilde{u}(k)^* = \arg\min_{\tilde{u}(k),\alpha} J(u,\alpha) \tag{22}$$

for the faulty system (14)–(15) under constraints (19), (8) and (9). The structure of the proposed strategy is outlined by Algorithm 2.

Algorithm 2. *Predictive FTC*

Step 0. Set k=0.

Step 1. Measure the state $x(k)$ and the actual production and transportation times $\mathbb{P} = t_1, \ldots, t_{n_t}, d_1, \ldots, d_{n_d}$ and then calculate the residual:

$$s_i = p_i - p_{f,i}, i = 1, \ldots, n_t + n_d \tag{23}$$

where $p_i \in \mathbb{P}$ stands for the nominal production/transportation time.

Step 2. If the fault tests (16) or

$$\text{if } s_i > \delta_i, \text{ then } i\text{thsystem production component is faulty} \tag{24}$$

indicate that there is no fault then obtain $\tilde{u}(k)^*$ by solving the constrained optimization problem (13) else obtain $\tilde{u}(k)^*$ by solving (22) with (14)–(15),

Step 3. Use the first vector element of $\tilde{u}(k)^*$ (i.e., $u(k)^*$) and feed it into the system.

Step 4. Set $k = k + 1$ and go to *Step 1*.

6 Illustrative Example

For the illustration purpose and due to the space limitations, let us consider a simplified battery assembly model described by first 5 state variables only, i.e. $x_1, \ldots, x_5$. Moreover, the following fault scenario is considered:

- starting from $k = 4$ there is a delay in production time d_2, which is equal to 3 min
- starting from $k = 6$ there is a second mobile robot fault expressed in 5 min delay
- the system has to fulfill the nominal schedule

$$\begin{aligned} t_{ref}(0) &= [4, 12, 4, 19, 31]^T \\ t_{ref}(1) &= [14, 22, 14, 29, 41]^T \\ t_{ref}(2) &= [24, 32, 24, 39, 51]^T \\ &\vdots \end{aligned} \tag{25}$$

Additionally, the robot performance constraint is neglected while the rate of constraint change (9) is defined as $z_1 = 5$ and $z_1 = 6$. Moreover, the prediction horizon was set to $N_p = 4$ along with $q_1 = q_2 = 1$ shaping the cost function.

The objective of further study is to compare the behaviour of *Algorithm 2* with its modified version without fault-tolerance, i.e. with deactivated fault diagnosis and fault compensation. Figure 4 shows the difference between the actual state and the reference trajectory with FTC and without it. It can be observed that the FTC strategy brings the system to the nominal performance in just two cycles (event counter) while the strategy without FTC causes a permanent delay.

It should be notated that the realization cost of MPC is $J = -1100$ while for the predictive FTC it is $J = -1054$. It can be concluded that the additional cost of FTC

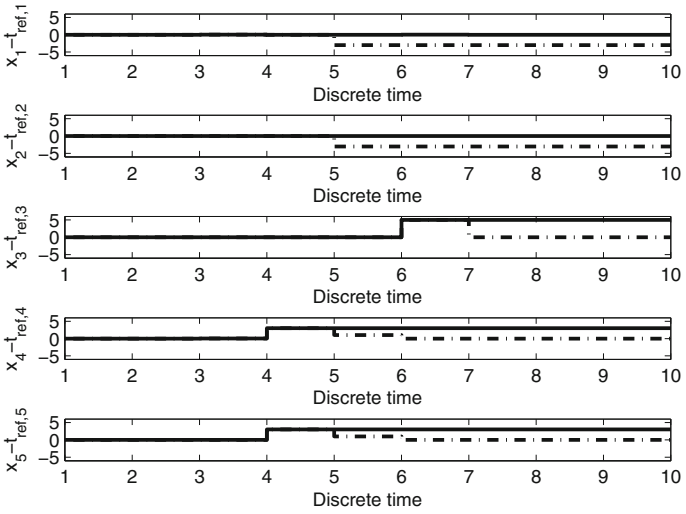


Fig. 4 Difference between actual state and the reference trajectory with FTC (*dashed line*) and without it

is fully justified and exhibits with a high performance of the system, which results in a faster product outcome comparing to the situation without FTC.

7 Conclusions

The main objective of this paper was to propose a unified FTC MPC design procedure for the battery assembly system located at the RAFI Company, allowing a high production flexibility under real production conditions. In particular, one of the objectives was to describe the system within the max-plus algebra framework along with the suitable constraints inevitably present in all real systems. The main advantage of the proposed approach is the avoidance of the non-linear optimization problem, which is the main drawback of the classical algebra framework applied to such tasks. Apart from a suitable modeling framework, the FTC MPC-based control procedure was provided. Indeed, a suitable control criterion and constraints were provided. The main advantage comparing to the classical framework is that the cost function is linear but not quadratic. This significantly reduces the computational burden. Because of lack of space, the presented experimental results concern a selected part of the entire system but they clearly confirm its high performance.

Acknowledgments The work was supported by the National Science Center of Poland under grant: 2014-2017.

References

1. Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.P.: Synchronization and linearity: an algebra for discrete event systems. J. Oper. Res. Soc. **45**, 118–118 (1994)
2. Blanke, M., Schröder, J., Kinnaert, M., Lunze, J., Staroswiecki, M.: Diagnosis and Fault-tolerant Control. Springer, Berlin (2006)
3. Butkovic, P.: Max-linear Systems: Theory and Algorithms. Springer, London (2010)
4. Camacho, E.F., Bordons, C.A.: Model Predictive Control in the Process Industry. Springer, New York (1997)
5. Chan, C.: The state of the art of electric and hybrid vehicles. Proc. IEEE **90**(2), 247–275 (2002)
6. De Schutter, B., Van Den Boom, T.: Model predictive control for max-plus-linear discrete event systems. Automatica **37**(7), 1049–1056 (2001)
7. Gunasekaran, A.: Agile manufacturing: a framework for research and development. Int. J. Prod. Econ. **62**(1), 87–105 (1999)
8. Mrugalska, B., Akielaszek-Witczak, A., Stetter, R.: Robust quality control of products with experimental design. In: Popescu, D. (ed.) 2014 International Conference on Production Research—Regional Conference Africa, Europe and the Middle East and 3rd International Conference on Quality and Innovation in Engineering and Management, pp. 343–348 (2014)
9. Mrugalski, M.: An unscented kalman filter in designing dynamic gmdh neural networks for robust fault detection. Int. J. Appl. Math. Comput. Sci. **23**(1), 157–169 (2013)
10. Nair, N.K.C., Garimella, N.: Battery energy storage systems: assessment for small-scale renewable energy integration. Energy Build. **42**(11), 2124–2130 (2010)
11. Prodan, I., Olaru, S., Stoica, C., Niculescu, S.I.: Predictive control for trajectory tracking and decentralized navigation of multi-agent formations. Int. J. Appl. Math. Comput. Sci. **23**(1), 91–102 (2013)
12. Rossiter, J.: Model-based Predictive Control: A Practical Approach. CRC Press, Boca Raton (2013)
13. Witczak, M., Puig, V., Oca, S.D.: A fault-tolerant control strategy for non-linear discrete-time systems: application to the twin-rotor system. Int. J. Control **86**(10), 1788–1799 (2013)
14. Witczak, M.: Fault Diagnosis and Fault-tolerant Control Strategies for Non-linear Systems. Lecture Notes in Electrical Engineering, vol. 266. Springer International Publishing, Heidelberg (2014)
15. Zhang, Y., Jiang, J.: Bibliographical review on reconfigurable fault-tolerant control systems. Annu. Rev. Control **32**(2), 229–252 (2008)

Part IV
Industrial and Medical Diagnostics

Approximate Models and Parameter Analysis of the Flow Process in Transmission Pipelines

Zdzisław Kowalczuk and Marek Tatara

Abstract Basically, the paper deals with the problem of early leak detection in transmission pipelines. First we present the derivation of state-space equations of the flow process in the pipelines. This description is then aggregated in order to obtain a principal model. Next, the problem of process model parametrization is addressed, taking into account the maximization of a model stability margin. The location of the maximum is determined using optimization methods and curve fitting techniques. In such a way an optimal process parametrization is obtained. A simplified state-space model is then derived based on diagonal approximation, referred to as the analytic model (AMDA). Finally, the useful properties of the developed model are analyzed, including the speed and accuracy of an applied inverse matrix.

Keywords Flow processes · Modeling · Leak detection and identification · Transmission pipeline diagnostics

1 Introduction

Transport through transmission pipelines is an efficient way of delivering gases and fluids over long distances. Nevertheless, there is always a risk of leakage, which exposes an owner of the pipeline transport to financial losses, environment to pollution and nearby humans to danger. Such reasons make the problem of early and accurate leak detection important. Thus, the Leak Detection and Identification systems (LDI) are expected to assure maximum sensitivity, accuracy, reliability and robustness.

Z. Kowalczuk (✉) · M. Tatara
Faculty of Electronics, Telecommunications and Informatics, Department of Robotics and Decision Systems, Gdańsk University of Technology, Gdańsk, Poland
e-mail: kova@eti.pg.gda.pl

M. Tatara
e-mail: martatar@eti.pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_17

Over the years, many kinds of LDI systems were proposed. In 1987 a model-based diagnostic system was introduced by Billmann and Isermann [2], where two partial differential equations, derived from the laws of conservation of momentum and mass, were discretized using the central difference method and then arranged into a state-space model. Such a model can be used to obtain estimates of the mass flow and the pressure along the diagnosed pipeline. These estimates when compared to measured data can generate residual signals. Analysis performed on the residuals leads to the estimation of the leak location and size. The model has many applications and further extensions, like the ones by Kowalczuk and Gunawickrama [6–8] concerning on-line friction factor estimation, a model-based cross-correlation method (compensating modeling errors), and others. An approach based on the identification of parameters, utilizing an Extended Kalman Filter can also be found in [12].

The basic analytical tools cover only the case of simple pipelines (i.e. without branches). Another model was recently proposed in [10] that allows to diagnose more complex pipeline systems, at the cost of a higher number of sensors placed along the pipeline. It is thus clear that diagnostic systems for pipeline networks is an important target of research.

In this paper we analyze the flow process for the purpose of obtaining a new state-space model efficient from the computation viewpoint.

2 State Space Model

Our model derivation starts with rearrangement of the mass and the momentum conservation law. Using the assumption of isothermal flow, the following set of equations is obtained [2]:

$$\frac{S}{\nu^2}\frac{\partial p}{\partial t} + \frac{\partial q}{\partial z} = 0 \tag{1}$$

$$\frac{1}{S}\frac{\partial q}{\partial t} + \frac{\partial p}{\partial z} = -\frac{\lambda \nu^2}{2DS^2}\frac{q|q|}{p} - \frac{gsin\alpha}{\nu^2}p \tag{2}$$

where S is the cross-sectional area $[m^2]$, ν is the isothermal velocity of sound in the fluid $[\frac{m}{s}]$, D is the diameter of the pipe $[m]$, q is the mass flow rate $[\frac{kg}{s}]$, p is the pressure $[Pa]$, t is a time co-ordinate $[s]$, z is a spatial coordinate $[m]$, λ is the generalized friction factor [-], α is the pipeleg inclination angle $[rad]$.

After discretization based on the following central difference method

$$\frac{\partial x}{\partial t} = \frac{3x_z^{k+1} - 4x_z^k + x_z^{k-1}}{2\Delta t} \tag{3}$$

$$\frac{\partial x}{\partial z} = \frac{x_{z+1}^{k+1} - x_{z-1}^{k+1} + x_{z+1}^k - x_{z-1}^k}{4\Delta z} \tag{4}$$

and further rearrangements, the following discrete-time singular system, consisting of an even number (N) of the segments of the pipeline, is obtained:

$$\mathbb{A}\hat{\mathbf{x}}^k = \mathbb{B}\hat{\mathbf{x}}^{k-2} + \mathbb{C}\left(\hat{\mathbf{x}}^{k-1}\right)\hat{\mathbf{x}}^{k-1} + \mathbb{D}\mathbf{u}^{k-1} + \mathbb{E}\mathbf{u}^k \tag{5}$$

where $\hat{\mathbf{x}}^k = \left[\hat{q}_0^k\ \hat{q}_2^k\ \hat{q}_4^k \cdots \hat{q}_N^k\ \hat{p}_1^k\ \hat{p}_3^k\ \hat{p}_5^k \cdots \hat{p}_{N-1}^k\right]^T \in \mathbb{R}^{N+1}$ is a state vector at a time instant k, and $\mathbf{u}^k = \left[p_0^k\ p_N^k\right]^T \in \mathbb{R}^2$ is an input vector at that instant k. The caret symbols stand for estimates. The matrices $\mathbb{B}$, $\mathbb{C}$, $\mathbb{D}$ and $\mathbb{E}$ are deciphered in [6]. The most important matrix $\mathbb{A}$, also referred to as the recombination matrix, which is the subject of our further analysis in this paper, can be constructed as follows:

$$\mathbb{A} = \begin{bmatrix} \mathbb{A}_1 & \mathbb{A}_2 \\ \mathbb{A}_3 & \mathbb{A}_4 \end{bmatrix} = \left[\begin{array}{cccc:ccccc} & & & & 2b & 0 & \cdots & 0 & 0 \\ & & & & -b & b & \cdots & 0 & 0 \\ & & D_{\frac{N}{2}+1}(c) & & \vdots & & \ddots & & \vdots \\ & & & & 0 & 0 & \cdots & -b & b \\ & & & & 0 & 0 & \cdots & 0 & -2b \\ \hdashline -b & b & 0 & \cdots\ 0 & & & & & \\ 0 & -b & b & \cdots\ 0 & & & & & \\ \vdots & & \ddots & \vdots & & & D_{\frac{N}{2}}(a) & & \\ 0 & \cdots & -b & b\ 0 & & & & & \\ 0 & \cdots & 0 & -b\ b & & & & & \end{array}\right] \tag{6}$$

where $D_W(x)$ is a $W \times W$ diagonal matrix with the argument x in all its main diagonal elements, $a = \frac{3S}{2\nu^2\Delta t}$, $b = \frac{1}{4\Delta z}$, and $c = \frac{3}{2S\Delta t}$ are physical coefficients.

Based on the fact that the recombination matrix is always invertible [9], the model (5) can be represented in the following nonsingular form:

$$\hat{\mathbf{x}}^k = \mathbb{A}^{-1}\mathbb{B}\hat{\mathbf{x}}^{k-2} + \mathbb{C}\left(\hat{\mathbf{x}}^{k-1}\right)\hat{\mathbf{x}}^{k-1} + \mathbb{D}\mathbf{u}^{k-1} + \mathbb{E}\mathbf{u}^k \tag{7}$$

By aggregating the two state vectors $\tilde{\mathbf{x}}^k = \left[\hat{\mathbf{x}}^k\ \hat{\mathbf{x}}^{k-1}\right]^T$ and the two input vectors $\tilde{\mathbf{u}}^k = \left[\mathbf{u}^k\ \mathbf{u}^{k-1}\right]^T$, the above model can be further simplified [9]. With the use of the Matrix Inversion Lemma (MIL) [3]:

$$\mathbb{A}^{-1} = \begin{bmatrix} \mathbb{A}'_1 & \mathbb{A}'_2 \\ \mathbb{A}'_3 & \mathbb{A}'_4 \end{bmatrix} = \begin{bmatrix} (\mathbb{A}_1 - \mathbb{A}_2\mathbb{A}_4^{-1}\mathbb{A}_3)^{-1} & -\mathbb{A}_1^{-1}\mathbb{A}_2(\mathbb{A}_4 - \mathbb{A}_3\mathbb{A}_1^{-1}\mathbb{A}_2)^{-1} \\ -\mathbb{A}_4^{-1}\mathbb{A}_3(\mathbb{A}_1 - \mathbb{A}_2\mathbb{A}_4^{-1}\mathbb{A}_3)^{-1} & (\mathbb{A}_4 - \mathbb{A}_3\mathbb{A}_1^{-1}\mathbb{A}_2)^{-1} \end{bmatrix} \tag{8}$$

one obtains an integrated/aggregated form of the dynamical state-space equation describing the flow process, referred to as the principal model

$$\tilde{\mathbf{x}}^k = \mathbb{A}_c\tilde{\mathbf{x}}^{k-1} + \mathbb{B}_c\tilde{\mathbf{u}}^k \tag{9}$$

where

$$\mathbb{A}_c = \begin{bmatrix} \mathbb{A}^{-1}\mathbb{C} & \mathbb{A}^{-1}\mathbb{B} \\ \mathbb{I} & \mathbf{0} \end{bmatrix} \tag{10}$$

$$\mathbb{B}_c = \begin{bmatrix} \mathbb{A}^{-1}\mathbb{E} & \mathbb{A}^{-1}\mathbb{D} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{11}$$

It should be noted that $\mathbb{A}_c$ is a function of the state vector $\tilde{\mathbf{x}}^{k-1}$ and $\mathbb{B}_c$ depends on the friction factor λ.

The system (9) is nonsingular and can be analyzed using control theory methods such as stability analysis. However, to make a better use of the stability analysis, the recombination matrix has to be inverted first.

3 Model Parameterization

When approximating the differential equations with the above difference schemes, the problem of proper parameterization needs to be addressed. Due to the nature of the problem, first of all you should take care of the stability of the numerical solution with respect to the time and spatial increments. The chosen values should also ensure possibility of on-line simulation of the process.

The Courant-Friedrichs-Lewy condition (CFL) says [11] that in order to maintain stability, the speed of propagating information $\frac{\Delta z}{\Delta t}$ in explicit numerical methods must take precedence over the wave speed of the simulated process. Since the highest physical speed in the pipeline system is the velocity of sound (ν) in fluid, the following inequality has to be satisfied:

$$\frac{\Delta z}{\Delta t} \geqslant \nu \tag{12}$$

which is a necessary, but not sufficient condition for the stability of the model. Taking this into consideration, the condition (12) can be rewritten as an equality

$$\Delta t = \mu \frac{\Delta z}{\nu} \tag{13}$$

where μ is a coefficient selected within the range $(0, 1\rangle$.

Research unveils that for the specific physical parameters of pipe flow, there exists a value μ_{opt} assuring a maximum stability margin s_m for the discrete-time system (on the z-plane), calculated as:

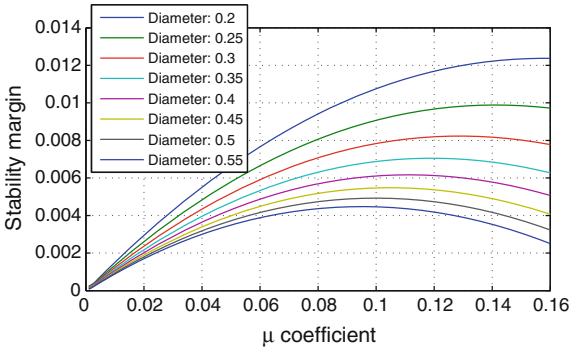


Fig. 1 Distribution of the stability margin versus the coefficient μ for 8 different diameters of the pipeline (the other parameters of the pipeline: $N = 10, L = 4000\,\text{m}, \lambda = 0.01, \nu = 304\frac{m}{s}, p_{inlet} = 3.2\,\text{MPa}, p_{outlet} = 3.0\,\text{MPa}$)

$$s_m = 1 - eig_{max} \tag{14}$$

where eig_{max} is the maximum eigenvalue of the state transition matrix. Exemplary results concerning the stability margin versus μ are shown in Fig. 1.

To avoid the impact of transitional processes, the eigenvalues of the state transition matrix have been calculated after 20,000 simulation iterations.

Since the maximal stability margin has the order of magnitude 10^{-2}, the problem of proper parameterization is even more important. This is because systems with such small margins can be easily destabilized by improperly selected simulation parameters. The influence of each of the physical parameters characterizing the pipeline flow on the margin s_m have been examined with the goal of obtaining an analytic function of μ_{opt} in terms of these parameters.

The Hooke-Jeeves algorithm has been implemented to numerically optimizing μ_{opt} for each physical parameter. 10 'measurements' of the optimal relation were gathered, and next the MATLAB toolbox cftool (*curve fitting tool*) was used to fit an analytic curve to the experimental data. The following analytic function was applied as a common curve for all the fitting processes:

$$\mu_{opt} = C_1 p_p^{C_2} \tag{15}$$

where C_1 and C_2 are coefficients to be optimized, p_p is one of the physical parameters (N, D, L, λ, p_m—mean pressure in the pipeline and p_d—pressure drop along the pipeleg). The goodness of this fit is determined using the R-squared Coefficient of Determination (the CoD R^2), defined as the ratio of the sum of squared differences between the model values and RMV to the sum of squared differences between experimental data and RMV, where RMV is a Referencing Mean Value, i.e. a mean value of the observed data [13]. The closer the value of R^2 to 1, the better the curve fits the experimental data. The fitting results are presented numerically in Table 1 and graphically in Fig. 2, where consecutive sub-figures present the obtained fitting w.r.t. particular physical parameters.

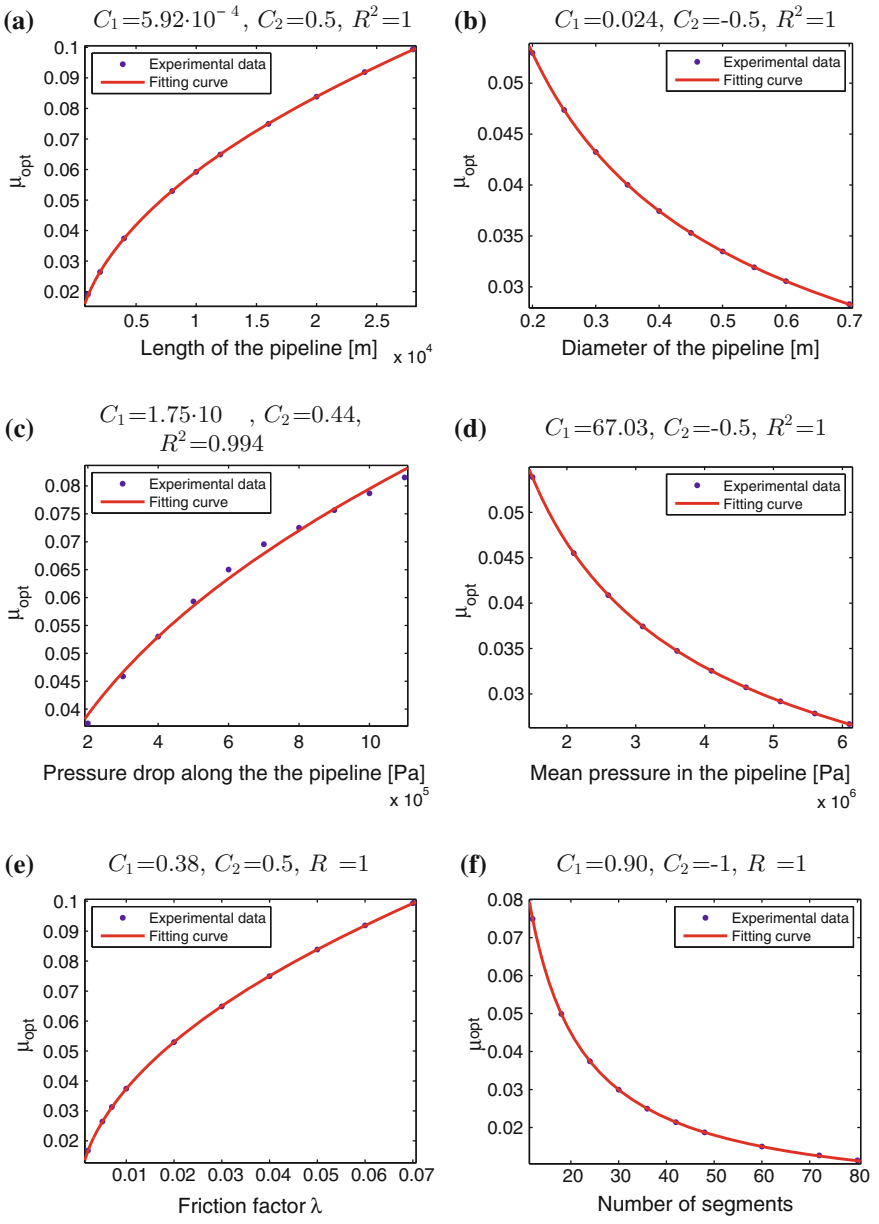


Fig. 2 Plots showing the results of fitting the function (15) for μ_{opt} for different: **a** length of the pipeline; **b** diameter of the pipeleg; **c** difference between the inlet and the outlet pressure; **d** mean pressure in the pipeline; **e** friction factor; **f** number of the pipeline segments

Table 1 The results of fitting (15) to experimental data for each physical parameter (AP—the effective Approximated Power of each physical parameter that results from a functional estimate of C_2)

Physical parameter of the pipeline	C_1	C_2	R^2	AP
Length	$5.92 \cdot 10^{-4}$	0.5	1	0.5
Diameter	0.024	−0.5	1	−0.5
Pressure drop along the pipeleg	$1.75 \cdot 10^{-4}$	0.44	0.994	0.5
Mean pressure	67.03	−0.5	1	−0.5
Friction factor	0.38	0.5	1	0.5
Number of segments	0.90	−1	1	−1

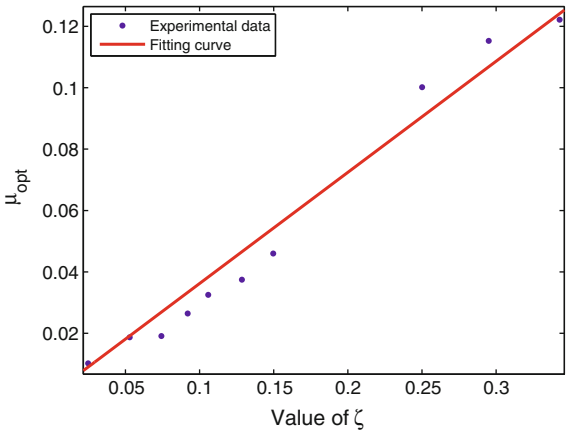


Fig. 3 Result of fitting the experimental data to the model (16)

Let us introduce the following interpretation of the identified parameters: C_2 as the power of the physical parameter and C_1 as the scaling factor. C_2 gives information about an individual impact of the parameter in the model, while the effect of C_1 can be expressed by an aggregate coefficient C_3 as follows

$$\mu_{opt} = C_3 \xi = C_3 \sqrt{\frac{p_d L \lambda}{p_m d}} \frac{1}{N} \tag{16}$$

where C_3 is the coefficient to be found. Once again the MATLAB cftool was used to determine C_3 with the fitting equation in the form (16), where 10 experimental points were 'measured' for randomly chosen values of the physical parameters.

The result of this fitting is presented in Fig. 3. The value of $C_3 = 0.362$ was obtained with the CoD $R^2 = 0.972$. The bias seen in the fitting plot may be due to

the applied approximation ($0.44 \approx 0.5$, see Table 1) of the coefficient while fitting the μ_{opt} to the pressure drop along the pipeline. Nevertheless, due to a high value of CoD we can agree that the curve fits the experimental data satisfactory and, as such, can be used to approximately represent the function of μ_{opt}. The final form of the model (16) is then

$$\mu_{opt} = 0.362\sqrt{\frac{p_d L\lambda}{p_m d}}\frac{1}{N} \tag{17}$$

Remark Function (17) is applicable solely for the principal model (9). For other models it need not assure maximum stability margin, nor even stability at all.

4 Analytic Model from Diagonal Approximation

A method for the recombination matrix using the inversion method [5] for tridiagonal matrices has been presented in [9]. For higher order systems, however, the method produces numbers from outside the acceptable computer-representation range, which in turn makes it impossible to analyze the accuracy and the convergence of the numerical methods for practical pipeline segmentation. To avoid this, an analytic method of inversion is proposed.

4.1 Approximate Model Derivation with Applicability Conditions

Due to fact the two submatrices of the inverted recombination matrix, obtained from the application of MIL, are tridiagonal:

$$\mathbb{A}_1' = \begin{bmatrix} c+\frac{2b^2}{a} & -\frac{2b^2}{a} & 0 & \cdots & 0 & 0 & 0 \\ -\frac{b^2}{a} & c+\frac{2b^2}{a} & -\frac{b^2}{a} & \cdots & 0 & 0 & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & 0 & 0 & \cdots & -\frac{b^2}{a} & c+\frac{2b^2}{a} & -\frac{b^2}{a} \\ 0 & 0 & 0 & \cdots & 0 & -\frac{2b^2}{a} & c+\frac{2b^2}{a} \end{bmatrix}^{-1} \tag{18}$$

$$\mathbb{A}_4' = \begin{bmatrix} a+\frac{3b^2}{c} & -\frac{b^2}{c} & 0 & \cdots & 0 & 0 & 0 \\ -\frac{b^2}{c} & a+\frac{2b^2}{c} & -\frac{b^2}{c} & \cdots & 0 & 0 & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & 0 & 0 & \cdots & -\frac{b^2}{c} & a+\frac{2b^2}{c} & -\frac{b^2}{c} \\ 0 & 0 & 0 & \cdots & 0 & -\frac{b^2}{c} & a+\frac{3b^2}{c} \end{bmatrix}^{-1} \tag{19}$$

we propose to approximate them with their diagonal counterparts. To perform such an operation, the values on the main diagonal have to be significantly greater than the ones on the superdiagonal and the subdiagonal. To achieve this, the following condition needs to be satisfied:

$$|c| \gg |2\frac{b^2}{a}| \tag{20}$$

The condition (20) is sufficient for both of the matrices (18) and (19). By taking into account the physical counterparts of the coefficients a, b and c, Eq. (20) can be rewritten as

$$|\frac{3}{2S\Delta t}| \gg |2\frac{1}{16\Delta z^2}\frac{2\nu^2\Delta t}{3S}| \tag{21}$$

Assuming that Δt, Δz and ν are positive, and after rearrangements, we obtain the following relation:

$$\Delta t^2 \ll 18\frac{\Delta z^2}{\nu^2} \tag{22}$$

which can be rewritten as

$$\Delta t \ll \sqrt{18}\frac{\Delta z}{\nu} \tag{23}$$

When comparing the basic CFL condition (12) with (23), one concludes that the following restriction should be followed

$$\mu \ll \sqrt{18} \tag{24}$$

Assuming a respective ratio of two orders of magnitude (at least), we can give it a practical conditioning:

$$\mu < 0.042 \tag{25}$$

By comparing inequality (25) to a Courant-Friedrich-Lewy result discussed in [4], which sets μ to be lower than 0.90–0.95 for stationary flows (depending on flow parameters), it is clear that our result is more restrictive. The difference is due to the fact that the CFL criterion defines solely a necessary condition (used in a task of stabilization of flow equations using Lyapunov functions). Now, when our condition (25) is satisfied, the two submatrices of the inverted recombination matrix can be approximated as

$$\mathbb{A}_1' \approx \breve{\mathbb{A}}_1' = \begin{bmatrix} c+\frac{2b^2}{a} & 0 & 0 \cdots 0 & 0 & 0 \\ 0 & c+\frac{2b^2}{a} & 0 \cdots 0 & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 \cdots 0 & c+\frac{2b^2}{a} & 0 \\ 0 & 0 & 0 \cdots 0 & 0 & c+\frac{2b^2}{a} \end{bmatrix}^{-1} \tag{26}$$

and

$$\mathbb{A}_4' \approx \breve{\mathbb{A}}_4' = \begin{bmatrix} a+\frac{3b^2}{c} & 0 & \cdots\ 0\ 0 & 0 & \\ 0 & a+\frac{2b^2}{c} & 0 \cdots 0 & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 \cdots 0 & a+\frac{2b^2}{c} & 0 \\ 0 & 0 & 0 \cdots 0 & 0 & a+\frac{3b^2}{c} \end{bmatrix}^{-1} \tag{27}$$

It is thus clear that the inversions of the matrices (26) and (27) can simply be obtained by computing the reciprocals of the values on their main diagonal. Therefore, utilizing the MIL on the remaining submatrices, the analytic inversion of the recombination matrix is the following

$$\mathbb{A}^{-1} \approx \left[\begin{array}{ccccc:ccccc}
\frac{a}{\sigma} & 0 & \cdots & 0 & 0 & -\frac{2b}{3b^2+ca} & 0 & \cdots & 0 & 0 \\
0 & \frac{a}{\sigma} & & 0 & 0 & \frac{b}{3b^2+ca} & -\frac{b}{\sigma} & 0 & 0 & 0 \\
 & & & & & 0 & \frac{b}{\sigma} & -\frac{b}{\sigma} & 0 & 0 \\
\vdots & & \ddots & & \vdots & \vdots & & \ddots & & \vdots \\
 & & & & & 0 & 0 & \frac{b}{\sigma} & -\frac{b}{\sigma} & 0 \\
0 & 0 & & \frac{a}{\sigma} & 0 & 0 & 0 & 0 & \frac{b}{\sigma} & -\frac{b}{3b^2+ca} \\
0 & 0 & \cdots & 0 & \frac{a}{\sigma} & 0 & 0 & \cdots & 0 & \frac{2b}{3b^2+ca} \\
\hdashline
\frac{b}{\sigma} & -\frac{b}{\sigma} & 0 \cdots & 0 & 0 & \frac{c}{3b^2+ca} & 0 & \cdots & 0 & 0 \\
0 & \frac{b}{\sigma} & -\frac{b}{\sigma} & 0 & 0 & 0 & \frac{c}{\sigma} & & 0 & 0 \\
0 & 0 & \frac{b}{\sigma} & 0 & 0 & & & & & \\
\vdots & & \ddots & & \vdots & \vdots & & \ddots & & \vdots \\
0 & 0 & -\frac{b}{\sigma} & 0 & 0 & & & & & \\
0 & 0 & \frac{b}{\sigma} & -\frac{b}{\sigma} & 0 & 0 & 0 & & \frac{c}{\sigma} & 0 \\
0 & 0 \cdots & 0 & \frac{b}{\sigma} & -\frac{b}{\sigma} & 0 & 0 & \cdots & 0 & \frac{c}{3b^2+ca}
\end{array}\right] \tag{28}$$

where $\sigma = 2b^2 + ca$.

The above scheme highlights a simple explicit form of the inverted matrix, which should be less time-consuming and more effective for on-line simulations. The model that uses the recombination matrix inverted in this way will be called Analytic Model of Diagonal Approximation (AMDA). This is apparently only an approximated model, thus the accuracy and the speed of this method will be examined in the following subsection.

4.2 Model Analysis

Before the model will be analyzed, we have to consider the influence of the CFL condition (25) on the model (17). Combining these two discoveries we obtain

$$\mu_{opt} = 0.362\sqrt{\frac{p_d L\lambda}{p_m D}}\frac{1}{N} < 0.042 \tag{29}$$

and, next, after rearrangements we derive a formula for the applicable cardinality of the pipeline segments (when using (17) to select μ_{opt} for AMDA):

$$N > 8.62\sqrt{\frac{p_d L\lambda}{p_m D}} \tag{30}$$

Note that μ_{opt} may differ slightly for each model. With the increasing accuracy of the inversion, however, the value of the optimal coefficient for the AMDA model approaches to its 'location' for the principal model. Note also that the segmentation calculated by (30) has to be rounded up the nearest even integer.

As has been mentioned, AMDA is an approximated model. Now it is time to verify the accuracy of the inversion. The matrix was inverted using two methods: the analytic approximation and the optimized MATLAB function inv(). The criterion of comparison has been the Euclidean norm of the difference between the two inverted matrices. The results of the study of the inversion methods versus μ and versus N are presented in Fig. 4. Additionally, the necessary inversion (computation) time versus the parameters μ and N is also shown there.

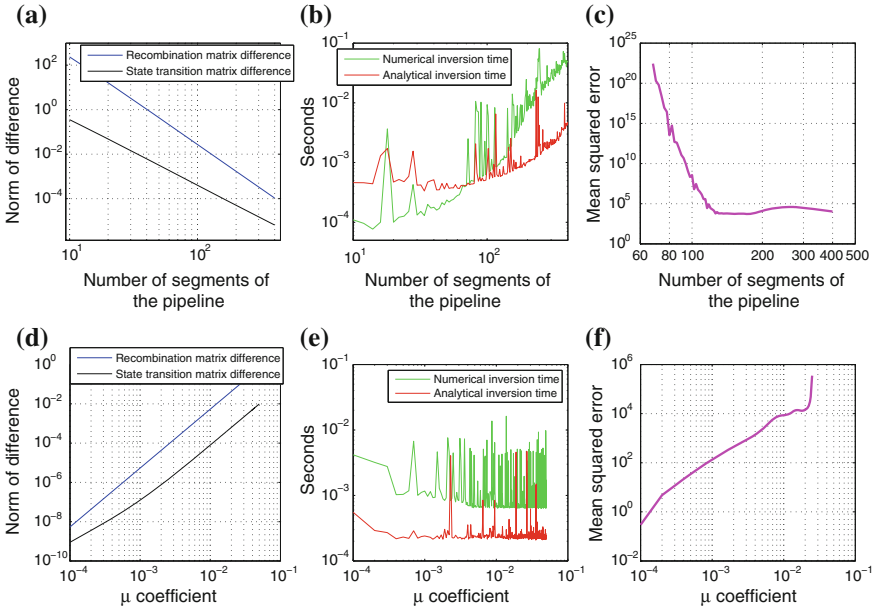


Fig. 4 Analysis of the AMDA inversion scheme as compared with the MATLAB function inv for a pipeleg with $L = 2000\,\text{m}$, $\nu = 304.23\frac{m}{s}$, $D = 0.4\,\text{m}$, $p_{inlet} = 3.2\,\text{MPa}$, $p_{outlet} = 3.0\,\text{MPa}$, $\lambda = 0.1$ in the log-log scale with N ranging from 10 to 400 for subplots (**a**–**c**) and $N = 100$ for subfigures (**d**–**f**); μ was calculated using (17): **a**, **d** *blue line* is the Euclidean norm of the difference between the inverted recombination matrices obtained by AMDA and inv functions; *black line* is the Euclidean norm of the difference between state transition matrices obtained by AMDA and inv functions (**b**, **e**). *Green line* is the numerical inversion time (using the inv function), *red line* is the analytic inversion time (using AMDA) (**c**, **f**) *magenta line* shows mean quadratic difference between mass-flow estimates obtained by the two analyzed models

Figure 4 shows that the inversion time of the recombination matrix is almost independent of μ itself (Fig. 4d) and increases when the number of segments is increased (Fig. 4b). In Fig. 4b one can see that for a greater number of segments, inversion using the AMDA scheme takes less time than inversion using the optimized MATLAB inv function. Moreover, Fig. 4d shows that the lower μ coefficient, the more accurate matrix is obtained by the AMDA model, what is visible in a sense of mean quadratic error between the estimates of mass flow, obtained by the two models (Fig. 4f).

5 Summary

The paper has described a procedure of obtaining a state-space model of pipeline flow processes. The problem of proper parameterization has been addressed. As the research reveals that there exists an optimal parameterization that assures a maximum stability margin, the suitable model (17) for selecting the factor μ has been derived. The resulting value of μ assures a maximal stability margin for the principal model (9) with satisfactory accuracy.

Due to the specific structure of the recombination matrix, two of the submatrices of the inverted matrix has been approximated by diagonal matrices. Such an approximation can be used under the condition (25) (the remaining two submatrices are calculated based on the former two inverted submatrices).

The applied approximations of the inverted recombination matrix and the state transition matrix have been compared with the ones obtained from the numerical inversion applied to the principal model (9). The research has revealed that the lower coefficient μ, the closer to each other are the matrices obtained numerically and analytically in the sense of the Euclidean norm of the difference between them. The derived AMDA model can successfully approximate the principal state-space model for a greater number of the pipeline segments (leading to a more precise model).

In our further research we are going to cover analysis of applicability of the derived model to the LDI systems based on detection observers.

References

1. Belsito, S., Lombardi, P., Andreussi, P., Banerjee, S.: Leak detection in liquefied gas pipelines by artificial neural networks. AIChE **44**(12), 2675–2688 (1998)
2. Billmann, L., Isermann, R.: Leak detection methods for pipelines. Automatica **23**(3), 381–385 (1987)
3. Brogan, W.: Modern Control Theory, 3rd edn. Prentice Hall, New Jersey (1991)
4. Dick, M.: Stabilization of the gas flow in networks: boundary feedback stabilization of quasilinear hyperbolic systems on networks. Ph.D. thesis, Friedrich-Alexander-Universität, Erlangen-Nürnberg (2012)
5. Da Fonseca, C.M., Petronilho, J.: Explicit inverses of some tridiagonal matrices. Linear Algebra Appl. **325**, 7–21 (2001)

6. Gunawickrama, K.: Leak detection methods for transmission pipelines. Ph.D. thesis supervised by Z. Kowalczuk, Gdańsk Univeristy of Technology, Gdańsk (2001)
7. Kowalczuk, Z., Gunawickrama, K.: Detection of leakages in industry pipelines using a cross-correlation approach (in Polish: wykrywanie przecieków w rurociagach przemysłowych metodą korelacyjno-modelową). Pomiary Automatyka Kontrola **44**(4), 140–146 (1998)
8. Kowalczuk, Z., Gunawickrama, K.: Detection and localisation of leaks in transmission pipelines. In: Korbicz, J., Kościelny, J.M., Kowalczuk, Z., Cholewa, W. (eds.) Fault Diagnosis: Models, Artificial Intelligence, Applications, pp. 821–864. Springer, Berlin (2004). Chapter 21
9. Kowalczuk, Z., Tatara, M.: Analytical modeling of flow processes: Analysis of computability of a state-space model. In XI International Conference on Diagnostics of Processes and Systems, pp. 74.1-12 [USB]. Łagów Lubuski (2013). 8–11 September 2013
10. Reddy, H.P., Narasimhan, S., Bhallamudi, S.M., Bairagi, S.: Leak detection in gas pipeline networks using an efficient state estimator. Part-I: theory and simulations. Comput. Chem. Eng. **35**(4), 651–661 (2011)
11. Strikwerda, J.: Finite Difference Schemes and Partial Differential Equations. SIAM, Philadelphia (2007)
12. Torres, L., Besançon, G., Verde, C.: Leak detection using parameter identification. The 8th IFAC Symposium SAFEPROCESS-2012, Mexico City, Mexico (2012)
13. Walpole, R.E., Myers, R.H., Myers, S.L., Ye, K.E.: Probability and Statistics for Engineers and Scientists, 9th edn. Prentice Hall, Boston (2012)

Leak Detection in Liquid Transmission Pipelines During Transient State Related to a Change of Operating Point

Paweł Ostapkowicz and Andrzej Bratek

Abstract This article presents leakage detection techniques in a liquid transmission pipeline. It is focused on leaks, which occur during changes of a pipeline's operating conditions. Elaborated procedures aimed at leakage evaluation are presented. They are based on measuring of process variables such as flow and pressure. The presented solutions do not involve implementing of complex mathematical process dynamics models. The procedures are evaluated by carrying out experimental tests on a physical model of the pipeline.

Keywords Pipeline · Leak detection · Transients

1 Introduction

Leak detection system (LDS) installed on the transmission pipelines widely utilize the analytical methods (internal methods) which are focused on measuring of a pipeline's internal flow parameters: flow rate, pressure, temperature.

Such methods, which might be reviewed in the papers [2, 7, 8], considered separately cannot ensure achieving all diagnostic goals, due to the fact that their utility is restricted to defined pipeline's operational states and leakage characteristics.

In case of liquid transportation pipelines, operating in steady states, so called simplified methods, are found to be quite efficient [3]. Among the examples of such commonly used methods are: gradient method [6], pressure wave detection method [4] and flow balance method [9]. Nevertheless, in case of leakages in transition

P. Ostapkowicz (✉)
Faculty of Mechanical Engineering, Bialystok University of Technology, Bialystok, Poland
e-mail: p.ostapkowicz@pb.edu.pl

A. Bratek
Industrial Research Institute for Automation and Measurements, Warsaw, Poland
e-mail: abratek@piap.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_18

states, which are the consequences of routine operational actions like control set-points change, opening or closing of valves or pump's startup or stoppage, the standard configurations of mentioned above methods might turned out to be inefficient.

Leakage diagnosis in unsteady states is usually carried out on the basis of models of object dynamics. The review of such methods could be found in the paper [5].

According to [7], solving a system of complex differential equations (over a dozen or even several dozens) built for typical transmission pipeline's section of 30–100 km long might be computationally difficult. Not only object dynamics model is considered by such solutions, but also they include the use of state observers, estimation and correlation techniques, that are often essential for estimating many of process parameters, e.g. friction coefficient [1, 2, 7, 8, 11, 12].

Performing leakage detection, based on object dynamics models, creates strong requirements measurement systems must meet. This results in high costs of LDS system, related to the need of using the appropriate quantity of very precise measurement instruments, well synchronized data transmission system and powerful computers.

In many cases, methods presented in scientific literature are strictly theoretical and often are based on the data obtained from simulations and not from real pipeline installations, what may have a negative impact on the ability to clearly characterize flow phenomena. In addition, only very small changes of a pipeline's operating point are considered in the researches, which may result in obtaining a rather simplified solution, especially in case of major changes of operational conditions.

Due to the mentioned above factors, it was assumed that to detect a leakage it is not only recommended to implement complex object dynamics models, but also it might be worth considering using simplified procedures as well or potentially using of the existing simplified method that had been modified in a certain way. Such approach has been adopted in this paper. Besides, the scope of these methods was determined to capture even considerable changes in a pipeline's operating point and a single leakage. In order to evaluate the efficiency of elaborated methods, data obtained from an experimental pipeline's physical model installation was utilized.

Besides discussing and verifying the defined range of applications, arguments were also presented in favor of potential usage of elaborated procedures to diagnose leaks in different operational conditions.

2 Experimental Data Acquired from the Model Pipeline

The pilot pipeline designed for pumping water is located in the Faculty of Mechanical Engineering of the Bialystok University of Technology.

2.1 The Pilot Pipeline Installation

The examined pipe's length is $L = 380$ m. It is made of polyethylene tubes (HDPE) of an outer diameter $D = 40$ mm and an internal one $d = 34$ mm. It is equipped with a variable flow pump and two semi-open tanks (on inlet and outlet), each one with a capacity $V = 300$ dm^3. Its measuring instruments fulfil the general requirements raised by analytical techniques in respect to the following measurements:

- pressure $p(z_{in}, t)$ at the inlet, $p(z_{out}, t)$ at the outlet and $p(z_i, t)$ at several points $i = 1,\ldots, j$ down the pipeline
- volumetric flow rate $q(z_{in}, t)$ at the inlet and $q(z_{out}, t)$ at the outlet of the pipeline,
- states of control devices (pumps, valves) to monitor changes of operational conditions in the pipeline,

where z describes distance from the pipeline's inlet; L is total length of the pipeline; t is time; z_{in}, z_{out} is inlet and outlet coordinates (usually $z_{in} = 0$, $z_{out} = L$).

On the pilot pipeline two electromagnetic volume flowmeters (mounted at the inlet and outlet at coordinates $z_{in} = -6$ m and $z_{out} = 382.2$ m), six pressure sensors (mounted at points $z_{1(in)} = 1$ m, $z_2 = 61$ m, $z_3 = 141$ m, $z_4 = 201$ m, $z_5 = 281$ m and $z_{6(out)} = 341$ m) and one temperature sensor (mounted at the inlet) have been installed.

2.2 Conditions of Experiments

Series of experiments were performed on the pipeline's model. They consisted on changing nominal parameters of water pumping, i.e. the pipe's operating point,[1] by increasing the pump's rotation velocity in conjunction with a simultaneous simulation of a single leakage in the transient state.

The experiments' time slot covered 300 s time frame. In this period the measurement and data acquiring system, mounted on the pipeline, all flow and pressure signals, pump's rotation velocity signals and signals controlling the valve's aperture to simulate a leak were registering. Signals were sampled at $f_p = 100$ Hz.

Prior to changing the pump's rotation velocity, the pipeline was operating in steady state conditions. The shift of setting occurred in the 180th second with identical characteristics for all the experiments. It resulted in a significant change of nominal flow rate from $q_{nom} \approx$ 140l/min (before pump switching) to $q_{nom} \approx$ 155l/min, amounting to around a 10 % change in flow rate, while the pressure behind the pump (on the inlet) increased from $p_{in} \approx 7.5$ bar to $p_{in} \approx 9.0$ bar.

[1]Operating point of the pipeline is defined by the values of nominal flow rate as well as inlet and outlet pressure.

The leaks were simulated at several points on the pipeline by rapid opening of electromagnetic valves, mounted in leak taps. The leaks' intensity observed was from 0.4 to 1.0 % of the nominal flow rate in the pipeline (in reference to the value before the shift of pump setting). The leakages at individual pipeline's points were simulated in the following manner: their starting point was synchronized with the time when the pressure wave, resulting from the change of the pump's operating conditions, had reached these points.

Besides carrying out the leakage simulations, the similar experiments without simulating of leaks were performed.

3 Leak Detection Procedures and Results of their Implementation

The proposed procedures cover performing of the whole range of diagnostic tasks i.e. detection, localization and size estimation of single leakage, whose occurrence takes place in transient conditions of a flow. The presented in the following sections procedures were implemented in the Matlab environment.

3.1 Leakage Detection

Procedures' Characteristics. The researches covered testing of a few leak detection procedures. The implemented solutions are based on generating of numerical indicators (indexes), which may be considered as a measurement of a pipeline's tightness. Their value at the point of crossing the threshold indicates that a leak occurred in the pipeline. Among these procedures we might find the following:

- $B_0\{IF\}$ procedure: based on the volume flow balance on a pipeline's inlet and outlet (1), which use the flow rate measurements q_{in} and q_{out}

$$B_0(t) = q_{in}(t) - q_{out}(t) \tag{1}$$

- $G_{n-m}\{IF\}$ procedure: it analyzes the pressure gradient distribution on two sections of a pipeline with the use of the pressure measurements p_{n-1}, p_n, p_m, p_{m+1}

$$G_{n-m}(t) = G_{in} - G_{out} \tag{2}$$

where G_{in} is pressure gradient for the section of the pipeline in front of the leak point (i.e. between sensors $n-1$, n), and G_{out} is pressure gradient for the section

of the pipeline behind the leak point (i.e. between sensors m, $m + 1$); wherein for the pilot pipeline $2 < n < m < 5$.
The gradients G_{in} and G_{out} are determined as follows:

$$G_{in}(t) = \frac{p_{n-1}(t) - p_n(t)}{z_n(t) - z_{n-1}(t)} \tag{3}$$

$$G_{out}(t) = \frac{p_m(t) - p_{m+1}(t)}{z_{m+1}(t) - z_m(t)} \tag{4}$$

In general, this procedure (similarly to the gradient method [6]), involves installing of at least four pressure sensors along the pipeline. Leak detection might be only identified in the pipeline's section located between the second and the last but one pressure measurement point down the pipeline.

- $AR_k\{IF\}$ procedure: it is based on the analysis of the deviations (residuum) between the pressure measured in the determined point $i = k$ $(1 < k < j)$ and the value modelled at this point. The model is considered as a pipeline's pressure distribution based on the measurements: $p_{1(in)}$ and $p_{j(out)}$. To run the procedure, it is required to use the pressure measurements p_1, p_k, p_j:

$$AR_k(t) = p_k(t) - \hat{p}_k(t) \tag{5}$$

where $\hat{p}_k$ is the estimated pressure value at the measurement point k.

Features and Results of Procedures' Preliminary Implementation. The procedures were tested with the following algorithm:

1. Apply to the obtained data a moving time window of the width $\tau = 1$ s, by replacing consecutive 100 samples of measured flow and pressure signals with their window averaged value. Next, shift the window by $s = 1$ measurement sample. Variables obtained this way will be used in the next steps.
2. Determine the model pressure value $\hat{p}_k$ for the AR_k procedure using captured data at the pipeline's inlet and outlet pressure measurement points.
3. Determine the IF indicator functions for individual leak detection procedures: B_0, G_{2-5}, G_{2-4}, G_{2-3}, G_{3-5}, G_{3-4}, G_{4-5}, AR_2, AR_3, AR_4, AR_5.
4. Calculate indicator functions' statistical data measures: $E\{IF\}$, $std\{IF\}$ on the basis of data obtained for a testing period. In the case of performed experiments, the following test time window was defined: 10 s $< t <$ 90 s.
5. Determine the values of the alarm threshold P.
   For individual procedures the threshold values were set on the basis of the average statistical data measures values of the indicator function IF in the defined test period

$$P = aver(E\{IF\}) - b \cdot aver(std\{IF\}) \tag{6}$$

Coefficient b in the relationship (6), which decides about the procedure's sensitivity, was determined as a duplicated minimum value of coefficient $b_{\min}$, at which the *IF* function value in the test window didn't drop below the threshold in all test attempts related to the same leakage location. For the investigated procedures, the coefficient $b = 2 \cdot b_{\min}$.

6. Generate an alarm for leak occurrence when the given indicator function *IF* exceeds its *P* threshold value, that is

$$IF < P \tag{7}$$

With the aim to detect a leakage, pump's control signal was also used. Two areas for the individual *IF* indicator functions analysis were determined as well:

- before switching of the pump (range *I*)
- after switching of the pump (range *II*).

The range *I* covered the period 90 s $< t <$ 180 s, while the range *II* was related to 185 s $< t <$ 275 s time window. Between both windows, i.e. 180 s $< t <$ 185 s, transient state—provoked by a rotation velocity change—was extinguished. At this time the preliminary filtering of measurements, before performing the leakage analysis in the range *II*, took place as well. It was assumed that in the range *I* (leak less transmission) the values of the individual indicator functions *IF* should be over their threshold values *P*. While in the range *II* (leakage transmission) the *IF* indicator yet should drop below the *P* threshold at the very beginning.

The results of tested procedures obtained for the experiments with different size of leakage at the point of coordinate 155 m are presented in Table 1. Detection of a given size leakage was indicated with a '●/' sign. Non detection of leakage i.e. when the *IF* indicator function's value didn't overcome the *P* alarm threshold in the *II* range was indicated with a '°/' sign.

Table 1 Leakage detection results (standard and modified procedures)

No.	z_{leak}	q_{leak} (%)	Procedures										
			B_0	G_{2-5}	G_{2-4}	G_{2-3}	G_{3-5}	G_{3-4}	G_{4-5}	AR_2	AR_3	AR_4	AR_5
1	155	1.0	●/-	●/●	●/●	°/●	●/°	●/●	°/°	°/●	●/●	●/●	●/●
2		1.0	●/-	°/●	●/●	°/●	●/●	●/●	°/°	°/●	●/●	●/●	°/●
3		0.6	●/-	°/●	°/●	°/●	●/●	●/●	°/°	°/●	●/●	●/●	°/●
4		0.7	●/-	°/●	°/●	°/●	●/●	●/●	°/°	°/●	●/●	●/●	°/●
5		0.5	●/-	°/●	°/●	°/●	●/●	●/●	°/°	°/●	●/●	●/●	°/●
6		0.5	●/-	°/●	°/●	°/°	●/●	●/●	°/°	°/●	●/●	°/●	°/°
7		0.4	●/-	°/●	°/●	°/●	●/●	●/●	°/°	°/●	●/●	°/●	°/°
8		0.4	●/-	°/●	°/●	°/°	●/●	●/●	°/°	°/●	●/●	°/●	°/°

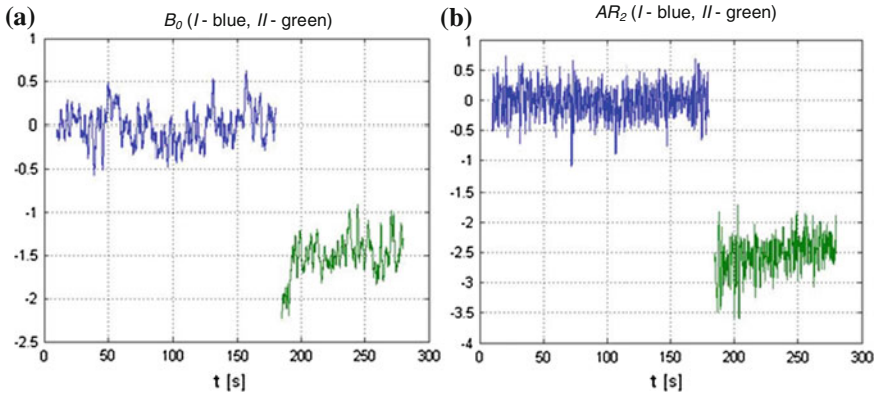


Fig. 1 Indicator functions *IF* for procedures: **a** B_0, **b** AR_2

Additionally, Fig. 1 presents example indicator function's distributions B_0, AR_2, determined in the leakage simulation at the same pipeline's point, whose intensity was 1 % of the nominal flow rate. While analyzing the function B_0, AR_2 (Fig. 1) in the range *II*, we might observe a clear change of its values with comparison to their values in the range *I*. Furthermore, it is evident that the change has a solid sustained character due to the fact that the function value gets stabilized at the new level, what indicates clearly the leakage.

Nevertheless, it is worth mentioning further that for all carried out experiments with leakage simulations, the individual indicator functions *IF* fitted the accepted variation range (i.e. over the *P* alarm threshold) in range *I*. While, in the case of experiments without leakages, indicator functions *IF* were in the accepted range both in *I* and *II* range.

In case of the largest simulated leakages, the changes in indexing values are considerable enough, so that it is quite easy to detect a leak. On the contrary, in case of smaller leakages, it might be a problem.

Hence, except for the procedure B_0, it was suggested to modify the procedures. The modification consisted in including of reference element retrieved from the indicator function for identical pump's rotation velocity change, but without a leakage simulation. In this manner it was possible to evaluate the growth factor of the indicator function *IF* for every single procedure, which relates to the change in pipeline's operating point. The growth factor is defined as follows:

$$\Delta_{IF} = aver\left(E\left\{IF^{II}\right\}\right) - aver\left(E\left\{IF^{I}\right\}\right) \tag{8}$$

where *I* is the range before pump's switching, and *II*—range after pump's switching.

While calculating again the indicator functions IF on the basis of the relationships defined previously, these were modified with the growth factor in the below way:

$$IF' = IF + \Delta_{IF} \tag{9}$$

Taking into account the newly defined P' alarm thresholds for individual indicator functions IF', the obtained results for experiments are also presented in Table 1. Detection or no detection of a leak of a given size is indicated by '/•' and '/°'.

Similar to the previously discussed cases, this time also all indicator functions IF' determined in range I for the performed experiments stayed within the accepted variation range. Moreover, indicator functions IF' were within the accepted range both in range I and range II in case of additional leak-free tests.

The discussed methods varied from the leak detection effectiveness point of view. In case of their standard variants, the best results were achieved for the flow balance method B_0, as the leaks were successfully encountered in every single simulation. The other methods (AR_k and G_{n-m}) gave, between them, similar results. On the other hand, while comparing the methods according to the measurement data source, more efficient techniques were those which were focused on pressure measurements carried out near the leakage location, than the ones performed at farther distant points.

Modification of indicator functions had a positive effect. The effectiveness of the procedures AR_k and G_{n-m} improved significantly. Besides, more balanced leak detection was obtained on the basis of measurements near leak location.

The best results of leak detection were obtained in the points located in the middle section of the pipeline, while slightly worse results were achieved for the pipeline's outlet and the worst for the pipeline's inlet. Such distribution of the results is a well-known issue in leakage diagnostics. This certainly might be due to the large extent and dynamics of flow parameters' change resulting from the shift of the installation operating point after increasing pump's rotations, mainly at the pipeline's inlet.

Without any doubt, the value of coefficient b has a significant impact on the level of leak detection. On the other hand, the extent to which we separate the threshold P (or P') from the normal range of indicator function IF (or IF') results in a loss of technique's sensitivity and also in its vulnerability to false alarms increases.

3.2 Leakage Localization

In order to diagnose leaks, a two-stage method was implemented:

- rough localization—to identify the leakage area,
- precise localization—to determine the leak's location coordinate z_{leak}.

This technique uses the calculated results obtained from modified indicator functions IF' (see Table 1).

Besides the procedure mentioned above, a different technique, focused on decreasing the time needed to detect and locate a leakage, was also proposed.

Rough Localization Procedure. Rough localization of leakage area was carried out at exactly the same moment when a leak was detected within the B_0 procedure. The rough localization technique consisted in determining pressure sensor number $i = k$, for which the AR_k procedure's indicator function IF' achieved its minimal value. The possible leak area was limited to the pipeline's segments adjacent to the sensor $i = k$, i.e. pipeline's segments between $i = k - 1$ and $i = k + 1$ pressure sensors.

Table 2 presents exemplary results of the rough localization procedure gained while detecting the simulated leakages at coordinate 155 m. The results in Table 2 have been generalized due to the fact that, in the case of simulated leak experiments at the determined pipeline's points, the acquired localization results were exactly the same. It is also crucial to highlight that the results of the rough localization method obtained for all simulated leakages corresponded to their real localization.

Depending on the leaking location, detection was performed between 4.2 and 4.7 s, counting from the moment of their occurrence. In reality, implementation of procedures G_{n-m} and AR_k might result in much quicker identification of leakage, even up to 1–1.6 s earlier. Moreover, the leak detection time could be further shorten, reducing two or three times the averaging period of measurements.

Precise Localization Procedure. Precise localization was performed within the gradient method [6]. To determine the required pressure gradients, sensor pairs $g_{in} = \{i = k - 2; i = k - 1\}$ and $g_{out} = \{i = k + 1; i = k + 2\}$, situated the closest possible to the sensor $i = k$ on its both sides, identified with the rough localization, were used. The leak localization was defined as follows:

Table 2 Leaks localization results, where A_{leak} is leakage area, t_{loc} is localization time, L means localization successful result (or failure), and U is leak standard deviation

No.	Z_{leak}	q_{leak} (%)	Rough localization procedure		Precise localization procedure			Fast localization procedure	
			t_{loc} (s)	A_{leak}	t_{loc} (s)	L (m)	U (m)	t_{loc} (s)	L (m)
1	155	1.0	4.5	$i = 2 < k < i = 4$	6.5	140.5	5.6	0.85	153.1
2		1.0				138.7	4.9	0.96	155.2
3		0.6				138.9	5.2	0.88	155.9
4		0.7				143.2	6.2	0.93	155.7
5		0.5				141.8	6.6	0.95	153.3
6		0.5				149.9	9.5	0.90	153.5
7		0.4				135.4	9.6	0.97	159.5
8		0.4				135.7	8.7	0.96	153.3

$$z_{leak} = \frac{1}{N} \sum_{S=1}^{N} z_{(100)S} \tag{10}$$

where $z_{(100)S}$ is the leak's coordinate found on the basis of the S-th sequence of 100 following pressure samples, and z_{leak} is the localization result based on N calculation cycles performed just after a leakage detection ($N \cdot 100$ scans).

Table 2 columns, corresponding to the precise localization procedure, show the results of the localization z_{leak}, with the averaging parameter $N = 3$, obtained 6.5 s after a leak occurrence and a standard deviation of the variable $z_{(100)S}$, calculated on the basis of the population of the results for $S = 1, 2, \ldots, 27$.

The localization results for individual leak tests indicate the leakage's location to be between 138 and 151 m of a pipeline. It is quite difficult to observe a relationship between the localization's error and the leak's intensity (size). However, it is indeed observed almost twice increase of the location's standard deviation with the decrease of the leakage from 1, to 0.4 % of the pipeline's nominal flow rate. The leakage's area is significantly narrowed in comparison to the leak's area obtained with the rough localization procedure, although the results are asymmetrically shifted in comparison to the real leak's position.

Fast (Early) Localization Procedure. This procedure is based on the well-known wave pressure detection method [4, 10], which is commonly used to diagnose single leaks in steady conditions. The discussed procedure assumes that it is also possible to observe the phenomenon of pressure wave propagation if a leak occurs in a transient state and notice the waves' front in particular, which is presented in Fig. 2a.

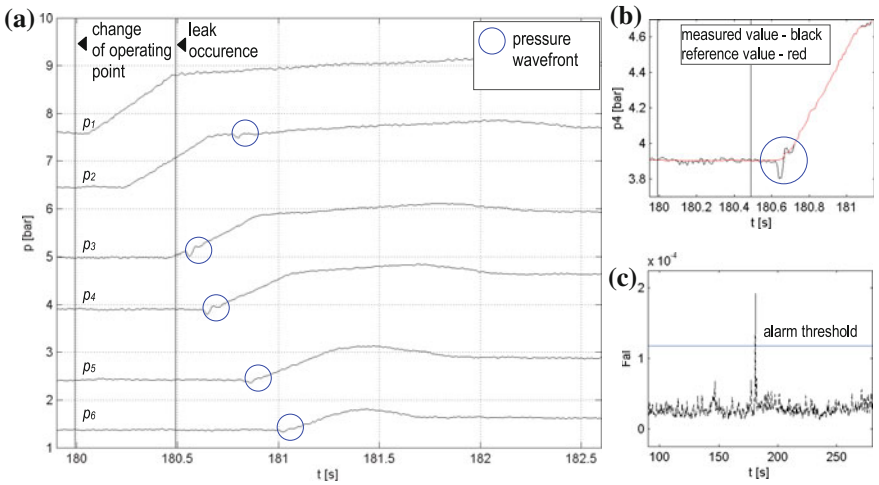


Fig. 2 Procedure execution: **a** measured pressure signals, **b** measured pressure signal and its reference, **c** series of function values generated by the leak detection algorithm

In practice such assumption applies to sudden leakages. Therefore, this procedure might be less effective in case of the pressure waves with smoothen fronts, which are formed as a result of slowly developing leaks and ineffective in case of extremely slowly in leakages.

From a practical point of view, the procedure is implemented by applying two concurrent algorithms. Both algorithms were created with the use of the solutions proposed in paper [10]. They use residual variables Δp_i calculated for each pressure measurement point z_i according to the following:

$$\Delta p_i^s = p_i^s - \bar{p}_i^s \tag{11}$$

where p_i^s is the value of measured pressure signal at the moment s, $\bar{p}_i^s$ is the reference value calculated by applying median filter, and s means the moment corresponding to the sampling of measured signals (see Fig. 2b).

The aim of the first algorithm is to detect a leakage. The second one is responsible for detection and identification of pressure wave's front in individual measurement points z_i. The time moments $t_{wav}(z_i)$, being a result of the pressure wave's front identification, along with known coordinates of the measurement points z_i, enable the leakage localization. Implementation of the reduced alarm threshold margins, which are used to detect and identify a wave's front, is an innovative solution. Such approach should increase a time moment's $t_{wav}(z_i)$ determination accuracy. However, in order to avoid generating false alarms in states without leakages, the second algorithm is synchronized with the first one.

In Fig. 2c the shape of diagnostic function F_{al}, generated with the first algorithm, is presented (for the same experiment as shown in Fig. 1). In Table 2 we might review the results of localization of identical leakages, with the means of previously discussed methods, along with the time required to perform it.

Considering the presented results, we might notice the proposed procedure provides satisfactorily accurate leak localization and significantly shortens the time needed to perform it.

3.3 Other Potential Applications of Elaborated Procedures

The elaborated procedures might as well be applied to detect multiple leaks. In case of transmission pipelines non-concurrent leaks are quite common (in contrary to concurrent leaks). Procedure B_0 enables direct estimation of the intensity of the single leakage and potentially multi-non-concurrent leaks. To calculate the intensity estimation, it is required to evaluate the flow balance, on the basis of the measured flow signals q_{in} and q_{out} and the calculated intensities q_{leak} for the following leaks. However, such calculations should correspond to the new steady states achieved between the occurrences of consecutive leaks, which might result in quite long diagnostic time.

Detection and localization of non-concurrent leaks, occurred between enough long periods of time, should be available with the procedures G_{n-m}, AR_k and the procedure based on the wave pressure detection. In case of detection of concurrent leaks, the procedure G_{n-m} could be used providing that it is possible to estimate the pressure gradient between the locations of following leaks or with a support of AR_k procedure. The procedure based on the wave pressure detection might be used as well combined with identification of propagation, reflection and attenuation of pressure waves, caused by each leakage.

Considering the complexity of multiple leaks diagnosis, which is mentioned in [12], the above presented arguments should rather be treated supposedly due to the fact that they would still need to be confirmed in experiments. Besides, in case of the elaborated, simplified procedures, it might be indispensable to modify them to certain extent or even enlarge their scope by introducing elements used by more advanced solutions, which may be found in papers [7, 8, 11, 12].

4 Conclusion

This paper presents several simplified though efficient procedures to be used in diagnosing leakages in transmission pipes' transient states.

The procedures' effectiveness has been proved by the obtained results of their implementation on the model pipeline, when for the given change of pipeline's transmission conditions, related to a pump's rotation velocity change, of around 10 % of the nominal flow rate, it was possible to detect even 0.4 % volume single simulated leakages in transient states. Such results were obtained for all discussed procedures, which were based on: flow balance on pipeline's inlet and outlet (procedure B_0), pressure gradient analysis (procedure G_{n-m}) and a comparison of measured pressure with calculated pressure according to the model distribution (procedure AR_k).

Moreover, the elaborated methods provided satisfactorily precise localization results for simulated leakages. Besides rough localization and precise methods, a procedure was proposed, which is based on pressure wave detection method and uses own developed algorithms. Under some flow conditions they enable even faster leakage detection and localization.

In conclusion, it might be stated that the presented above simplified procedures might be complimentary to the used process dynamics models to detect leakages.

Acknowledgment This research work has been financed with the means of the Ministry of Science and Higher Education (Poland) in the years 2010–2015 as the research project Nr N N504 494439.

References

1. Begovich, O., Pizano-Moreno, A., Besancon, G.: On-line implementation of a leak isolation algorithm in a plastic pipeline prototype. Latin Am. Appl. Res. **42**, 131–140 (2012)
2. Billman, L., Isermann, R.: Leak detection methods for pipelines. Automatica **23**, 381–385 (1987)
3. Bratek, A., Turkowski, M.: Analytical system of leak detection and localization for long range liquid pipeline. Pomiary Automatyka Kontrola **58**(1), 15–18 (2012)
4. Chun-hua, T., Jun-chi Y., Jin, H., Yu W., Dong-Sup, K., Tongnyoul, Y.: Negative pressure wave based pipeline leak detection: challenges and algorithms. IEEE International Conference on Service Operations and Logistics, and Informatics, pp. 372–376, Suzhou, China (2012)
5. Colombo, A.F., Lee, P., Karney, B.W.: A selective literature review of transient-based leak detection methods. J. Hydro-Environ. Res. **2**(4), 212–227 (2009)
6. Feng, J., Zhang, H.: Oil pipeline leak detection and location using double sensors pressure gradient method. 5th World Congress on Intelligent Control and Automation, vol. 4, pp. 3134–313, Hang Zhou, China (2004)
7. Isermann, R.: Leak detection of pipelines. In: Fault-Diagnosis Applications: Model-Based Condition Monitoring: Actuators, Drives, Machinery, Plants, Sensors and Fault-tolerant Systems, pp. 181–204. Springer, Berlin (2011)
8. Kowalczuk, Z., Gunawickrama, K.: detecting and locating leaks in transmission pipelines. In: Korbicz, K.J., Kościelny, J.M., Kowalczuk, Z., Cholewa, W. (eds.) Fault Diagnosis: Models, Artificial Intelligence, Applications, pp. 822–864. Springer, Berlin (2004)
9. Liou, C.: Pipeline leak detection based on mass balance. Proceedings of the International Conference on Pipeline Infrastructure II, ASCE (1993)
10. Ostapkowicz, P.: Leakage detection from liquid transmission pipelines using improved pressure wave technique. Maint. Reliab. **16**(1), 9–16 (2014)
11. Torres, L., Verde, C., Besancon, G., Gonzalez, O.: High gain observers for leak detection in subterranean pipelines of liquefied petroleum gas. Int. J. Robust Nonlinear Control **24**, 1127–1141 (2014)
12. Verde, C.: Multi-leak detection and isolation in fluid pipelines. Control Eng. Pract. **9**, 673–682 (2001)

Accuracy Investigations of Turbine Blading Neural Models Applied to Thermal and Flow Diagnostics

Anna Butterweck and Jerzy Głuch

Abstract Possibility of replacing computational fluid dynamics simulations by a neural model for fluid flow and thermal diagnostics of steam turbines is investigated. Results of calculations of velocity magnitude of steam for a 3D model of the stator of a steam turbine is presented.

Keywords Artificial neural networks · Steam turbine modeling · Computational fluid dynamics

1 Introduction

Technical diagnostic [1] is the examination of symptoms and syndromes to determine the nature of faults or failures of technical objects. A symptom [1] is a perception, made by means of human observations and measurements, which may indicate the presence of an abnormal condition with certain probability. A syndrome [1] is a group of symptoms that collectively indicate or characterize an abnormal condition. Technical diagnostic systems allow to assure safe operation of various types of objects.

The systems are often based on measurements of physical quantities during operation of the technical object. When it comes to diagnostic of energy facilities, like power plants, it occurs that measurements can provide only partial information of the operation. Power plant efficiency is determined mainly by efficient operation of all the devices and components of the turbine thermal cycle [2, 3]. Quantities like pressure, temperature, mass flow etc. can be measured only in some locations like inlet of the turbine, outlet of the turbine, extractions etc. but never inside of the turbine itself. Precise information about fluid flow through turbine blade rows would allow predicting and fast detecting of the operation degradation of the turbine that have direct influence on the turbine efficiency. Fluid flow diagnostics may employ

A. Butterweck · J. Głuch (✉)
WETI, Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland
e-mail: jgluch@pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_19

various methods. Detection requires the identification of the symptoms of inefficient work. Methods based upon symptoms represent the pattern-based diagnostics [2, 4–8].

The basic requirement for diagnostic models is fast pattern recognition [4, 5]. Up to date, the applied 1D diagnostics models are fast enough but they do not provide details of the flow [4, 5]. Spatial models (like Computational Fluid Dynamics (CFD) simulations) allow accessing distributions of thermodynamic parameters of the flow, but the time of the calculations is vastly prolonged. The application of detailed multidimensional methods in the diagnostics must therefore lead to their modification, which should keep the accuracy of the calculation, significantly reducing the computation time. Such can be achieved by the use of neural models (models based on artificial neural networks (ANN)).

The possibility of application of neural models to thermal problems was discussed in literature, especially complex analysis is given in [9]. Also certain examples of diagnostic applications of neural models were reported [10–13], but they mostly concern application to gas turbines and none of them deal with gaining the distributions of thermal parameters in turbine blading flow channels.

Neural modeling that replaces the detailed 3D methods, has already been reported in the literature [14, 15]. However, this neural modeling was only used to support the design of thermal turbines, and therefore to determine the geometric parameters for the design load of palisades. Neural modeling for diagnostic purposes has to be done in a different way as the reference geometry and thermal parameters are known but, working conditions vary and degradation of the object occurs during the operation of the object [2]. Example of the degradation of turbine blades is presented in Fig. 1.

Selected results of neural modeling of fluid flow through channel between blades of steam turbine (of known geometry) with changing operating conditions are shown. The aim is to create a substitute model to CFD. New model (artificial neural network) should be characterized by short computation time and high accuracy. The training may take a vast amount of time, but the resulting model takes minutes or seconds for the range of boundary conditions.

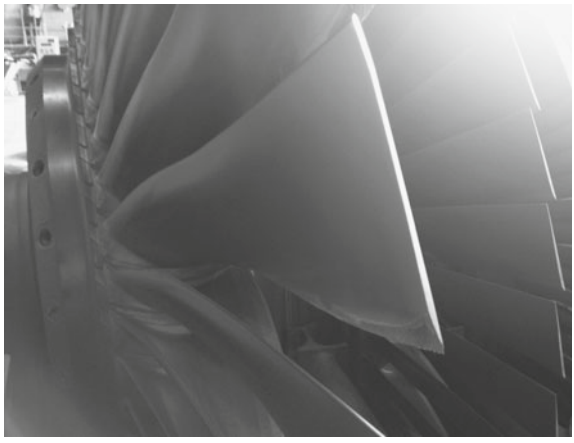


Fig. 1 Example of degradation of the *trailing* edge of the turbine

2 Modeling of the Flow

The CFD simulations are made by calculating equations (the continuity equation, momentum conservation equation, energy conservation equation, equation of fluid, state equation) in every finite volume (node of the mesh) of fluid flow channel. The calculations are based on geometrical information, initial and boundary conditions and they result in distributions of thermodynamic and flow parameters (Fig. 2). Fluid flow equations, that are highly nonlinear, are solved in iterative way converging on different levels of residual criteria resulting in long computation time. To apply certain model in fluid flow diagnostics its short time of calculations (within seconds) is essential. To find substitute to commonly used CFD simulations, authors decided to test the capabilities of neural models.

As a test case the channel between two stator blades of steam turbine was chosen. To train the neural model it is needed to have a reference database. To provide training data, CFD simulations were made, providing the distributions of quantities like pressure, velocity, temperature, etc. in flow channel. During operation of the turbine the boundary conditions (presented in Table 1) varies, the simulations were calculated from 100 % to 93 % of designed boundary conditions (BC). As a test cases, excluded from training process, 94 and 94.5 % of design boundary conditions (BC) were chosen. In Fig. 3 the intersection of the 3D mesh of fluid flow channel is presented.

Neural models themselves do not contain any physics of the modeled phenomena. Therefore, it is a great responsibility to set proper input values to neural model in order to be able to reproduce physics of the fluid flow.

To introduce physics of the fluid flow and processes of energy transformation in the flow channel input data to neural model, it was assumed that such should correspond to the data used in analytic models. This neural model applies only to working medium (gas) of well-known properties. Individual neural models were built for each

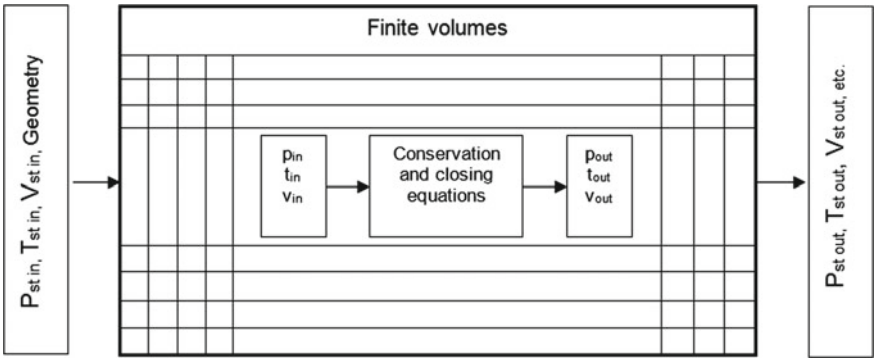


Fig. 2 Finite volume scheme for 3D numerical calculations of the turbine blading

Table 1 Design boundary conditions

In pressure [MPa]	In temperature [K]	Out pressure [MPa]	Out temperature [K]
7,93	746	7,22	732

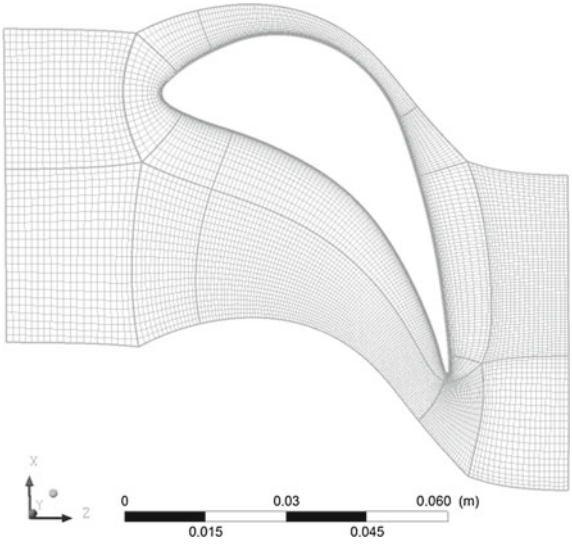


Fig. 3 Intersection of 3D mesh

of the quantities describing the flow field, such as pressure, velocity, energy loss, etc., providing better quality of the modeling [5, 16].

In Fig. 5 a simplified diagram of individual model of the flow channel is shown. The input parameters of the neural models (Fig. 5) were set as follows:

- thermodynamic parameters, represented by inlet pressure of investigated flow channel in changing load conditions
- physical boundaries of channel, defined by geometry of the turbine blades.

Testing artificial neural networks for turbine blading showed that using one neural network for whole fluid flow channel do not meet accuracy standards to use it as a substitute to CFD simulations for fluid flow diagnostics. Also attempts were made to divide fluid flow channel into few volumes to use neural networks for the volume but the results were also unsatisfying. That is why approach where one finite volume corresponds to one artificial neural network has been applied. The approach allow preserving accuracy of the calculations with neural model for design geometry of turbine blade operating in varying conditions. Neural model bases on feedfoward artificial neural networks and was trained with Levenberg–Marquardt algorithm.

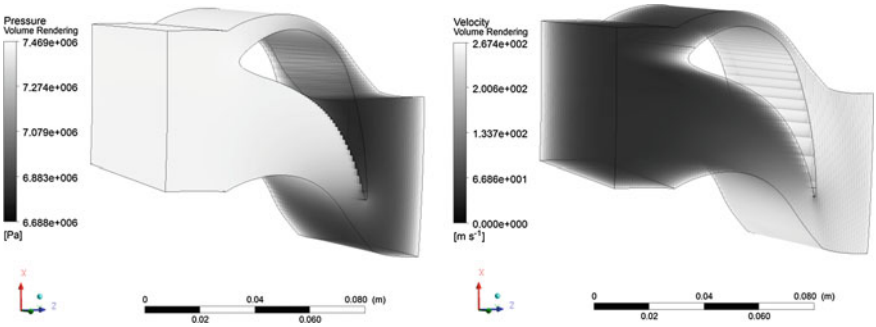


Fig. 4 Example of the results calculated with the CFD solver: (*left*) 3D distribution of pressure (94 % of design BC); (*right*) 3D distribution of velocity (94 % of design BC)

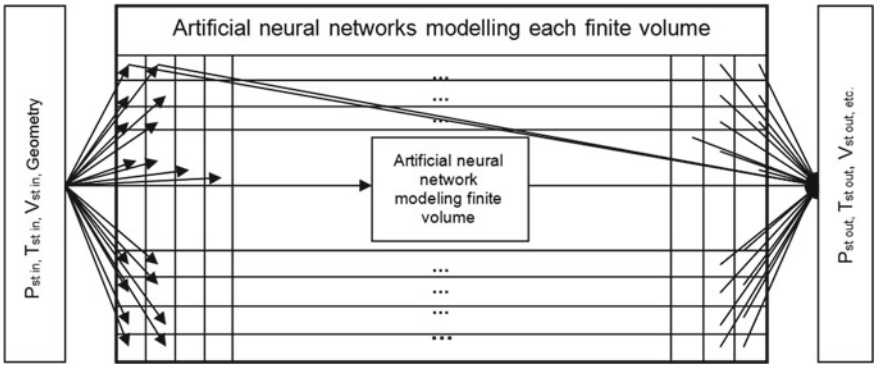


Fig. 5 3D neural modeling of steam turbine blading (for each finite volume one artificial neural network is used)

The time of training over 440000 artificial neural networks was extended, but the applied computerization of the process practically compensated this effect. The neural implementation of the trained model allowed us to shorten the time of calculation is about 100 times as compared to the CFD simulations.

3 Results

As a test cases of neural model 94 and 94,5 % of inlet and outlet of designed pressure were chosen. Test cases were not included in training process. To estimate the accuracy of the results of neural model the value of the error in every node of the mesh was calculated, given as relative percentage error:

$$error = \frac{v_{mag,CFD} - v_{mag,ANN}}{v_{mag,CFD}} 100\,\% \tag{1}$$

Table 2 Calculated errors for test cases

Case (percentage of design BC)	94 %	94,5 %
Maximum error [%]	0,642	0,639
Average error [%]	0,0038	0,0043
Number of points with error>0,2 % [%]	7	3
% of points with error >0,2 % [%]	0,001589	0,000681

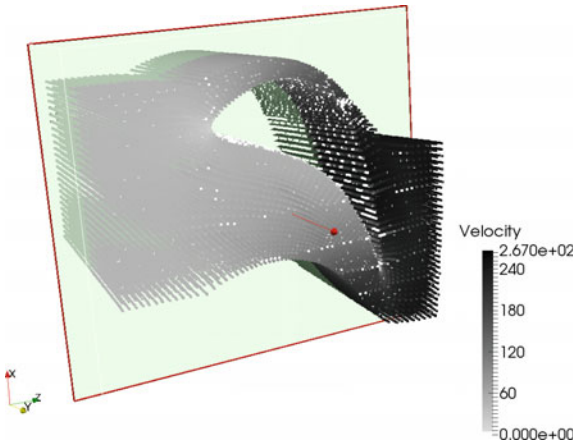


Fig. 6 Example (93,5 % of initial boundary conditions) of 3D distribution of velocity magnitude with the marked plane of intersection

where: *error*—value of the error [%], $v_{mag,CFD}$—magnitude of velocity calculated with CFD solver, $v_{mag,ANN}$—magnitude of velocity calculated with ANN solver. Maximum error, in both test cases, is about 0,6 %. The error higher than 0,2 % occured in 7 nodes (in 94 % of design BC) and in 3 nodes (in 94,5 % of design BC) out of over 440000 nodes. In Table 2 maximum error and average error for both test cases were presented.

To visualize the results of neural model intersection of the flow channel was made. The intersection was made with a plane showed in Fig. 6, also the up section of the geometry was shown as the results (Figs. 7b, 8b).

In Fig. 7 distribution of magnitude of velocity for 94 % of the design BC case is given. In Fig. 8 distribution of error is presented. For 94 % of design BC maximum error is $0,642$ %, error higher than $0,2$ % occurs in 7 of the 440537 mesh points. Application of neural model to preform calculations through flow channel of stator of steam turbine was investigated. The obtained results confirm high accuracy of the neural model.

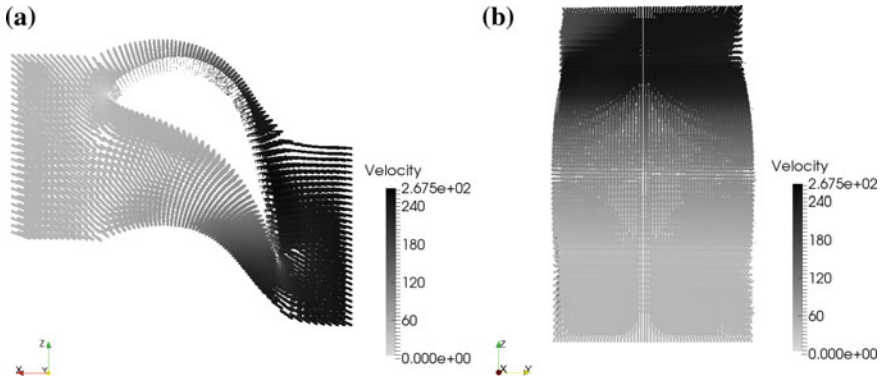


Fig. 7 Distribution of magnitude of velocity calculated with the neural model for 94 % of BC case: **a** side intersection; **b** up section

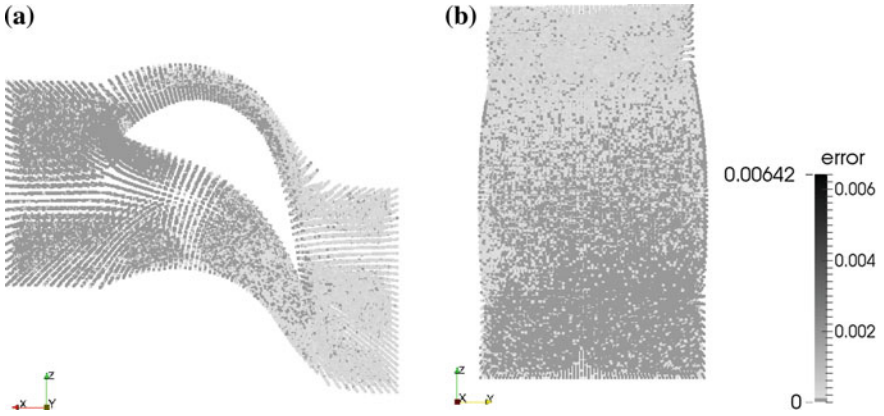


Fig. 8 Distributions of error for 94 % of BC case: **a** side intersection; **b** up section

4 Conclusions

In the paper the investigation of possible application of neural model to thermal and flow diagnostics on the example of steam turbine stator was performed. As the reference data, to train the neural model, the CFD calculations results were used. The neural model showed high accuracy. Moreover, its application shortened time of calculation over 50 times.

Future work will be focused on application of the neural model to turbine stage (stator + rotor) of steam turbine maintaining the high level of accuracy. It requires new set of reference data from CFD calculations. The mesh of the stage of steam turbine will consist of about 10^6 nodes and the CFD calculations will take vast amount of time. Thanks to application of neural model grater shortening of calculation time, than in presented case, is predicted.

References

1. Czichos, H.: Handbook of technical diagnostics. Springer, Heidelberg (2013)
2. Gardzilewicz, A., Gluch, J., Bogulicz, M., Walkowiak, R., Najwer, M., Kiebdój, J.: Experience in application of thermal diagnostics in the turów power station. In: ASME International Joint Power Conference, IJPGC2003-40017, Atlanta (2003)
3. Kosowski, K., Tucki, K.: Turbine stage design aided by artificial intelligence methods. Expert Syst. Appl. **36**(9), 11536–11542. Elsevier (2009)
4. Głuch, J., et al.: Heat and Flow Diagnostic Relations under Industrial Conditions (in Polish). Wyd. Wydział Oceanotechniki i Okrętownictwa Politechniki Gdańskiej, Gdańsk (2007)
5. Głuch, J., et al.: On Application of neural simulators of turbine blading systems 3d flow to the reference state determination in thermal and flow diagnostics (in Polish). Wyd. Politechniki Gdańskiej, Gdańsk (2014)
6. Korbicz, J., Kościelny, J.M., Kowalczuk, Z., Cholewa, W. (eds): Fault Diagnostics. Models, Artificial Intelligence, Applications. Springer, Berlin (2004)
7. Krzyżanowski, J., Głuch, J.: Heat and flow diagnostics of energetic objects (in Polish). wydawnictwo imp pan, Gdańsk (2004)
8. Ślęzak-Żołna, J., Głuch, J.: Towards symptoms of degradation in on-line thermal and flow diagnostics of power objects. In: Safe Process. IFAC, Pekin (2006)
9. Kwang-Tzu, Y.: Artificial neural networks: a new paradigm for thermal science and engineering. J. Heat Transfer **130**(9), 093001 (2008)
10. Mohammadi, R., Naderi, E., Khorasani, K., Hashtrudi-Zad, S.: fault diagnosis of gas turbine engines by using dynamic neural networks. In: Proceedings of ASME Turbo Expo: Power for Land, Sea and Air, GT2010-23, UK, Glasgow (2010)
11. Loboda, I.: Gas turbine fault classification using probability density estimation. In: Proceedings of ASME Turbo Expo: Turbine Technical Conference and Exposition, GT2014-27265, Germany, Dusseldorf (2014)
12. Bettocchi, R., Pinelli, M., Spina, P.R., Venturini, M.: Artificial inteligence for the diagnostics of gas turbines—part I: neural network approach. J. Eng. Gas Turbines Power **129**(3), 711–719 (2006)
13. Loboda, I., Yepifanov, S., Feldsheteyn, Y.: An integrated approach to gas turbine monitoring and diagnostics. In: ASME Turbo Expo 2008: Power for Land, Sea, and Air Volume 2: Controls, Diagnostics and Instrumentation. Cycle Innovations. Electric Power, Germany, Berlin (2008)
14. Duch, W., Korbicz, J., Rutkowski, L., Tadeusiewicz, R.: Neural Networks. In: Nałęcz, M. (ed.) Biocybernetics and Biomedical Engineering (in Polish) 6. Akademicka Oficyna Wydawnicza Exit, Warszawa (2000)
15. Pierret, S.: Designing Turbomachinery Blades by Means of the Function Approximation Concept Based on Artificial Neural Network, Genetic Algorithm, and the Navier-Stokes Equation. Ph.D. Thesis, Faculte Polytechnique de Mons, Von Karman Institute for Fluid Dynamics, Turbomachinery Department (1999)
16. Butterweck, A., Głuch, J.: Neural network simulator's application to reference performance determination of turbine blading in the heat-flow diagnostics. In: Intelligent Systems in Technical and Medical Diagnostics, pp. 137–148. Springer, Berlin (2014)

Proposition of Electromyographic Signal Interpretation in the Rehabilitation Process of Patients with Spinal Cord Injuries

Martin Tabakov, Paweł Kozak and Stefan Okurowski

Abstract Surface electromyography (sEMG) is one of the examinations within the protocol of neuro-rehabilitation processes, that allow the assessment of possible patient progress with respect to conductivity of neurons and skeletal muscle functionality. The interpretation of sEMG signal is one of the critical issues that should be considered in order to diagnose patients with severe spinal cord injuries. Currently, it is very hard to relate values gathered from sEMG to existing reference scale of patient rehabilitation progress. What more, the interpretation of the signal data is very subjective and it is also strongly related to current physical disposition of the patient. Therefore, the objective of our research, is to introduce a mathematical approach which determines the patient's physical condition, based on sEMG data. To achieve this goal, we propose to use properly defined fuzzy Sugeno integral. The proposed operator allows to combine both: subjective expert knowledge and signal data.

Keywords Electromyography · Signal processing · Fuzzy measures · Fuzzy Sugeno integral · Rehabilitation · Neuro-rehabilitation · Spinal cord injuries

M. Tabakov (✉) · P. Kozak
Department of Computational Intelligence,
Wrocław University of Technology, Wrocław, Poland
e-mail: martin.tabakow@pwr.edu.pl

P. Kozak
e-mail: 183610@student.pwr.edu.pl

S. Okurowski
The Neuro-Rehabilitation Center for the Treatment of Spinal Cord Injuries 'AKSON',
Wrocław, Poland
e-mail: aksonaxis@wp.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_20

1 Introduction

Surface electromyography (sEMG) delivers a noninvasive method for the objective evaluation of the electrical activity of the skeletal muscles. It provides a measurement procedure accepted as diagnostic tool for analysis of muscular diseases and neurogenic disorders, providing information about the functionality of the peripheral nerves and muscles [7]. This information has significant influence to the preparation of relevant procedures for patient rehabilitation. Myoelectric signals refer to the system of voluntary muscle contraction, which motor units are activated at different frequencies and their contributions to the signal are added asynchronously [5]. This signal presents harmonics with frequencies ranging from 15 Hz to about 500 Hz, and amplitude from approximately 50 μV–5 mV [3].

Referring to rehabilitation, the basic problem in the therapy procedures for patients with severe spinal cord injuries, is the recognition of small changes of the physical condition of the patient. The generally applicable ASIA scale, introduced by the American Spinal Injury Association which includes the Lovett system [11] are too general, i.e. they are not designed for tracing of small changes of values, for example in the sEMG or other signals, in purpose to recognize relatively small changes of patients physical condition during the rehabilitation process. The recognition, the tracing and finally the visualization of these changes is of significant importance for modern, supported by digital technologies, rehabilitation. But even tracing these values, there occurs the problem of the subjective interpretation of the signal itself, especially for patients with severe spinal cord injuries who reach slight variations in signal data. What more, the signal present the current electrical activity of the skeletal muscles, which is very dependent of the current disposal of the patient and is composed of different muscle contractions. The last causes difficulty in the diagnosis process.

Therefore, the main objective of our research was to apply a mathematical model suitable to provide solution to the problem of interpretation of sEMG data with respect to the current patient disposition, in such a way that it generates one numerical value, which can be traced and which relates to the patient rehabilitation progress.

This article is organized as follows: in Sect. 2 the basic notions referring the mathematical concept proposed are given, in Sect. 3 the basic concept of EMG signal interpretation with respect to the rehabilitation process are presented, in Sect. 4 some experiments regarding to the proposed concept are described, and finally conclusions and further research directions are presented.

2 Basic Notions—Fuzzy Sugeno Integral

In this section, the preliminaries of fuzzy sets [13] and fuzzy Sugeno integral [12] are briefly introduced.

2.1 Fuzzy Sets

Let $X =_{df.} \{x_1, x_2, \ldots, x_n\} \subseteq \mathbf{R}$ be some finite set of elements (domain), then we shall call 'A' the fuzzy subset of X, if and only if: $A =_{df.} \{(x, \mu_A(x)) \mid x \in X\}$, where μ_A is a function that maps X onto the real unit interval [0, 1], i.e. μ_A: $X \rightarrow [0, 1]$. The function μ_A is also known as the *membership function* of the fuzzy set A, as its values represents the grade of membership of the elements of X to the fuzzy set A. Here the idea is that we can use membership functions, as characteristic functions (any crisp set can be defined by its characteristic function) for fuzzy, imprecisely described sets. Let A and B be two fuzzy subsets of X. Then the basic set operations: *union* and *intersection* of A and B, are defined as follows: $\mu_{A \cup B}(x) =_{df.} \max\{\mu_A(x), \mu_B(x)\}$, $\mu_{A \cap B}(x) =_{df.} \min\{\mu_A(x), \mu_B(x)\}$. Additionally, to combine fuzzy values, various t- and s-norms [2] may be applied.

2.2 Fuzzy Sugeno Integral

Sugeno introduced the theory of fuzzy measures and fuzzy integrals [12]. The fuzzy integral is based on the concept of fuzzy measure, which is a useful generalization of probability measure. The concept is effective in combining information in certain applications.

Consider a finite set $X = \{x_1, x_2, \ldots, x_n\}$ of sources of information. A *fuzzy measure* g is a real valued function $g : 2^X \rightarrow [0, 1]$, satisfying the following properties:

i. $g(\varnothing) = 0$; $g(X) = 1$;

ii. $g(A) \leq g(B)$, if $A \subseteq B$; $A, B \subseteq X$.

For a fuzzy measure g, let $g^i = g(\{x_i\})$. The mapping $x_i \rightarrow g^i$ is called a fuzzy density function. The fuzzy density value is interpreted (often subjective, supplied by experts) as the importance of the ith information source in determining an answer to the particular problem.

A fuzzy measure is a Sugeno measure (or g_λ—fuzzy measure) if it satisfies the following additional condition for some $\lambda > -1$:

For all $A, B \subseteq X$, $A \cap B = \varnothing$:

$$g_\lambda(A \cup B) = g_\lambda(A) + g_\lambda(B) + \lambda g_\lambda(A) g_\lambda(B) \tag{1}$$

The λ value can be calculated regarding to the condition $g(X) = 1$, using the following equation:

$$\lambda + 1 = \prod_{i=1}^{n} (1 + \lambda g^i) \tag{2}$$

The *Fuzzy Integral* (in the literature also called as *Sugeno Integral*) can be perceived as an aggregation operator. Let X be a set of information sources (e.g. features, sensors, classifiers) and let h: X → [0, 1], $h(x)$ denotes the confidence value delivered by element x (e.g., the class membership of data determined by a specific classifier). The fuzzy integral of h over A (A is a subset of X) with respect to the fuzzy measure g can be calculated as follows:

$$\int_A h(x) \circ g =_{df} \sup_{\alpha \in [0,1]} \{\alpha \wedge g(A \cap H_\alpha)\} \tag{3}$$

where $H_\alpha =_{df.} \{x \mid h(x) \geq \alpha\}$.

In a case of finite sets, suppose $h(x_1) \geq h(x_2) \geq \ldots \geq h(x_n)$ (if it is not true, then we can reorder the set of information sources X, so that this relation is true), then the Sugeno fuzzy integral takes the form:

$$S_g(h) =_{df} \vee_{i=1}^{n} \{h(x_i) \wedge g(H_i)\} \tag{4}$$

with $H_i = \{x_1, x_2, \ldots, x_i\}$.

The fuzzy Sugeno integral has been applied successfully in many different research areas as also in the field of bioinformatics [4]. This operator is very powerful in resolving multicriteria decision making problems, where the information that is combined is determined by experts.

2.3 Fuzzy Sugeno Integral—Interval based Interpretation

Let suppose, that the values of the function h (see Sect. 2.2 above) are not discrete numbers but intervals. This case would be more convenient, because it gives more flexibility of defining the confidence values for the information sources used. Theoretically, this refers to similar approach as the extension of classical fuzzy sets to fuzzy sets of type 2 [8, 9] which applications give better results in practice.

Therefore, assuming interval based integrant (function h) and using the well-known property of any continuous non-decreasing function defined on an interval, that it produces interval itself, we can extend the fuzzy Sugeno integral, as the following interval:

$$\int_A \overline{h}(x) \circ g =_{df} [\int_A \overline{h}^l(x) \circ g, \int_A \overline{h}^r(x) \circ g] \subseteq [0, 1], \tag{5}$$

where: $\overline{h}(x)$ represents the set of all intervals (the confidence values for each information source x_i), $\overline{h}^l(x)$ is the set of all lower endpoints of each interval and $\overline{h}^r(x)$ is the set of all upper endpoints of each interval.

If one numerical value is needed, as it is in our experiments, we propose to apply the following calculation:

$$Final\ value =_{\mathrm{df}} \frac{\int_A \overline{h}^l(x) \circ g + \int_A \overline{h}^r(x) \circ g}{2} \tag{6}$$

The calculation of the two Sugeno integrals with respect to the above assumptions, is exactly the same as described in Sect. 2.2.

3 The sEMG Signal Interpretation

In this section, we describe the standard interpretation of sEMG signal with respect to the rehabilitation process and next, we introduce our interpretation based on the fuzzy Sugeno integral.

3.1 The sEMG Interpretation in the Rehabilitation Process

Below, we present the standard approach of EMG signal processing and interpretation, applied in order to recognize possible progress of rehabilitation.

In our experiments, we used the Delsys® Bagnoli™ EMG System for signal acquisition, along with dedicated tool for signal processing and analyses EMG-Works® Software. The Bagnoli acquisition device and EMG signal processing software completely cover all the steps necessary for the processing of any raw sEMG signal, which are described below:

(a) acquisition of the signal,
(b) proper identification of the sEMG signal bandwidth—this is done by Band Pass filter with parameters: 15–500 Hz,
(c) elimination of the energy power grid utility frequency—this is done by Band Stop filter, depending on the country (in Poland: 50 Hz),
(d) RMS (Root Mean Square) filtering with time window, applying default options: window length 0.125 and window overlap 0.0625.

In Fig. 1, a sample raw sEMG signal (taken during repetitive rehabilitation exercises) and its processed form are shown, after applying steps (a)–(c) and (a)–(d).

From rehabilitation perspective, i.e. regarding to diagnostic procedures, the above signal processing, should be extend by the following analyses:

- Apply MVIC (maximum voluntary isometric contraction) technique—i.e. recognize the current maximum muscle contraction potential with respect to isometric

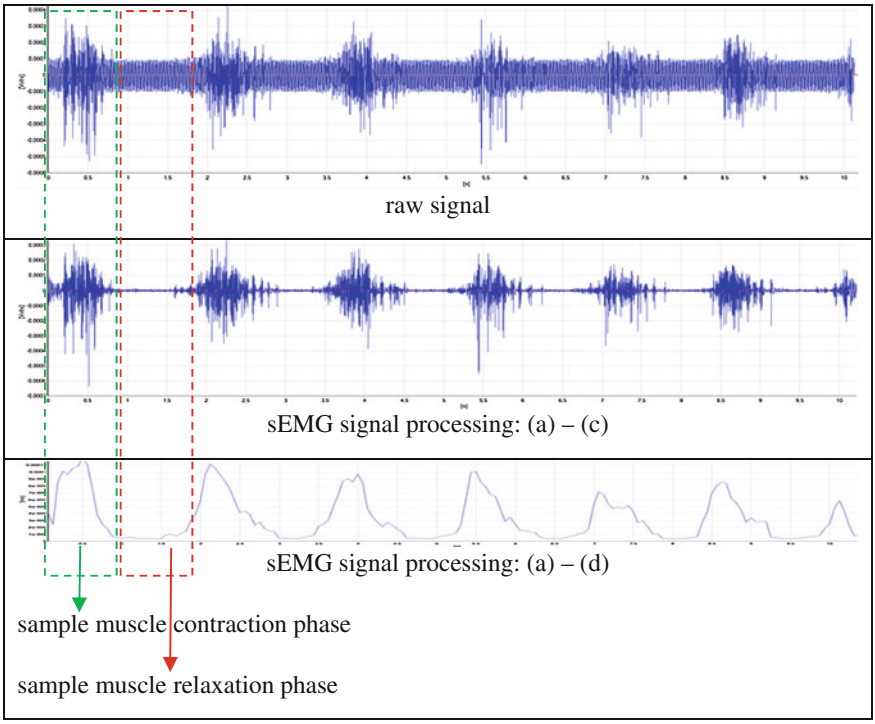


Fig. 1 Processing of a raw sEMG signal

contraction. This technique provides reference data which can be used to diagnose the current physical condition of patient muscles [1, 6, 10]. Usually, the MVIC is measured at the beginning of each rehabilitation session, which take place every few weeks (3–4 weeks, depending of the rehabilitation program).

- Refer performed rehabilitation exercises to the reference data, by signal amplitude analysis. The variations of the thus obtained values, may give information about possible muscle regeneration.

For better understanding of the MVIC technique, applied in the rehabilitation process, see Fig. 2 (data taken from a healthy person).

The tracing of the achieved maximum degree of muscle contraction with respect to the MVIC technique during rehabilitation exercises, gives a possibility to recognize possible rehabilitation progress. But there are some serious deficiencies:

(1) the measured values does not include any information about the current patient disposition;
(2) in practice, the procedure introduced is too subjective, especially the evaluation of the rehabilitation process (including all repeats of the corresponding rehabilitation exercise).

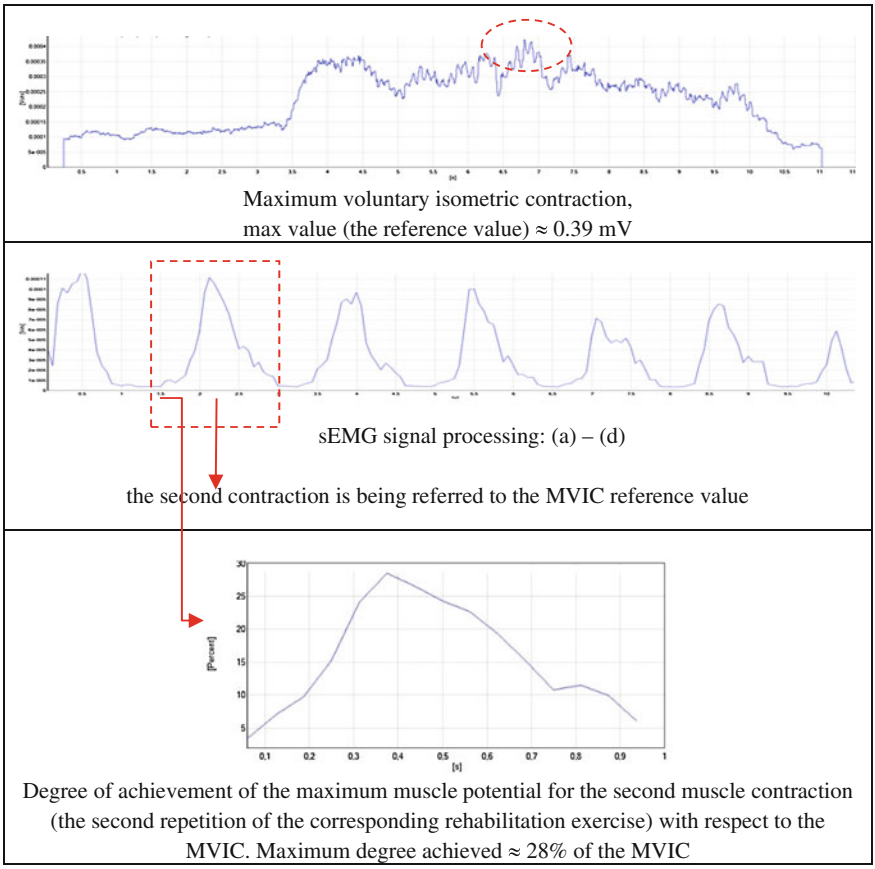


Fig. 2 Recognition of the physical condition of the patient with respect to the MVIC technique

3.2 sEMG Interpretation based on the Fuzzy Sugeno Integral

To eliminate the above mentioned disadvantages, we propose to apply the fuzzy Sugeno integral as an aggregation operator defined over the sEMG signal. The major assumptions are described below:

- Choose a set of representative muscle contractions for a corresponding rehabilitation process.
- Provide assessment of the performed exercises (assuming synchronization between the sEMG signal and corresponding video stream), by experts (physiotherapists). These values given by experts, in the range of 0–10, define the values of the fuzzy density function (see Sect. 2.2) assuming normalization. Such an assessment will involve not only physiotherapist's knowledge about the

correctness of the execution of the exercise but also will take into account the patient current physical disposition.

- Assume each muscle contraction chosen, as an information source and define the corresponding confidence values (see Example 1 below).
- Define the corresponding interval based fuzzy Sugeno integral (see Example 1 below).

For the clarity of the presentation, consider the following example.

Example 1 Representative muscle contractions

(a) Let consider the processed sEMG presented in Figs. 1 and 2, and let chose three representative muscle contractions (see Fig. 3).

(b) Let assume that the assessment given by experts for the chosen contractions are: $g^1 = 0.8$, $g^2 = 0.4$, $g^3 = 0.6$, we choose three contractions, because it is sufficient for proper diagnosis and the λ value (see Eq. 2), can be easily calculated as well.

(c) Apply the MVIC techniques for the chosen contractions (i.e. refer the values to the MVIC reference value, see Fig. 2).

(d) Define the corresponding confidence values of the information sources assumed (the chosen muscle contractions). We define these values as number intervals.

Note, to define interval we must ensure that functions defined over corresponding time intervals are non-decreasing, which is not preserved here. But, the polynomial character of changes of signal values is not important in the diagnostic process. Therefore, we can assume functions defined over corresponding time intervals, which takes as values the sorted (in ascending order) values of the corresponding sEMG sub signals (presenting each muscle contraction).

By applying the above assumption, to define corresponding intervals, it is enough to choose the minimum and the maximum values of the sub signals:

$$\overline{h} = \{[17.6\ \%,\ 28.2\ \%]_{contarction1}, [14.7\ \%,\ 24.7\ \%]_{contarction2}, [17.4\ \%,\ 25.6\ \%]_{contarction3}\}$$

or

$$\overline{h} = \{[0.176,\ 0.282]_{contarction1}, [0.147,\ 0.247]_{contarction2}, [0.174,\ 0.256]_{contarction3}\}$$

$$\overline{h}^l = \{0.176,\ 0.147,\ 0.174\} \quad \text{and} \quad \overline{h}^r = \{0.282,\ 0.247,\ 0.256\}$$

(e) Calculate the corresponding interval based fuzzy Sugeno integral:

$$\int_A \overline{h}(x) \circ g =_{df} [\int_A \overline{h}^l(x) \circ g,\ \int_A \overline{h}^r(x) \circ g] = [0.176,\ 0.282]$$

and therefore, we can calculate the final value (Eq. 6) = 0.229.

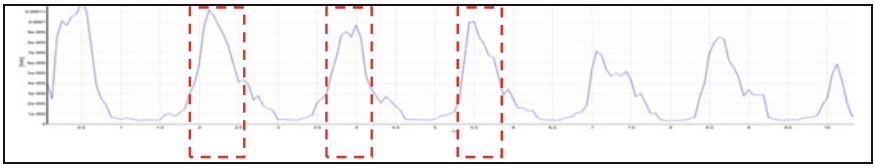


Fig. 3 Exemplary representative muscle contractions chosen by experts

The calculated numerical value characterizes the whole signal in sense of valuable rehabilitation information.

Using the above final value, it makes possible to observe very precisely progression of patients during their rehabilitation with respect to following MVIC measurements and physiotherapist knowledge.

4 Experiments

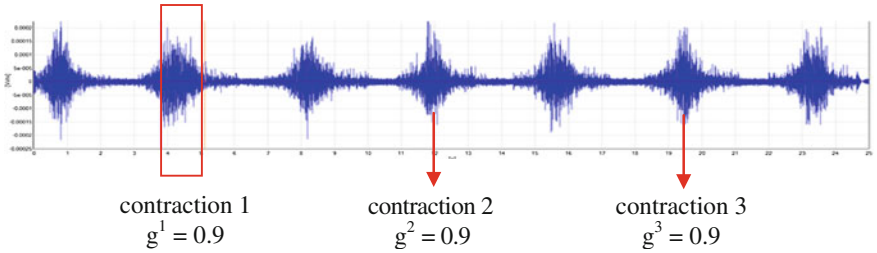
In the experiment below (Table 1 and Fig. 4), we present the distribution of the proposed 'final values' for a certain patient, with a spinal cord injury at the level of Th12, as an example of the possibility to trace possible progression of the patient during the rehabilitation process. If there are no any significant positive changes, this may imply the introduction of different rehabilitation procedure for the patient.

Additionally, the MVIC value was modified by a parameter w (mV), because the contractions measured during rehabilitation exercises were conducted with weights, which increased the sEMG amplitude in comparison with the maximum voluntary contraction. It is assumed, that the sEMG values during the rehabilitation exercises should not be greater than the MVIC in respectively short period of time. The value of the parameter was estimated during experiments with isometric contractions with supporting weights.

The results show a slight decrease in the efficiency of the patient (however within the value of 0.5, without large variations). This may be due to too frequent exercises. What more, objective conclusions can be made at a sufficiently long follow-up (e.g. on a scale of several months).

But the important issue is that on the basis of such charts it is possible to observe changes in patient's condition with high accuracy. What more, assuming respectively numerous observations of healthy people and calculating the above proposed value, it is possible to define a theoretical curve by nonlinear regression. The so defined curve could be used for comparison analysis with patients, with spinal cord injuries and therefore, it would contribute to the introduction of a new evaluation scale of the condition of patients with spinal cord injuries.

Table 1 Examining of *musculus quadriceps femoris* (lat.) of the left leg, five observations

Received MVIC reference value ≈ 0.045 mV + w ($w \approx 0.04$) = 0.085 mV
Observation 1—selected muscle contractions:

Contraction 1—sample frames extracted from corresponding synchronized, with the sEMG signal, video stream:

$\overline{h} = \{[0.111,\ 0.765]_{contarction1}, [0.123,\ 0.788]_{contarction2}, [0.153,\ 0.882]_{contarction3}\}$
Final value = 0.518
Observation 2:
$g^1 = 1$, $g^2 = 0.8$, $g^3 = 0.9$
$\overline{h} = \{[0.199,\ 0.901]_{contarction1}, [0.141,\ 0.724]_{contarction2}, [0.207,\ 0.73]_{contarction3}\}$
Final value = 0.554
Observation 3:
$g^1 = 0.8$, $g^2 = 1$, $g^3 = 1$
$\overline{h} = \{[0.248,\ 0.766]_{contarction1}, [0.228,\ 0.669]_{contarction2}, [0.263,\ 0.634]_{contarction3}\}$
Final value = 0.515
Observation 4:
$g^1 = 1$, $g^2 = 0.9$, $g^3 = 1$
$\overline{h} = \{[0.104,\ 0.623]_{contarction1}, [0.17,\ 0.623]_{contarction2}, [0.17,\ 0.647]_{contarction3}\}$
Final value = 0.409
Observation 5:
$g^1 = 0.8$, $g^2 = 0.8$, $g^3 = 0.8$
$\overline{h} = \{[0.163,\ 0.694]_{contarction1}, [0.171,\ 0.657]_{contarction2}, [0.194,\ 0.619]_{contarction3}\}$
Final value = 0.444

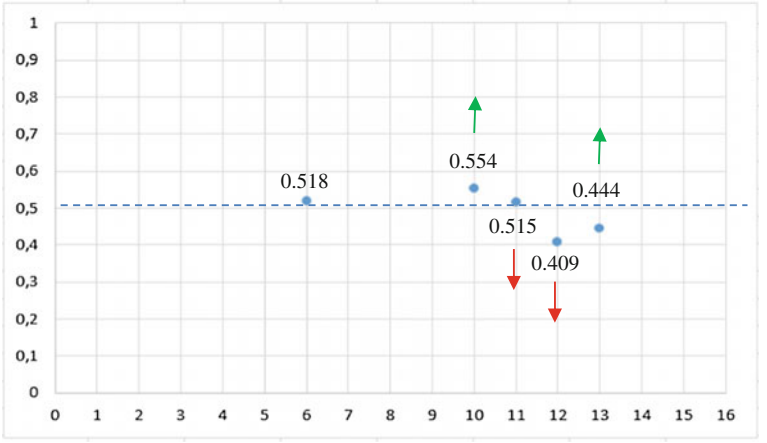


Fig. 4 The calculated values with respect to the subsequent measurements (the x-axis shows the day of the month wherein the measurement was performed)

5 Conclusion

In this paper we have proposed the use of corresponding mathematical model to combine both: subjective expert knowledge and sEMG signal data. The application of the proposed aggregator provides better interpretation of the signal.

What more, tracing the changes of the proposed numerical coefficient, provides a very significant knowledge in the rehabilitation process, which can be used to increase the quality of the process. The proposed value can be applied also to define theoretical curves, by nonlinear regression analysis, which could be used as benchmarks. Therefore, in our further work, we will expand our research on a large scale in order to confirm the proposed coefficient of rehabilitation progress.

Acknowledgment The authors would like to thank the Wroclaw City Council, for the opportunity to work with the Neuro-Rehabilitation Center for the Treatment of Spinal Cord Injuries 'Akson', under the 'Mozart' city programme 2014/2015.

References

1. Basmajian, J.V., De Luca, C.J.: Muscles Alive: Their functions revealed by electromyography. In: Zhou, P., Rymer, W.Z. Factors Governing the Form of the Relation between Muscle Force and the EMG: A Stimulation Study. J. Neurophysiol. **92**, 2878–2886 (2004)
2. Bronstein, I.N., Semendjajew, K.A., Musiol, G., Mühlig, H. Taschenbuch der Mathematik, p. 1258. Verlag Harri Deutsch, Frankfurt am Main (2001)
3. Cram, J.R., Kasman, G.: Introduction to Surface Electromyography. Aspen Publishing, Gaterburg, PA (1998)

4. Dong, X., Keller, M.J., Popescu, M., Bondugula, R.: Applications of fuzzy logic in bioinformatics. Adv. Bioinf. Comput. Biol. **9** (2008)
5. Ielpo, N., Calabrese, B., Cannataro, M., Palumbo, A., Ciliberti, S., Grillo, C., Iocco, M.: EMG-Miner: automatic acquisition and processing of electromyographic signals: first experimentation in a clinical C text for Gait disorders evaluation. In: IEEE 27th International Symposium on Computer-Based Medical Systems, pp. 441–446 (2014)
6. Kaplanis, P.A., Pattichis, C.S., Hadjileontiadis, L.J., Roberts, V.C.: Surface EMG analysis on normal subjects based on isometric voluntary contraction. J. Electromyograph. Kinesiol. **19**(1), 157–171 (2009)
7. Konrad, P.: The ABC of EMG: A Practical Introduction to Kinesiological Electromyography, (version 1.0). Noraxon Inc., US, April (2005)
8. Mendel, J.M.: Type-2 fuzzy sets and systems: an overview. IEEE Comput. Intell. Mag. **2**, 20–29 (2007)
9. Mendel, J.M.: Type-2 Fuzzy sets and systems: how to learn about them. IEEE SMC eNewslett. **27** (2009)
10. Merletti, R., Botter, A., Troiano, A., Merlo, E., Minetto, M.A.: Technology and instrumentation for detection and conditioning of the surface electromyographic signal: state of the art. Clin. Biomech. **24**, 122–134 (2009)
11. Segen, J.C.: Concise Dictionary of Modern Medicine. McGraw-Hill, New York (2006)
12. Sugeno, M.: Fuzzy Measures and Fuzzy Integrals—A Survey, pp. 89–102. North-Holland, Amsterdam (1977)
13. Zadeh, L.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)

Hybrid Classification of High-Dimensional Biomedical Tumour Datasets

Liliana Byczkowska-Lipinska and Agnieszka Wosiak

Abstract This paper concerns hybrid approach to classification of high-dimensional tumour data. The research presents a comparison of hybrid classification methods: bagging with Naive Bayes (NaiveBayes), IBk, J48 and SMO as base classifiers, random forest as a variant of bagging with a decision tree as a base classifier, boosting with NaiveBayes, SMO, IBk and J48 as base classifiers, and voting by all single classifiers using majority as a combination rule, as well as five single classification strategies, including k-nearest neighbours (IBk), J48, NaiveBayes, random tree and sequential minimal optimization algorithm for training support vector machines. The major conclusion drawn from the study was that hybrid classifiers has demonstrated its potential ability to accurately and efficiently classify both binary and multiclass high-dimensional sets of tumour specimens.

Keywords Hybrid classification · Ensemble classifiers · High-dimensional datasets · Tumour classification

1 Introduction

In general, tumour can be related to many pathways, various genes as well as exogenous factors, and can be regarded as the plague of our century. Even though researchers tend to thoroughly investigate the broad field of human cancer diagnosis, the general mechanism of tumour genesis and development has not been discovered yet. Moreover, treatment of advanced stage cancers is frequently not therapeutically effective. Therefore, early diagnosis of tumour is of great importance and

L. Byczkowska-Lipinska (✉)
University of Computer Sciences and Skills, ul. Rzgowska 17 a,
93-008 Lodz, Poland
e-mail: liliana.byczkowska-lipinska@p.lodz.pl

A. Wosiak
Institute of Information Technology, Lodz University of Technology,
ul. Wolczanska 215, 90-924 Lodz, Poland
e-mail: agnieszka.wosiak@p.lodz.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_21

many efforts are made in order to develop reliable therapies. Various subtypes of cancer demonstrate very different responses to therapy, suggesting that tumours are molecularly distinct entities. Therefore, accurate and reliable classification of cancer samples based on molecular signatures is significant for specifically tailored and eventually successful cancer treatment. Scientists continuously try to confront all the challenges related to the cancer diagnosis in order to support the efforts of clinicians.

Machine learning, embracing a variety of statistical, probabilistic and optimization methods, is especially well-suited to medical applications based on usually complex proteomic and genomic measurements. Although many methods have been developed recently, further improvement in classification accuracy is required before molecular-based machine learning techniques will be able to replace laborious histological approaches.

The aim of this paper is to create a comparison of single and hybrid classification techniques applied to the problem of binary and multi-class cancer classification. Even though both binary and multi-class sample classification have been studied extensively over the past few years [19, 21, 26], no exact solution has been discovered. Nowadays, still there is no perfect classification method as applied to high-dimensional yet small sample size microarray data. This research does not only constitute an independent contribution to the relevant literature, but also strive for finding a successful way to perform accurate classification of tumour specimens.

The rest of the paper is organized as follows. Section 2 corresponds to the medical background of this research and is followed by the literature overview. In Sect. 3 we describe the classification methods used in the experimental part of our research. Section 4 is dedicated to the experiments conducted on sample data and the results. Finally, in Sect. 5, the concluding remarks are discussed.

2 Problem Statement and Related Works

Nowadays, more than 200 types of cancer have already been identified [18]. In order to choose an appropriate and effective therapy, clinicians ought to be certain about the kind of cancer they are going to treat. Early cancer diagnosis is crucial for successful treatment. The most popular diagnostic technique used in medicine is still the analysis of biopsy specimens. Nevertheless, it yields only limited information and may not take into account many relevant tumour aspects. What is more, there is a wide spectrum in cancer morphology. A lot of tumours are atypical, missing morphological features that are usually useful for diagnosis [18, 26]. Therefore, cancer diagnosis is still a very challenging task [4].

Large number of researches have been discussed in the literature during the last few decades. This section pertains to a brief overview of microarray-based studies to provide a general idea of the current state of the art.

Wang et al. [20] applied two machine learning classifiers, kNN and SVM, in order to measure the classification accuracy of gene subsets subjected to the HBSA algorithm. HBSA-based classifier was built by taking advantage of the majority voting

strategy on the basis of the selected optimum gene subsets chosen by HBSA in order to significantly improve the stability of the classification performance. It appeared that HBSA-KNN is slightly superior to HBSA-SVM method in prediction accuracy, provided that the number of top-ranked genes selected using HBSA gene ranking approach is small enough. It was also shown that the HBSA-based methods are less computationally demanding than other exhaustive search strategies used for classification objectives.

Li et al. [12] focused on multiclass cancer classification as opposed to the majority of researchers dealing with a binary classification problem for the gene expression datasets. They highlighted that in general most of the real-world problems pertain to the multiclass classification. In their work, Li et al. [12] studied four multiclass decomposition techniques for SVM: one-versus-the rest, pairwise comparison as well as error-correcting output coding (ECOC) with two code generation strategies—random coding and exhaustive coding. The authors stated that SVM occurred to be the best classifier for tissue classification on the basis of gene expression data. The most meaningful conclusion drawn from the study of Li et al. [12] was that the prediction accuracy was dramatically lower in the case of the datasets consisting of a large number of classes (e.g. GCM).

Li et al. [11] focused on multicategory classification of the lung cancer microarray gene expression dataset. The 10-fold cross-validation technique was applied to evaluate the accuracy of SMO (sequential minimal optimization algorithm) classifier. The study of Li et al. [11] suggests that more in-depth research ought to be done in terms of used classifiers in order to produce models capable of dealing with high-dimensional, small sample size microarray data.

In [5], Elshazly et al. described two ensemble classifiers, namely Random Forest (RF) and Rotation Forest (ROT) for classification of high-dimensional cancer microarray data. Elshazly et al. [5] demonstrated that the Rotation Forest (ROT) performs considerably better when compared to other ensembles, chiefly owing to its high accuracy and diversity. The authors conducted their research on five human tumour-related sets of data. The evaluation was based on the 10-fold cross-validation method. It was shown that ROT classification outperforms other ensemble classifiers in terms of classification accuracy and diversity, providing at most 97 % accuracy on the breast cancer dataset.

The problem of classification of high-dimensional tumour datasets was also introduced by authors of this paper in [23]. However that research concerned at combining classification of high-dimensional biomedical data and feature selection and aimed at reducing dimensionality of the microarray data. The research presented a comparison of pairwise combinations of six classification strategies as well as seven attribute selection methods. The conclusion drawn from the study was that SVM-RFE feature selection technique combined with SMO classifier has demonstrated its potential ability to accurately and efficiently classify both binary and multiclass high-dimensional sets of tumour specimens.

3 Multiple Classification Methods

Classification of objects into pre-defined sets of categories or classes is a very important task in the field of machine learning. Identification of the common characteristics of subsets of data is one of the issues that researchers have to deal with. The main purpose of classification is to identify which set of categories a new observation belongs to. This is done on the basis of a training set consisting of instances that are already assigned to the known classes.

The main idea behind the multiple classification methodology is to weigh several individual pattern classifiers, and combine them in order to obtain a classifier that outperforms every one of them. In the literature, there are two terms that refer to multiple classification: "ensemble methods" and "hybrid classifiers". The first one usually refers to collections of models that are minor variants of the same basic model, whereas hybridization allows combining classifiers from different families.

3.1 Classifier Combination Rules

One of the most important problem to solve in hybrid classification is developing efficient combination rules for fusion of classifiers. In practice majority voting schemes are implemented in three main versions [10, 24]:

- unanimity, where the answer requires that all classifiers agree
- simple majority, where the answer is given by greater than half majority of classifiers
- plurality voting, taking the answer with the highest number of votes.

In literature the term majority voting usually refers to the last version—plurality votes. It can be written as

$$class(x) = \underset{c_i \in dom(y)}{arg\,max} \left(\sum_k g(y_k(x), c_i) \right) \tag{1}$$

where $y_k(x)$ is the classification of the k'th classifier and $g(y, c)$ is an indicator function defined as

$$g(y, c) = \begin{cases} 1, & y = c \\ 0, & y \neq c \end{cases} \tag{2}$$

3.2 Bagging and Boosting

Bagging and boosting are techniques that improve the accuracy of a classifier by generating a composite model that combines multiple classifiers all of which are derived from the same inducer.

The term bagging was introduced by Breiman in [1] as an acronym for Bootstrap AGGregatING. The idea of bagging is to create an ensemble classifiers based on bootstrap replicates of the training set. The classifier outputs are combined by the plurality vote [2].

A variant of bagging is a random forest [3]. It is a general class of ensemble building methods using a decision tree as the base classifier.

Boosting improves the performance of a weak learner as the method iteratively invokes a classifier on training data that is taken from various distributions. The classifiers are generated by resampling the training set and then combined into a single strong composite classifier. Boosting was based on an on-line learning algorithm called Hedge(β) [7]. This approach allocates weights to a set of strategies used to predict the outcome of a certain problem. The distribution is updated after each new outcome and strategies with the correct prediction receive higher weights while the impacts of the strategies with incorrect predictions are reduced.

One of the most popular ensemble algorithm that improves the simple boosting algorithm by an iterative process is AdaBoost (Adaptive Boosting). It was first introduced in [6]. The basic AdaBoost algorithm deals with binary classification. The classification of a new instance is performed according to the following equation:

$$class(x) = \underset{y \in dom(y)}{arg\ max} \left(\sum_{t : M_t(x) = y} \log \frac{1}{\beta_t} \right) \tag{3}$$

where $\beta_t = \frac{\varepsilon}{1-\varepsilon}$,

$\varepsilon_t = \sum_{i : M_t(x_i) \neq y_i} D_t(i)$,

$$D_{t+1}(i) = D_t(i) \cdot \begin{cases} \beta_t & M_t(x_i) = y_i \\ 1 & Otherwise \end{cases}$$

$D_1(i) = 1/m;\ i = 1, \ldots, m$

Bagging and boosting follow a voting approach to combine the outputs of different classifiers. However in boosting, each classifier is influenced by the performance of predecessors. To be specific, the new classifier pays more attention to classification errors that were done by the previously built classifiers. Besides in boosting, instances are chosen with a probability that is proportional to their weight, whereas in bagging, each instance is chosen with equal probability.

3.3 Hybrid Classification

Hybrid classifiers [8, 10, 16, 24] (also named multiple classifier systems) are designed to increase the accuracy of a single classifier by training several different classifiers and combining their decisions to output a single class label. The hybridization exploits the strength of each component [14].

The fundamentals of hybrid approach can be based on Wolpert's theorem ("no free lunch" theorem), which says that there is no single pattern recognition algorithm, which can be appropriate for all the classification tasks we deal with [22]. This theorem implies that a certain inducer will be successful only insofar its bias matches the characteristics of the application domain. Thus, given a certain application, the practitioner need to decide which inducer should be used. Using the multi-inducer obviate the need to try each one and simplifying the entire process [16].

Different levels can be distinguished to obtain hybrid classifier [25]:

- use different (distributed) data sources for training
- apply different data types and knowledge representations to merge them into one unified representation
- use trained models but take additional knowledge into consideration, e.g., additional constrains and
- use trained models to achieve the common decision based on combined classifier approaches.

For hybrid approach, the diversity is supposed to provide improved accuracy and classifier performance [20]. Therefore most works try to obtain maximum diversity by different means: introducing classifier heterogeneity, bootstrapping the training data, randomizing feature selection, randomizing subspace projections or boosting the data weights. Nevertheless, the diversity hypothesis has not been fully proven [20].

4 Experimental Results

The main objective of the experiments was to examine the accuracy of different hybrid classifiers in comparison to the results derived from application of single classification algorithms.

There were five different either binary or multi-class cancer microarray gene expression datasets used in the research: Colon Cancer Dataset (binary), Lung Cancer Dataset (binary), Lymphoma Dataset (multi-class), GCM Dataset (multi-class) and CNS Dataset (binary). The summary of all the sets of biomedical data is given in Table 1. The diversity of sample characteristics within datasets after performing multidimensional scaling is shown in Fig. 1. One may notice that the number of features is huge. The problem of feature selection methods in terms of these high-dimensional biomedical datasets was considered in [23] and therefore the experimental studies focus on classification stage of data analysis.

In the first step of the experiments, single classification algorithms were applied. Five approaches were considered: k-nearest neighbours (IBk where $k = 5$), J48, Naïve Bayes (NaiveBayes), random tree and sequential minimal optimization algorithm for training support vector machines (SMO). Multi-class problems (Lymphoma and GCM datasets) using SMO algorithm were solved using pairwise classification (1-vs-1) [9]. The results of classification are presented in Table 2. In order

Table 1 Datasets description

Dataset name	No. of samples	No. of features	No of classes
CNS	60	7129	2
Colon	62	2000	2
Lung	181	12600	2
Lymphoma	96	4026	11
GCM	192	16063	14

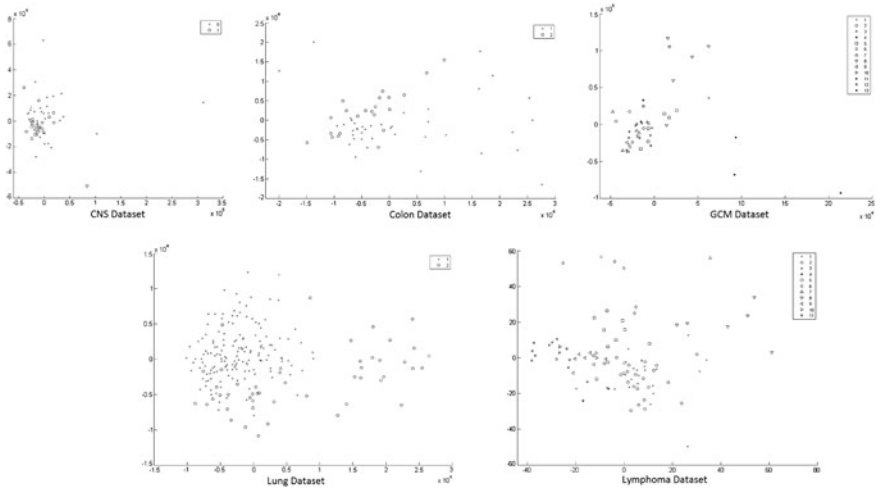


Fig. 1 Visualisation of characteristics for datasets

to assess the performance of various classification methods, following comparison criteria have been used: accuracy, sensitivity, precision and root mean square error. To assess the accuracy of classifiers, 10-fold cross-validation was used, however the original division into test set and training set was maintained for GCM dataset.

In the case of classification conducted by single classifiers, SMO significantly outperformed other classifiers in terms of classification accuracy. The best results attained 95 % for multiclass Lymphoma dataset and 85 % for binary Colon dataset. There is no denying that the multiclass GCM classification provided the worst results, achieving up to 54 % for SMO classifier. The average accuracy of single classification approach attained 64 %.

Next step of the experiments concerned performing classification using hybrid classifiers. Different combinations were applied:

- bagging with IBk, Naive Bayes, J48 and SMO as base classifiers
- random forest as a variant of bagging with a decision tree as a base classifier
- boosting with IBk, Naive Bayes, J48 and SMO as base classifiers
- voting by all single classifiers using majority as a combination rule.

Table 2 Single classification results

Dataset	Criterion	IBk	J48	NaiveBayes	RandomTree	SMO
CNS	ACC	56.667	58.333	61.667	63.333	68.333
	PREC	0.576	0.560	0.630	0.651	0.674
	SENS	0.567	0.583	0.617	0.633	0.683
	RMSE	0.647	0.643	0.619	0.606	0.562
Colon	ACC	77.420	82.258	53.226	74.194	85.484
	PREC	0.774	0.820	0.596	0.735	0.854
	SENS	0.774	0.823	0.532	0.742	0.855
	RMSE	0.467	0.414	0.684	0.508	0.381
GCM	ACC	45.652	52.174	52.174	34.783	54.348
	PREC	0.554	0.553	0.566	0.387	0.599
	SENS	0.457	0.522	0.522	0.348	0.543
	RMSE	0.267	0.255	0.261	0.305	0.247
Lung	ACC	56.667	58.333	61.667	63.333	68.333
	PREC	0.576	0.560	0.630	0.651	0.674
	SENS	0.567	0.583	0.617	0.633	0.683
	RMSE	0.647	0.643	0.619	0.606	0.562
Lymphoma	ACC	72.917	67.708	76.042	39.583	94.791
	PREC	0.767	0.680	0.737	0.381	0.919
	SENS	0.729	0.677	0.760	0.396	0.948
	RMSE	0.212	0.237	0.209	0.321	0.265

The results of hybrid classifications are shown in Table 3. The visualisation of accuracy values is presented in Fig. 2. One can see, that in most cases hybrid classifications outperformed the corresponding single classifiers in terms of classification accuracy, as well as precision, sensitivity and root mean square error. Even SMO—one of the best single classification method—when boosted, improved the accuracy to more than 90 % for Lung and Lymphoma datasets. In case of majority voting method, the hybrid classification accuracy was highly improved for binary Lung dataset (38 % of improvement) and equalled the highest single classification accuracy for CNS and Colon datasets. However in case of multi-class GCM dataset, the hybrid classification accuracy appeared to be worse than single classifiers (up to 60 % instances classified correctly).

To sum up classification results, in Table 4 there are best accuracies attained for single and hybrid classification approaches. In four out of five cases, hybrid classifiers appeared to give at least as high accuracy as the best single classifier.

In regards to relevant scientific publications we can conclude that our results confirmed the ones reported in those papers. The authors of [17] used a heterogeneous ensemble classifier to improve the detection accuracy of major construction materials such as concrete, steel, and wood on construction sites. It was shown that better classification performance was achieved by the ensemble classifier than by the single

Table 3 Hybrid classification results

	Bagging with base				RF	Boosting with base				MV
	IBk	J48	NB	SMO		IBk	J48	NB	SMO	
CNS										
ACC	56.667	60.000	66.667	61.667	60.000	56.667	56.667	60.000	68.333	68.333
PREC	0.567	0.537	0.660	0.596	0.582	0.576	0.536	0.552	0.674	0.660
SENS	0.567	0.600	0.667	0.617	0.600	0.567	0.567	0.600	0.683	0.667
RMSE	0.568	0.494	0.528	0.516	0.456	0.647	0.653	0.621	0.562	0.577
Colon										
ACC	77.419	82.258	53.226	79.032	64.516	77.419	77.419	80.645	85.484	85.484
PREC	0.769	0.820	0.574	0.787	0.621	0.774	0.769	0.812	0.854	0.854
SENS	0.774	0.823	0.532	0.790	0.645	0.774	0.774	0.806	0.855	0.855
RMSE	0.414	0.383	0.536	0.381	0.443	0.467	0.455	0.427	0.381	0.381
GCM										
ACC	41.304	47.826	45.652	54.348	54.349	45.652	45.652	47.826	54.348	60.870
PREC	0.527	0.517	0.494	0.598	0.554	0.576	0.424	0.475	0.599	0.620
SENS	0.413	0.478	0.457	0.544	0.457	0.567	0.457	0.478	0.543	0.609
RMSE	0.237	0.205	0.218	0.234	0.218	0.267	0.250	0.269	0.247	0.236
Lung										
ACC	90.640	92.118	82.759	94.089	88.177	89.655	92.118	85.222	95.567	94.581
PREC	0.909	0.919	0.816	0.941	0.865	0.898	0.920	0.875	0.956	0.947
SENS	0.906	0.921	0.828	0.941	0.882	0.897	0.921	0.852	0.956	0.946
RMSE	0.181	0.153	0.237	0.317	0.195	0.201	0.170	0.220	0.319	0.142
Lymphoma										
ACC	73.958	79.167	54.167	89.583	94.375	72.917	71.875	76.042	94.792	91.667
PREC	0.803	0.775	0.567	0.860	0.814	0.767	0.684	0.737	0.919	0.891
SENS	0.740	0.792	0.542	0.896	0.844	0.729	0.719	0.760	0.948	0.917
RMSE	0.203	0.183	0.250	0.265	0.198	0.212	0.212	0.209	0.265	0.123

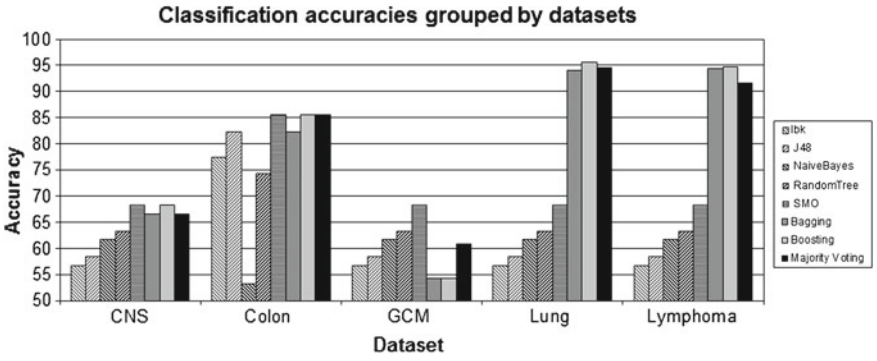


Fig. 2 Visualisation of classification accuracies

Table 4 Comparison of best accuracies for single and hybrid classifiers

Dataset	Accuracy for single classifier	Accuracy for hybrid classifier	Accuracy difference
CNS	68.333	68.333	0.000
Colon	85.484	85.484	0.000
GCM	68.333	60.870	-7.463
Lung	68.333	95.567	+27.234
Lymphoma	68.333	94.792	+26.459

classifiers applied (SVN, ANN, C4.5, NB, LR, kNN). In [15], the authors presented a combination of simple classifiers to classify microarray data samples. The study, carried out using six different microarray datasets from two different diseases, revealed that the proposed model obtained an improvement in the accuracy when compared with classical approaches (kNN, NBS, SVM and C4.5). For the multi-class scenario there was no superior model. Nevertheless, hybrid classifier stands out in two of the three available multi-class datasets. In [13] an evolutionary computation based classifier subset selection process was presented to construct a multiple classifier system. The obtained experimental results were comparable to other possibilities for multi-classifier construction (voting, hierarchical classifiers, etc.) and improved single classifiers.

5 Conclusions

Classification of high-dimensional biomedical datasets is regarded as a challenging task, requiring extremely high accuracy and as short computational time as possible. All of the already reported results concerning microarray data suggest that multi-class classification issues are typically more difficult than the binary ones. Therefore research on finding the most appropriate methods for a multi-class classification are conducted and often succeed in new approaches.

By comparing hybrid classifiers algorithms and single classification methods, it was demonstrated that the hybrid strategy resulted in the most satisfactory outcomes and confirmed other up-to-date researches on multiple classifier systems. In order to specifically tailor the hybrid approach so that the high classification accuracy is to be obtained regardless of the set of input data, one has to take into account a variety of aspects. These circumstances contribute to the difficulties related to finding the optimal and universal classification method, specifically tailored to handle biomedical datasets.

It was successfully proven that the hybrid classifiers outperform other classification methods in the majority of cases, regardless of the input dataset used for the purpose of training the model. In four out of five cases, hybrid classifiers appeared to

give at least as high accuracy as the best single classifier. The results of this study can constitute an independent contribution to the relevant state-of-art scientific papers, as most of them showed a significant improvement of hybrid classification over single classifiers [13, 15, 17].

High-throughput technologies used in genomic and proteomic studies provide the opportunity to analyze numerous biological samples. This results in high amounts of multivariate data corresponding to different biological aspects. Having a possibility to look at thousands of genes is beneficial but it may increase the risk of overfitting the data, especially when small number of samples is available. Therefore, feature selection is an important stage of data analysis in the case of biological studies where sample size is usually limited and it was a subject of our research described in [23].

Future studies ought to involve other algorithms and strategies as well, especially those ones specifically tailored to deal with the most challenging multi-class cancer classification tasks. In order to find the optimal solutions, other combinations of various classifiers should be investigated in depth.

References

1. Breiman, L.: Bagging Predictors. Technical Report 421, Department of Statistics, University of California, Berkeley (1994)
2. Breiman, L.: Bagging predictors. Mach. Learn. **26**(2), 123–140 (1996)
3. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001)
4. Dziomdziora A.: Comparative Study of Feature Selection Methods for High-dimensional Biomedical Datasets (Masters thesis supervised by A. Wosiak), Łódz Unversity of Technology, Łódz, Poland (2014)
5. Elshazly, H.I., Elkorany, A.M., Hassanien, A.E., Azar, A.T.: Ensemble classifiers for biomedical data: performance evaluation. In: Proceedings of the 9th International Conference on Computer Engineering & Systems (ICCES), pp. 184–189 (2013)
6. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Proceedings of the Thirteenth International Conference in Machine Learning, pp. 325–332 (1996)
7. Freund, Y., Schapire, R.E.: A decisiontheoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. **55**(1), 119–139 (1997)
8. Galar, M., Fernández, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Trans. Syst. Man, Cybern. Part C: Appl. Rev. **42**(4), 463–484 (2012). doi:10.1109/TSMCC.2011.2161285
9. Hastie, T., Tibshirani, R.: Classification by pairwise coupling. Ann. Stat. **26**(2), 451–471 (1998)
10. Kuncheva, L.I.: Combining pattern classifiers, methods and algorithms. Wiley, Hoboken (2004)
11. Li, X., Lu, H., Wang, M.: A Hybrid gene selection method for multi-category tumor classification using microarray data. Int. J. Bioautomation **17**(4), 249–258 (2013)
12. Li, T., Zhang, C., Ogihara, M.: A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. Bioinformatics **20**(15), 2429–2437 (2004)
13. Mendialdua, I., Arruti, A., Jauregi, E., Lazkano, E., Sierra, B.: Classifier subset selection to construct multi-classifiers by means of estimation of distribution algorithms. Neurocomputing **157**, 46–60 (2015)

14. Michalski, R.S., Tecuci, G.: Machine learning: a multistrategy approach. J. Morgan Kaufmann (1994)
15. Reboiro-Jato, M., Díaz, F., Glez-Peña, D., Fdez-Riverola, F.: A novel ensemble of classifiers that use biological relevant gene sets for microarray classification. Appl. Soft Comput. **17**, 117–126 (2014)
16. Rokach, L.: Pattern classification using ensemble methods. World Scientific Publishing Co. Inc, River Edge (2010)
17. Son, H., Kim, C., Hwang, N., Kim, C., Kang, Y.: Classification of major construction materials in construction environments using ensemble classifiers. Adv. Eng. Inf. **28**(1), 1–10 (2014)
18. Tiwari, M.: Microarrays and cancer diagnosis. J. Cancer Res. Ther. **8**(1), 3–10 (2012)
19. Wang, X., Gotoh, O.: A robust gene selection method for microarray-based cancer classification. Cancer Inf. **9**, 15–30 (2010)
20. Wang, S.L., Li, X.L., Fang, J.: Finding minimum gene subsets with heuristic breadth-first search algorithm for robust tumour classification. BMC Bioinformatics **13**(178), 1–26 (2012)
21. Wang, Y., Tetko, I.V., Hall, M.A., Frank, E., Facius, A., Mayer, K.F.: Gene selection from microarray data for cancer classification—a machine learning approach. Comput. Biol. Chem. **29**, 37–46 (2005)
22. Wolpert, D.H.: The supervised learning no-free-lunch. In: 6th Online World Conference on Theorems, Soft Computing in Industrial Applications, pp. 25–42 (2001)
23. Wosiak, A., Dziomdziora, A.: On Pairwise combinations of feature selection and classification methods for high-dimensional tumour biomedical datasets. Schedae Informaticae, 24 (Ahead of Print) (2015). doi:10.4467/20838476SI.15.005.3027
24. Wozniak, M., Graña, M., Corchado, E.: A survey of multiple classifier systems as hybrid systems. Inf. Fusion pp. 3–17 (2014). doi:10.1016/j.inffus.2013.04.006
25. Wozniak, M., Kasprzak, A.: Data stream classification using classifier ensemble. Schedae Informaticae 23 (Ahead of Print) (2014). doi:10.4467/20838476SI.14.002.3019
26. Zhang, X.W., Yap, J.L., Wei, D., Chen, F., Danchin, A.: Molecular diagnosis of human cancer type by gene expression profiles and independent component analysis. Eur. J. Hum. Genet. **13**(12), 1303–1311 (2005)

Part V
Artificial Intelligence

Learning and Memory Processes in Autonomous Agents Using an Intelligent System of Decision-Making

Zdzisław Kowalczuk, Michał Czubenko and Wojciech Jędruch

Abstract This paper analyzes functions and structures of the memory that is an indispensable part of an Intelligent System of Decision-making (ISD), developed as a universal engine for autonomous robotics. A simplified way of processing and coding information in human cognitive processes is modelled and adopted for the use in autonomous systems. Based on such a knowledge structure, an artificial model of reality representation and a model of human memory (using, in particular, the concept of Long-Term Memory) are discussed. Finally, the paper presents a way of rearranging the system memory and modelling the processes of learning.

Keywords Fuzzy systems · Cognitive robotics · Machine learning · Autonomous agents · Decision making · Knowledge representation

1 Introduction

At any level of abstraction, in order to consider the learning processes designed for the use in autonomous agents, one should first conceive and implement a method of representing data/information (such as features, objects, etc.) in such systems. Within the contemporary learning processes, two principal stages can be distinguished. The first stage concerns the classical machine learning [30] or other new powerful methods, like the deep learning [2], used to initially recognize objects. The second stage of the learning process is founded on a dynamic organization of data representation of previously recognized objects, and is used to perceive and recognize new

Z. Kowalczuk (✉) · M. Czubenko · W. Jędruch
Faculty of Electronics, Telecommunications and Informatics,
Gdańsk University of Technology, Narutowicza 11/12, Gdańsk, Poland
e-mail: kova@pg.gda.pl

M. Czubenko
e-mail: m.czubenko@gmail.com

W. Jędruch
e-mail: wjed@eti.pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_22

instances of the objects. Both above mentioned stages/areas are covered by a (complex/integrated) cognitive methodology which implements models of human information processing [10, 19–22]. Such pattern/feature recognition schemes make a powerful method for a great number of benchmark problems and different domain applications. Especially, the idea of feature creation and selection are important for the effectiveness of recognition problems [25].

Basically, with the use of great simplification and approximation, the methods of deep learning can be associated with the process of human cognition. Using this point of view, in both domains, human and computer, there are the following two actions: feature recognition and analysis. Thus, in general, such a modus operandi can be applied in solving various engineering tasks ranging from arts and architecture to computer science and robotics [8].

In this paper, we present arguments and some solutions for data representation using the results of psychology. Such types of results were applied during the process of constructing ISD – an Intelligent System of Decision-making [19], as well as in one of its implementation in the form of an *xDriver* system [10], an extensively autonomous agent, controlling a model of a car on a virtual highway.

2 Coding Methods in Cognitive Perception

There is a belief that knowledge is represented in human minds analogously to a corresponding sector of reality (*realism* [37]). To cover multiplicity, however, the realism theory would need a large number of copies of a single object in mind. An opposite direction to the realism theory is called *constructivism* [11, 17], which assumes the possibility of the existing variety of possible interpretations (views) for any single object. Furthermore, constructivism also states that learning is an active & constructive process. The learner in those terms constructs information actively by creating its own representations of the objective reality. Any new piece of information is linked to prior knowledge (this means the mental representations are subjective).

On the other hand, it is clear that the early psychology was focused on handling images in human minds, which was only a pure theory, not verifiable. Later on, cognitive psychologists assumed that images are not appropriate representation in the case of problem solving, and created symbolic representations, like semantic networks [32]. The symbolic representation and its formal transformation ideas known as the symbol manipulation paradigm dominated in artificial intelligence for many years [16].

We thus assume suitability of constructivism and its applicability for agent or robotic purposes. This means that we are going to represent objects existing in reality, along with various view-points (use-cases, associations) and references (relations, links, external-associations), which can be considered as generalized features (impressions, associations) in the semantic network approach. Note that such features play different roles in different contexts (emotions, locations, environment, etc.). It is thus clear that knowledge representation cannot be a pure projection of reality.

As already mentioned, cognitive science, especially the cognitive psychology provides a good basis for designing of learning mechanisms for autonomous agents. Cognitive psychology describes active processing of data (ranging from simple stimuli to a complex memory content) necessary in making decision concerning the issue of choosing an adequate practical reaction. Moreover, the cognitive science is based on a 'computational metaphor' that data processing in human minds may be partly, or completely [34], represented in a similar way as computational manipulations in computer systems.

From the neurology point of view the cognitive process may be seen as a perception-action cycle [13]. A crucial fact is that stimuli are processed in several layers of the system. A simplified schematic comparison of the discussed methods is presented in Fig. 1. The data/stimuli generated by the system's environment are passed over to the first stage of processing (sensory perception/primary sensory/input layer), where they are grouped and tentatively processing. Next, certain features are extracted: the impressions in the cognitive path (shapes, lines, spectrum), the features in the deep learning path, and straight features in the neurological path. The final stage is to merge those features into sensible groups (discoveries). Thereafter the decision concerning a desired reaction, can be worked out [2].

The above mentioned part of cognitive processes is referred to as cognitive perception. There are many works which intend to design a robotic system relying on cognitive perception, especially based on the perception-action cycle [9]. The system of perception appears to be the most crucial element for modern robotic systems. In this context, a *semantic* representation of the environment is very important, also [3]. An associated problem that also needs to be resolved, is the memory representation (*syntax*) of a single object.

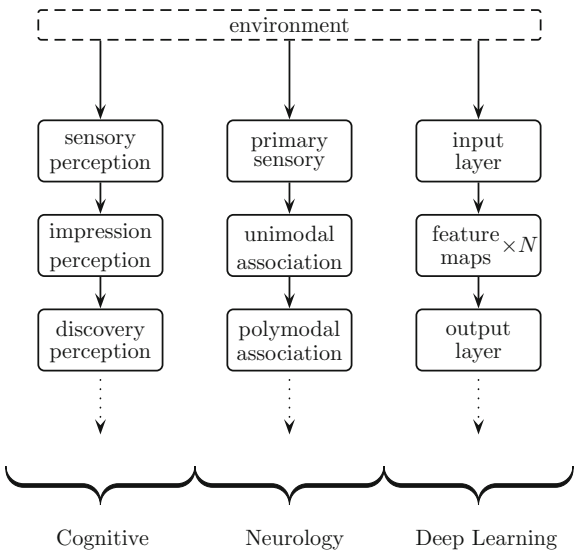


Fig. 1 Comparison of the cognitive, neurological and deep learning approaches to the stimuli processing

Note that constructivism and newer psychological studies lead to the following guidelines for the knowledge acquisition in humans:

- process the material semantically (as the knowledge is organized semantically, knowledge acquisition is optimized when the learner focuses on the meaning of new material)
- process and retrieve information frequently (retrieving or self-producing information can be contrasted with simply reading or copying it)
- connect new information to the prior knowledge
- create cognitive procedures (procedural knowledge is better retained and more easily accessed).

which also constitute a reasonable base both for the learner and for the learning applied in our developed ISD system.

2.1 Syntax of Memorized Objects

There are several theories categorizing mental representations of reality in cognitive psychology [33]. Allan Paivio [36] distinguishes *logogens*, memorized entities which can be put into words or labels (not displaceable or non-rotatable), and *imagens*, graphical items or other images of reality (which can have different placement or position and pose or view). In our interpretation such a label (logogen) will be equivalent to a notion (idea or concept). Imagens and logogens are appropriately connected using internal associations (references, links, or orderings). The theory of Paivio, also understood as dual-coding [39], is graphically interpreted in Fig. 2b.

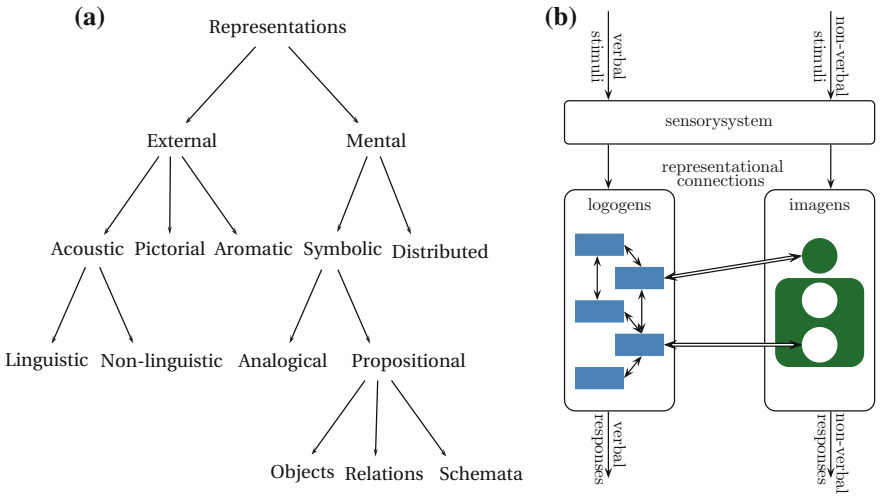


Fig. 2 The models of data representation **a** Types of representations in [12] **b** The dual-coding theory [36]

In accordance with [12], generalized objects (both external or physical and mental or abstract) can be classified with the use of a tree (Fig. 2a). It is clear that with this representation only some aspects of the environment can be described. We consider the symbolic representations to be most important within the right ('mental') branch. In our treatment, the 'analogical' part concerns images and imagens, whereas 'propositional' objects are composed of relations, (sheer) objects and schemata (all of them can be logogens).

Another theory of memorizing objects is proposed by Stephen Kosslyn [18], who brings-up the fact that physical objects exists in 3D, and can be represented in a mental three-dimensional space by means of a 3D mental medium, where information takes the form of a cloud of points [5]. Such a medium has three most important properties of geometry:

- it may show spatial properties of an object, as well as some specific relation to other objects in the space (e.g. proportions)
- it may be zoomed, rotated and translated
- its details can blur in a time course.

According to Kosslyn, the long term memory (LTM) [19], used for keeping memorized objects (generalized: physical and abstract), is composed of only two basic types of elements: images and propositions, which take the form of *files*, describing object properties [12]. Image files contain all information necessary to visualize an image in the medium. In addition, in our solution, the points of the cloud (medium) can have additional features (like an individual color).

Experiments [41] show that spatial representation in human minds is in daily use. Moreover, this representation can be sufficiently accurate for many purposes, and can also be faster than pure semantic representation. Clearly, people quickly recognize pictures (without verbal interpretation), as compared to a virtual scene described by the text (with the use of free interpretation). There are several qualities which make the imageries special [40]:

- Effortless structure: Visual features are more important than the other sensations (see the methods of presentation, first impression, etc.), as a vital result of the human evolution. The structure of imagery results from visual perception (see the highly specialized decoders, like face detectors).
- Determinism: Basically, at any moment, there is only a single set of structures applied in visual perception. Sometimes, however rarely, there are pictures that can be interpreted in two ways (illusions). Nevertheless, the imagery is much more clear and deterministic than semantic (text) description.
- Perception-action coupling: Visual perception is inextricably linked with levels of reactions. This feature is very important in case of obtaining new data about the object of interest. Without vision, an agent, robot, or human cannot easily interfere with the surrounding space and objects.
- Pre-interpretation: A bottom-up process of perception, allows us to pre-interpret certain visual stimuli. This phenomena is very important in case of emergency (needing a fast reacting).

From the engineering point of view, there are also other issues to be solved. An important one is what to do with mathematics? Recall that the dual-coding theory states that all objects can be represented as imagens or logogens. The same concerns the representations of abstract operations (like square root). Another problem is how to evaluate them: In a most simple and practical case, some procedural techniques can be applied here. McCloskey and Macaruso [27] propose using numerical methods, based on the language of mathematics [7]. Today, this kind of representation allows also for a faster manipulation of numbers.

In summary, the system of human mind representation is triple coded (logogen, imagen and numerics). Any object may be described using a semantic verbal description (e.g. 'a horse has 4 legs'). Such a verbal representation is also connected to a non-verbal one, being a pack of 'graphical' features of this object (e.g. shape, color, texture, etc.). It is common that the non-verbal representation includes also a sample of this object (its image as a whole). Both verbal and non-verbal representations can also use numbers (simplifying the description).

Recently, a whole branch of knowledge representation and automatic extraction of knowledge, called data mining, has been developed within computer science. Structures called ontologies have been created in networks communication. They are defined as a formal explicit specification of a shared conceptualization [6]. In general, knowledge can be represented as the first-order logic and description logic [6], Minski's frames [29], or semantic networks [35]. There are also methods of uncertain knowledge representation, like neural networks, fuzzy systems, or Bayesian networks [38, 44].

3 Knowledge Representation in ISD

The Intelligent System of Decision-making (ISD), developed by Kowalczuk and Czubenko [19, 20] is based on human psychology. The ISD already includes some basic forms and elements taken from the representation theories. They result from cognitive psychology that deals with the process of perceiving (Fig. 4). On a similar basis, the ISD has elementary mechanisms of memory [22], especially the semantic one [28], suitably modelled and implemented. The semantic memory is responsible for storing abstract data, commonly known as knowledge. In a most simple view, the idea can be described in short as the stimuli coded into impressions, which are recoded into discoveries.

An impression is a simple feature of an object (like color, texture, etc.). Impression results from the activity of ascending paths, extending from receptors [14]. There are two groups of impressions [22]:

- primal: physical features, connected with real features of an object, and
- secondary: features associated with an object in the agent's memory, like feelings (sub-emotional context), certain composed impressions (like: 'this object may be friendly'), or associations to all kinds of needs.

Impressions (especially the primal ones) are recognized by several mechanisms of human mind [23] modelled in the resulting impressions-extraction diagram of ISD shown in Fig. 4. Those features describe the perceived objects written in using the mental system representations. According to the Kosslyn theory, impressions may be parts of the images *files* [18] mentioned before and corresponding to the discovery concept of ISD. The impressions, as primitives, are stored in a highly unwriteable piece of LTM (Long Term Memory). Thus, though new impressions may appear, it is a rare case.

New impressions of the secondary type can describe complex features of objects which are connected to the evaluative function of the human mind (for instance, something may be evaluated as 'awful'). Moreover, the secondary impressions may refer to some motivative factors—like emotions (a sub-emotional context) or 'objective' needs. Both motivational factors: emotions and needs, are interpreted, stored, and applied with the use of fuzzy representation [42]. They appear to be natural for the purpose of decision-making and communicating with human. The fuzzy-set approach to modelling emotions seems to be more natural than any crisp mathematical model.

Discovery is an abstract representation of an object. In most cases it can be imprecise, subjective and incomplete. It contains a list of impressions associated with the object (both primal and secondary), an object label (logogen), and connections to other discoveries (relations). Relations between discoveries can be presented in the form of propositions. For our robotic purposes the applied concept of discoveries and impressions covers the previously mentioned theories to a great extent.

The idea of discovery, in a certain sense, may be generalized to the concept of (abstract) *a-discoveries* [22]. Such a generalization allows the system to create an abstract part of semantic memory (a part of LTM) in a tree form. The roots of the tree represent most abstract objects (eg. 'animals'), whereas the leaves are usually certain instances of objects (e.g. a black horse). Consecutive instances of leaves are also possible (e.g. 'Black Star' can be a dark horse). They are called *i-discoveries* (discovery instances). The basic relationship between a-discovery is the relation of inheritance or succession (e.g. 'horse is an animal'), whereas the most common relationship between i-discoveries is the relation of belonging or affiliation (e.g. 'Silver Star belongs to Joe'). Moreover, the i-discovery also inherits from its parent (e.g. 'Silver Star is a horse'). The basic structure of the discoveries is shown in Fig. 3. To optimize the searching problem, an activity level or forgetting [24] has been added to the concept of i-discovery. In such a way most important discoveries can be recognized faster.

3.1 Path of Information

The path between the stimuli and a reaction proceeds through several stages [20]. The first stage of data gathering is sensory perception (Fig. 4), which receives the stimuli from certain receptors responsible for senses (sight, hearing, taste, smell,

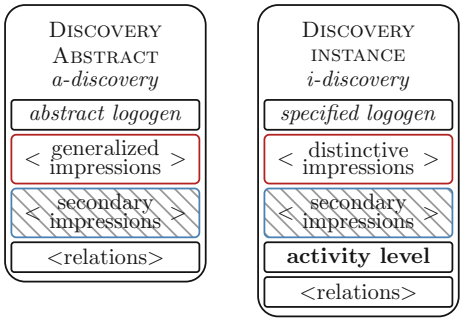


Fig. 3 Structures of discoveries [22]

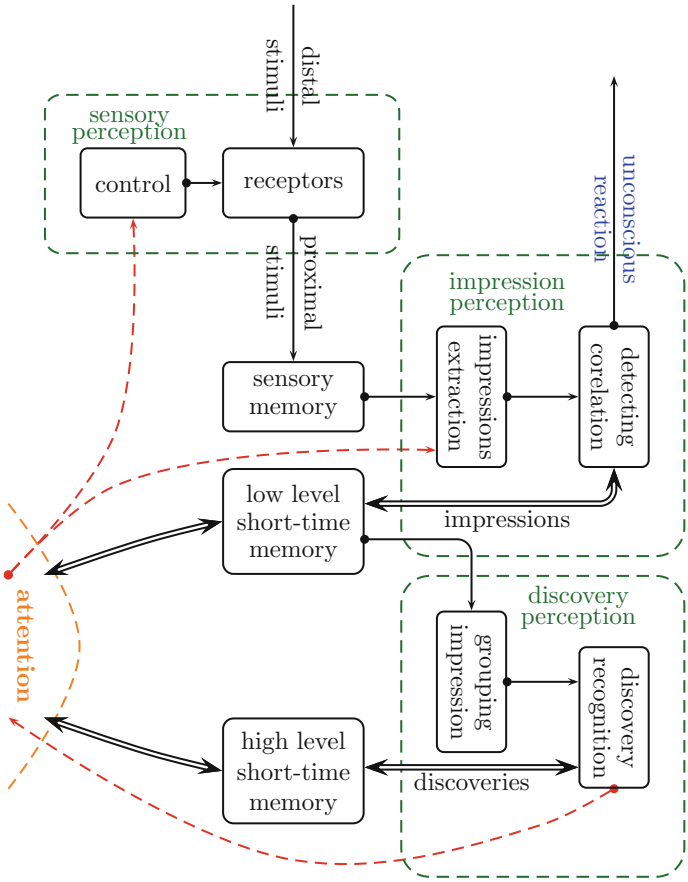


Fig. 4 The ISD perception process

touch, balance, temperature, kinesthetic, pain) [33]. The sensory perception has two phases, related with proximal (an image on receptors) and distal (real objects represented as a stimulus) stimuli [26]. Proximal stimuli are written into a sensory memory, the receptors can focus on recognizing certain concrete features (impressions). The process of impression recognition is led according to some algorithms, which recognize shapes, colors, contours, textures, etc. This process is called the bottom-up recognition of impressions. Such a process can also have a top-down form, where some mechanisms can search the memory for the secondary impressions, and compare them with the current proximal stimuli.

3.2 Discoveries and Their Tecognition

Recognized impressions are grouped according to their localization in the space of performed perception. A sensible assembly of such impressions is called discovery. Grouping due to localization is not sufficient for the recognition. The process of discovery recognition involves searching the a-discoveries tree (a part of the semantic memory [28]), and matching the impressions. A discovery is recognized based on a best match performed using a certain threshold (e.g. 90 % of agreement). Otherwise a signal FNO (Fetch New Object) is generated, and a new pack of discoveries is fetched and taken for comparison. After few trials the recognized discoveries are moved to a current scene memory, and the unrecognized ones are passed through to the second stage of searching and comparing.

In the second stage of the discovery recognition process, the agent compares unknown discoveries with some old, previously unrecognized objects (u-discoveries). In the case of matching (or similarity, in practice), the count number of this temporary discovery increases. When no match is found, the perception process generates a signal RNuO (Remember a New unrecognized Object) and creates a new unrecognized temporary discovery. At the end of this stage, when the count number of certain unrecognized discovery achieves a set level, this discovery is treated as an i-discovery and is moved to the semantic memory (yielding an associated signal of CNO, Create New Object). During this process, the analyzed discovery is consciously given a certain temporary name created by the thinking process. Presumably, some associated impressions of currently analyzed discoveries can be contradictory. In such cases, the discovery is dropped, issuing a respective signal DO, Drop the Object.

4 Autistic Thinking and Learning

The cognitive process of thinking can be divided into two parts: realistic and autistic [4]. Realistic thinking is designed to achieve a particular goal, whereas autistic thinking is the formation of loose associations, imaginary hypotheses, etc. [33].

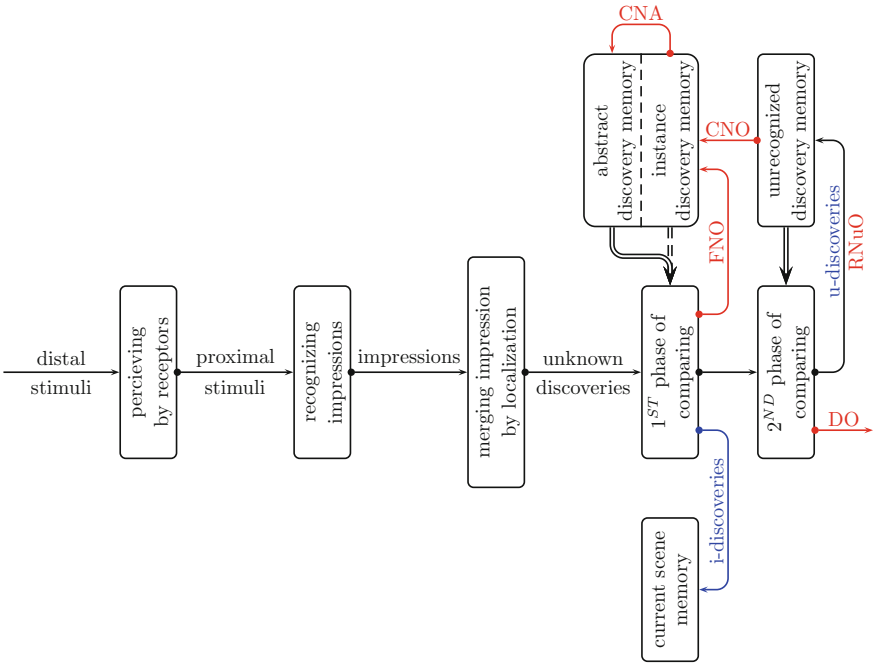


Fig. 5 Information path in the ISD system

It also allows us to generalize the newly discovered i-objects, hence creating new a-discoveries (Create New Abstract on Fig. 5).

On the other hand, according to the Baron's theory [1], there are two groups of thinking: search and inference. The search group refers to the goal-directed (realistic) thinking, and may concern three classes of items: goals (using Pareto optimization, for instance), possibilities (considering various paths leading to the goal) and premisses (for gaining the goals). Inference consists in formulating conclusions on the basis of known prerequisites (e.g., generalization of the purposes, the formulation of proofs, etc.). In addition, by inference, you can redefine poorly defined problems (or change of the goals), or close open problems [33]. Whereas the goals and possibilities refer to the goal-directed (active) thinking, the searching of evidences is rather associated to the autistic thinking.

Such a categorization allows us to present the autistic thinking as the one which manage the process of learning, especially its memorizing part. During autistic thinking, the mind can restructure the semantic memory (a part of Long Term Memory), add new relations, create new abstracts, and merge the old ones. Autistic thinking is also responsible for creating new impressions and accepting them to the set of the identified impressions, whereas realistic thinking allows, in the context of mental representation, to add new relations and impressions (from the accepted ones) to the currently known objects, as well as to name new discoveries.

4.1 Process of Restructuring the Memory

The process of restructuring the memory is concerned about both the abstract and the instance part of semantic memory. It performs the following four actions:

- creating a new a-discovery, by generalization of several i-discoveries, only when they are highly similar
- merging the old a-discoveries, in the case of high (fuzzy) similarity
- finding and evaluating new relations between the defined a-discoveries and i-discoveries
- creating new relation prototypes, based on a knowledge earlier obtained (e.g. a 'person' may be 'employee' of other 'person').

The best way to describe the relations taking place in the human mind and to implement them in computer science, lies in using linguistics, and fuzzy logic [43]. There are a lot of different relations grouped by their type [15]. Currently, in the ISD system, the relation prototypes includes only inheritance, possession, membership, instances (especially, the relations between the i-discovery and a-discovery), and feeling. This list can be further expanded in future.

5 Conclusions

The presented introduction to modeling of object representation in the memory of autonomous robots, is based on the psychological review of mental representations. The developed model uses the latest results from computer science, especially the ones concerning knowledge representation and fuzzy systems. As the experiments show [10], this combination seems appropriate for autonomous robot systems founded on cognitive models.

In future works we are going to implement the representation system described above on the ISD platform for new practical autonomous humanoid robot applications (like the NAO robot created by Aldebaran Robotics), which are to be based on the OWL-DL language and equipped with a number of mechanisms of reasoning (to mention the F-Logic system, and, especially, Flora-2).

Appendix: Memory Model in xDriver Simulations

This appendix presents a short note on an implementation of the ISD system, in which the memory model has been programmed in the language XML. Simulations were performed in JAVA, using several additional libraries (like guava, fuzzy110a, JFreeChart).

The aim of this study was to test the ISD system in the task of control of the car on a highway.

The *xDriver* system based on the ISD was used for the task of simulated car driving [10]. The principal aim of the *xDriver* simulation study was to test the ISD system (described partly here, and partly in our earlier publications [19–21]) in the task of autonomous driving.

Since the ISD system is a result of a thorough modelling of human psychology (generally known), there are concepts somehow similar to ISD which can be found in the literature (e.g. [31]). There are, however, no reports of a system which would model the human psychology for such practical purposes as autonomously driving, for instance. It is worth noting that our simulations have proven that the *xDriver* system may behave on the road as an inexperienced human driver [10].

The *xDriver* was tested only in virtual simulations, and real objects or images were not recognized. Nevertheless, it worked properly based on its own developed semantic memory. The applied concept of memory allowed the *xDriver* to 'understand' the traffic regulations, to make its own decisions, and, in consequence, to ride in accordance with the rules. In particular, the following types of abstract objects were applied in the *xDriver*: lane, horizontal road sign, vertical road sign, virtual static objects (trees, houses, etc.), and virtual dynamic objects (such as other cars, pedestrians, etc.).

In general, the semantic memory of the *xDriver* consists of abstract objects, which are represented as a-discoveries in the ISD system (Fig. 3), and instances of them (i-discoveries). Figure 6 shows a sample of the semantic memory of the *xDriver* that consists of i-discoveries and a-discoveries. Boxes represent discoveries. The i-discoveries are marked by the encircled 'i' in the left corner of the box, the other discoveries are the a-discoveries. A-discoveries have generalized features (impressions, in the first sub-box, shown in red color), and their own labels (in italics). I-discoveries have distinctive impressions, which distinguish them from the a-discoveries, and allows us to process the impact of them on the cognitive decision making performed by ISD. The i-discoveries have a need context and contain also secondary impressions (marked with 'e:'), which are vital for the emotional context of the *xDriver*. The emotional context (as fuzzy membership functions) takes values between 0 and 1 and is applied in a procedure [20, 21] weighting all actually perceived objects. The need context of an object can also assume a normalized integer value (see the *pedestrian* object in Fig. 6), interpreted as a simple shift (increment or decrement) in a need unfulfillment function (or a more involved procedural estimation) for the indicated need of the analysed object. The value of the shift can change during the learning process, which allows for improvement of the semantic knowledge.

The first sub-box (marked additionally in red color in Fig. 6) in each discovery definition describes here a single impression or a set of them (representing an object or discovery). The road signs (1–3) listed above (and defined as single impressions) can only trigger a kind of procedural evaluation of the actual needs of the *xDriver*. Whereas all the virtual objects (4–5) mentioned above can have their specific (not null) emotional or need contexts specified in the second sub-box (marked additionally in blue), that is, they can influence the needs and emotions of the xDriver. When the *xDriver* sees a sign 'speed limit 50' (in short 'ban 50' in Fig. 6, for instance, an evaluation of the indicated *xDriver*'s need starts, in which the *xDriver* re-estimates

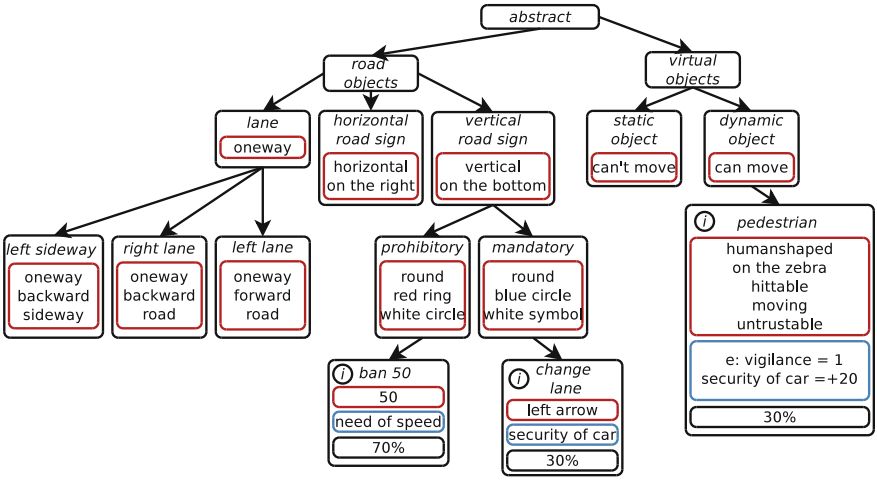


Fig. 6 Part of the semantic memory of the xDriver

its *need of speed*, in a procedure which takes into account the resulting offset of the actual speed of the car over the imposed restriction (50 km/h). Note that, as compared to the road signs, the virtual objects have a number of defining impressions or features, including impressions like 'hittable' or 'accidentable', which point to certain possible consequences of interacting with these objects (for the composed driving system). The third sub-box (marked in black in Fig. 6) represents an activity level of the i-discovery (in terms of a currently updated relative frequency of its appearance).

Based on the above description of the characteristics of the ISD system, it is clear that the set of the presented objects can easily describe a straight road environment with which the *xDriver* is able to interact.

References

1. Baron, J.: Thinking and deciding, 4th edn. Cambridge University Press, Cambridge (2008)
2. Bengio, Y.: Learning deep architectures for AI. Found. Trends Mach. Learn. **2**(1), 1–127 (2009)
3. Benjamin, D.P., Lonsdale, D., Lyons, D., Patel, S.: Using cognitive semantics to integrate perception and motion in a behavior-based robot. In: ECSIS Symposium on Learning and Adaptive Behaviors for Robotic Systems, pp. 77–82. IEEE (2008)
4. Berlyne, D.E.: Structure and direction in thinking. Wiley, New York (1965)
5. Beserra Gomes, R., Ferreira da Silva, B.M., Rocha, L.K.D.M., Aroca, R.V., Velho, L.C.P.R., Gonçalves, L.M.G.: Efficient 3D object recognition using foveated point clouds. Comput. Graph. **37**(5), 496–508 (2013)
6. Brachman, R., Levesque, H.: Knowledge representation and reasoning. The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann, San Francisco (2004)
7. Campbell, J.I.D.: Handbook of mathematical cognition. Psychology Press, New York (2005)
8. Caravantes, A., Galán, R.: Generic educational knowledge representation for adaptive and cognitive systems. Educ. Technol. Soci. **14**(3), 252–266 (2011)

9. Cutsuridis, V.: Cognitive models of the perception-action cycle: a view from the brain. In: The 2013 International Joint Conference on Neural Networks, pp. 1–8. IEEE (2013)
10. Czubenko, M., Ordys, A., Kowalczuk, Z.: Autonomous driver based on intelligent system of decision-making. Cogn. Comput. 1–13 (2015). doi:10.1007/s12559-015-9320-5
11. How we think. D.C. Heath & Company, New York (1910)
12. Eysneck, M.W., Keane, M.T.: Cognitive psychology: a student's handbook, 4th edn. Psychology Press, Hove (2000)
13. Fuster, J.M.: Upper processing stages of the perception-action cycle. Trends Cogn. Sci. **8**(4), 5–143 (2004)
14. Hebb, D.O.: Textbook of psychology. Saunders, Philadelphia (1958)
15. Hjørland, B.: Semantics and knowledge organization. In: Cronin, B. (ed.) Annual Review of Information Science and Technology, chap. 8, pp. 367–405. Wiley (2007)
16. The allure of machinic life: cybernetics, artificial life, and the new AI. A Bradford Book, Chester (2008)
17. Kant, I.: What does it mean to orient oneself in thinking? Berlinische Monatschrift **8**, 304–330 (1786)
18. Kosslyn, S.M.: Image and brain: the resolution of the imagery debate. A Bradford Book, Chester (1996)
19. Kowalczuk, Z., Czubenko, M.: Interactive cognitive-behavioural decision making system. In: Rutkowski, L. (ed.) Artifical Intelligence and Soft Computing, Lecture Notes in Artificial Intelligence, vol. 6114 (II), pp. 516–523. Springer, Berlin (2010)
20. Kowalczuk, Z., Czubenko, M.: Intelligent decision-making system for autonomous robots. Int. J. Appl. Math. Compu. Sci. **21**(4), 621–635 (2011)
21. Kowalczuk, Z., Czubenko, M.: Computational model of emotions dedicated to intelligent decision systems (in Polish: xEmotion – Obliczeniowy Model Emocji Dedykowany dla Inteligentnych Systemów Decyzyjnych). Pomiary Automatyka Robotyka **2**(17), 60–65 (2013)
22. Kowalczuk, Z., Czubenko, M.: Cognitive memory for intelligent systems of decision-making, based on human psychology. In: Korbicz, J., Kowal, M. (eds.) Intelligent Systems in Technical and Medical Diagnostics, Advances in Intelligent Systems and Computing, vol. 230, chap. Cognitive, pp. 379–389. Springer, Berlin (2014)
23. Linhares, A., Chada, D.M.: What is the nature of the mind's pattern-recognition process? New Ideas Psychol. **31**(2), 108–121 (2013)
24. Loftus, G.R.: Evaluating forgetting curves. J. Exp. Psychol. Learn. Mem. Cogn. **2**, 397–406 (1985)
25. Martinez, H.P., Bengio, Y., Yannakakis, G.N.: Learning deep physiological models of affect. IEEE Comput. Int. Mag. **8**(2), 20–33 (2013)
26. Maruszewski, T.: Cognitive Psychology (in Polish: Psychologia Poznania). Gdańskie Wydawnictwo Psychologiczne, Gdańsk (2001)
27. McCloskey, M., Macaruso, P.: Representing and using numerical information. Am. Psychol. **50**(5), 351–363 (1995)
28. McRae, K., Jones, M.: Semantic memory. In: Reisberg, D. (ed.) The Oxford Handbook of Cognitive Psychology. Oxford University Press, Oxford, New York (2013)
29. Minsky, M.: A Framework for representing knowledge. In: Winston, P.H. (ed.) The Psychology of Computer Vision. McGraw-Hill, New York (1975)
30. Mitchell, T.M.: Machine learning. McGraw Hill, New York (2008)
31. Muhlestein, M.: Counterfactuals, Computation, and consciousness. Cogn. Comput. **5**(1), 99–105 (2012)
32. Newell, A., Simon, H.A.: Human problem solving. Prentice-Hall, Englewood Cliffs (1972)
33. Nęcka, E., Orzechowski, J., Szymura, B.: Cognitive Psychology (in Polish: Psychologia Poznawcza). PWN, Warszawa (2008)
34. Ormerod, T.: Human cognition and programming. In: Hoc, J.M., Green, T.R.G., Samurçay, R., Gilmore, D.J. (eds.) Psychology of Programming, pp. 63–82. European Association of Cognitive Ergonomics and Academic Press (1990)

35. Ouyang, X.: Construction of the paradigmatic semantic network based on cognition. In: International Conference on Asian Language Processing, pp. 122–125. IEEE (2010)
36. Paivio, A.: Mental Representations: A Dual Coding Approach. Oxford University Press, Oxford (1990)
37. Plato: The Republic. Penguin Books Limited, London (2012)
38. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall, Upper Saddle River (1995)
39. Saltzman, E., Pick, H.L.: Modes of Perceiving and Processing Information. Lawrence Erlbaum Associates, Hillsdale (1978)
40. Schwartz, D.L., Heiser, J.: Spatial representations and imagery in learning. In: Sawyer, R.K. (ed.) The Cambridge Handbook of the Learning Sciences, chap. 17, pp. 283–298. Cambridge University Press, Cambridge (2005)
41. Standing, L.: Learning 10,000 pictures. Q. J. Exp. Psychol. **25**(2), 22–207 (1973)
42. Wu, Q., Miao, C.: Modeling curiosity-related emotions for virtual peer learners. IEEE Comput. Intell. Mag. **8**(2), 50–62 (2013)
43. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
44. Żurada, J., Barski, M., Jędruch, W.: Artificial Neural Networks (in Polish: Sztuczne Sieci Neuronowe). Wydawnictwo Naukowe PWN, Warszawa (1996)

Solving Highly-Dimensional Multi-Objective Optimization Problems by Means of Genetic Gender

Tomasz Białaszewski and Zdzisław Kowalczuk

Abstract Paper presents a computational optimization study using a genetic gender approach for solving multi-objective optimization problems of detection observers. In this methodology the information about an individual gender of all the considered solutions is applied for the purpose of making distinction between different groups of objectives. This information is drawn out of the fitness of individuals and applied during a current parental crossover in the performed evolutionary multi-objective optimization (EMO) processes.

Keywords Genetic algorithms · Multi-objectives · Pareto-optimality · Observers

1 Introduction

The evolutionary algorithms have found a great number of applications [2, 4, 9, 11, 13, 14]. The significance of optimization methods emulating the evolution of biological systems is approved by their usefulness and effectiveness. The features of biological systems are their ability to re-generate, perform self-control, re-product and adapt to the variable conditions of existence. On a similar basis, we also require that analogous features characterize systems designed in terms of adaptation,

T. Białaszewski · Z. Kowalczuk (✉)
Faculty of Electronics Telecommunication and Informatics, Gdańsk University of Technology, Narutowicza, 11/12, 80-233 Gdańsk, Poland
e-mail: kova@pg.gda.pl

T. Białaszewski
e-mail: tombiala@pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_23

optimality, immunity etc. We can thus formulate tasks concerning the optimality of solutions and their robustness to small changes in environmental conditions and parameters and to disturbances that allow obtaining effective and reliable engineering systems.

It appears clear that in design processes it is essential to globally optimize several objectives at the same time [2, 11, 13, 25]. Such multi-objective optimization tasks are though difficult to be performed, as the notion of optimality is not obvious.

In order to join a number of objectives together, it is necessary to define relations between partial objectives being considered (what can be done by setting weights). For the purpose of solving such optimality problems, various methods are proposed, including: (i) weighted profits, (ii) distance functions, (iii) sequential inequalities, (iv) lexicographic ordering, or (v) ranking with the use of Pareto-optimality [2, 3, 10, 11].

In contrast to the above, the fifth method, using ranking with respect to a measure of Pareto-optimality, avoids the arbitrary weighting of objectives. Instead, a constructive classification of solutions is applied that takes into account particular objectives more objectively. Though this idea of optimality does not give any hints as to the choice of a single solution from a generated set of Pareto-optimal solutions, the designer has always a chance to make an independent judgment of all the 'best' offers. The above-mentioned methods of qualifying the multi-objective solutions can be easily utilized in the (GA) genetic algorithms [10, 11].

It appears that there are two basic reasons and consequences of the evolution of gender in nature: (a) long-term—in search of new mutations, beneficial improvements, and adaptation, and (b) short-term—for the genetic variation, significant in terms of resistance to parasites (bacteria and viruses) [13].

In the above context, this paper presents a new method, referred to as the genetic-gender approach (GGA), initiated in [10], of solving multi-objective optimization problems by evolutionary search, with the use of Pareto-optimal ranking, where the information about a degree of membership to a given gender [19] is attributed to each newly generated solution under examination. This information is utilized in the process of parental crossover, in which only individuals of different genders are allowed to create their offspring.

A practical control-design example of the application of the proposed approach to multi-objective synthesis problems, complex FDI design issue [7, 10, 17] of a detection observer [2, 8, 14, 15, 20], which serve as a principal element in the procedures of detecting and isolating faults is considered. This design technique is illustrated with the use of a benchmark problem based on a ship propulsion system [6]. The complex engineering design effect given in the form of a robust optimal detection observer [8, 17, 20] demonstrates both prospective usefulness and effectiveness of the proposed GGA/genetic optimization method. Such an optimal system design tool allows designing systems, which perform their basic task, while having sufficient sensitivity (to errors in sensors and actuators, for instance) and simultaneously showing robustness (to certain modeling uncertainties).

2 Evolutionary Multi-Objective Optimization

From a formal viewpoint, a multi-objective optimization task [10, 13] can be defined by means of the following m-dimensional vector of objective functions

$$\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}) \quad f_2(\boldsymbol{x}) \quad \ldots \quad f_m(\boldsymbol{x})]^{\mathrm{T}} \in \mathbf{R}^m \tag{1}$$

where $\boldsymbol{x} = [x_1 \quad x_2 \quad \ldots \quad x_n]^{\mathrm{T}} \in \mathbf{R}^n$ means a vector of the parameters searched for, and $f_j(\boldsymbol{x}), j = 1, 2, \ldots, m$ denotes a given partial-objective function. Assuming, for simplicity, that all co-ordinates of the criterion vector (1) are profit functions, the analyzed multi-objective optimization task can be formulated as a multi-profit maximization task without constrains:

$$\max_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \tag{2}$$

In the last decades a number of evolutionary multi-objective optimization (EMO) methods [1, 3–7, 11, 12] have been proposed for solving multi-objective problems in multi-dimensional spaces, including: VEGA (Vector Evaluated Genetic Algorithm), LOGA (Lexicographic Ordering Genetic Algorithm), VOES (Vector Optimized Evolution Strategy), WBGA (Weight-Based GA), MOGA (Multi-Objective Genetic Algorithm), NPGA/NPGA2 (Niched Pareto Genetic Algorithm), NSGA/NSGA2 (Nondominated Sorting Genetic Algorithm), DPGA (Distance-based Pareto Genetic Algorithm), TDGA (Termo-Dynamical Genetic Algorithm), MOMGA (Multi-Objective Messy Genetic Algorithms), PAES (Pareto Archived, Evolutionary Strategies), PESA/PESA2 (Pareto Envelop-based Selection Algorithm), SPEA/ SPEA2 (Strength Pareto Evolutionary Algorithm), μGA/μGA2 (Micro GA-MOEA), MOBOA (Multi-Objective Bayesian Optimization Algorithm), GGA (Genetic-Gender Algorithm). A short description of the above algorithms can be found in [13].

3 The Genetic Gender Approach

In contrast to the great number of various mechanisms of generating new solutions and decision making processes proposed and implemented in genetic and evolutionary algorithms, there are only few isolated attempts of applying some sexual categories in the genetic reproduction mechanisms known from the literature: MSGA (Multi-Sexual Genetic Algorithm) [16], G-GA (Gendered Genetic Algorithm) [18], GAGS (Genetic Algorithm with Gendered Selection) [22], HRAGA (Adaptive Genetic Algorithm simulating Human Reproduction mode) [24], GASS (Genetic Algorithm with Sexual Selection) [19], GSGA (Gender Separation with Genetic Algorithms) [23], MOGASS (Multi-Objective Genetic Algorithms with

Sexual Selection) [21]. Characteristics of the gender multi-objective evolutionary algorithms can be found in [13].

In contrast to the other approaches [16, 18, 19, 21–24], we present our novel method for solving multi-criteria optimization tasks based on the EMO approach with the use of a genetic-gender idea, which consists in assigning each individual a specific gender relating to a degree of its membership to certain sub-populations associated with respective sub-criteria. The gendered individuals are submitted to a 'natural' crossover process of mating. To deliver a proper view on the GGA concept and the resulting GGA algorithm [10, 13], basic issues and mechanisms of the multi-objective optimization machinery applied (incl. rank estimation and global optimality concepts) are first explained, before the reasoning and details of the gender approach are given.

3.1 Concept of Genetic Gender

In nature the gender division of a species appears to differentiate individuals with reference to reproductive functions. Taking this into account, our concept of an artificial genetic gender (GG) consists in dividing the set of objective functions into subsets (Fig. 1), each of which has an attributed genetic gender Xj, $j = 1, 2, \ldots, s$, and portrays an assumed partial-scope suboptimality value of a certain utility in the designer's interpretation [10, 13]: GENDERS = {X1, X2, ... Xj, ..., Xs}. For example, GENDERS = {XX, XY} symbolizes the native two-element set of the XX and XY chromosomes (certain 'species' attributes), associated with some distinguishable characteristics of the individuals estimated in terms of the objectives considered [13].

In a particular context, a gender (a set) and a variant (a label assigned to each gender) are associated with both a subset of criteria and a subset of individuals (best in this context). Basically, the applied division of the set of attributes should be rational and distinctive to achieve the realization of the global optimization/survival goal as a result of the synergy of different genders. Yet, obtaining such an effect is not critical.

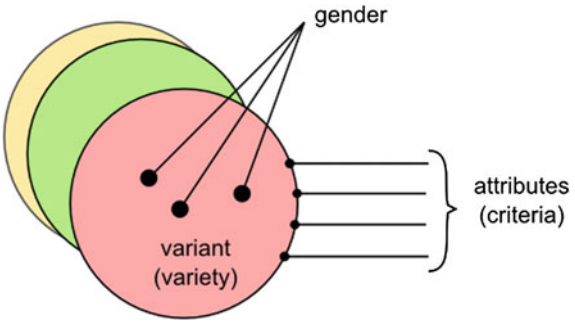


Fig. 1. The overall concept of the GGA method: a gender set with a variant (the name of the gender) and its attributes

In this way, one gender set (X*j*) can embrace objectives of a 'similar' character that are only in a kind of internal (secondary) rivalry in terms of an approximately equal meaning to the user from some pertinent point of view. Such an assortment can thus effectively discharge the designer from the cumbersome task of isolating a single solution from amongst all the Pareto-sub-optimal ones (obtained in the course of multi-objective optimization).

In contrast, different gender sets (X*j*) can express various groups of 'interests' that are difficult to be judged by the user in advance. In general, this division can be used to represent an external (primary) rivalry, which is not simple to be resolved. On a common basis, in such cases the best method is to use the notion of Pareto-optimality.

Thus in our consequent approach, we propose using the mechanism of the gender allotment during the whole computational evolution for the purpose of creating a few parental pools of different genders and generating new offspring by mating only apparently dissimilar individuals. In spite of such a fractional perspective, the notion of the set-fitting P-sub-optimality appears to be entirely clear and sufficient [10, 13].

The vector of the profit functions (1) can therefore be divided into s-subvectors

$$\boldsymbol{f}(\boldsymbol{x}) = [\boldsymbol{f}_1(\boldsymbol{x}) \quad \boldsymbol{f}_2(\boldsymbol{x}) \quad \ldots \quad \boldsymbol{f}_s(\boldsymbol{x})]^{\mathrm{T}} \in \mathbf{R}^m \tag{3}$$

where $\boldsymbol{f}_j(\boldsymbol{x})^{\mathrm{T}} \in \mathbf{R}^{m_i}$ describes the j-th subvector $(j = 1, 2, \ldots, s)$ defining the genetic-gender perspective, which by means of some measure will be used to specify the genetic-gender set of individuals all labeled by X*j*. Within each of these sets, P-sub-optimality-based ranking of individuals is applied. In effect, each of the individuals is allotted a vector of ranks

$$\boldsymbol{r}(\boldsymbol{x}_i) = [r_1(\boldsymbol{x}_i) \quad r_2(\boldsymbol{x}_i) \quad \ldots \quad r_s(\boldsymbol{x}_i)]^{\mathrm{T}} \in \mathbf{R}^s \tag{4}$$

where $r_j(\boldsymbol{x}_i), j = 1, 2, \ldots, s,$ represents the rank of the *i*-th individual within the *j*-th genetic gender (X*j*).

According to the proposed genetic-gender approach GGA, the assignment of the genetic gender l_i to each individual $\boldsymbol{x}_i$ in the population is performed by computing the following procedural quantities [10, 13]:

$$\varphi_i^j = \frac{r_j(\boldsymbol{x}_i)}{\max\limits_{i=1,2,\ldots,N}\left\{r_j(\boldsymbol{x}_i)\right\}}, \; \varphi_i = \max_{j=1,2,\ldots,s}\left\{\varphi_i^j\right\}, \; l_i = \arg\max_{j=1,2,\ldots,s} \varphi_i^j \tag{5}$$

where φ_i is the obtained highest degree of sub-optimality, meaning a fuzzy measure of the memberships of the *i*-th individual to the l_i-th variant of the genetic gender.

As can be seen from (5), the index of the maximal degree of sub-optimality φ_i determines the gender of a solution under estimation in an unambiguous way.

As suggested above, we assume that only individuals of different genders create their offspring in the crossover process of the GGA algorithm (i.e. we avoid crossing individuals of the same gender). The method of selecting the parental pool is carried out according to the stochastic-remainder method [10] based on the highest degree of the membership to the gender set (X*j*) considered.

In our GGA approach we assume that the number of individuals in each of the gender groups can change in the process of evolution. We only monitor a minimum power (cardinality) of the gender sets (for instance $N/(3s)$). The lacking positions (this problem occurs only when initializing the GGA procedure) can be supplemented by individuals from the lowest Pareto front of another gender set, which are left out in the course of the GGA selection process.

It is thus clear that by introducing the idea of genetic gender we can largely alleviate the issue of dimensionality by atomizing the scope of optimality and restricting the dimension of the objective spaces considered. One can easily anticipate that this manipulation will produce a greater number of Pareto fronts leading to a measurable diversity of the generated subpopulations and to an improved effectiveness of the genetic search into the direction of both partial and total objectives.

Interesting examples supporting the GGA can be found in the nature—like, for instance, some sexual behavior among cuttlefish, which are particular about the transfer of two unique characteristics to their offspring (size and intelligence).

3.2 Concept of the Hierarchical Genetic Gender

The ideas of Hierarchical Pareto ranking (HPR) or Hierarchical Genetic Gender (HGG) of the analyzed solutions [11, 13] are based on the principles of the genetic-gender approach and observations that human decisions per-formed in a process of a multi-objective problem have the nature of hierarchical evaluation. It is worth noticing that such an approach to multi-objective optimization is based on the Pareto sub-optimal ranking. But, at the same time, during evolutionary cycles the dynamic assignment of the genders to individuals is not considered, as well as no gender restrictions are imposed on the crossover process.

In the process of creating hierarchy, a primary partition of the complete set of criteria into disjoint subsets, referred to as primary variants, is first performed. As before, the variants are marked with some labels, called primary genders. Next, a similar treatment is carried out with respect to the obtained set of primary genders (represented by the allotted labels). As a result, we obtain a master collection of secondary genders, each representing a group of primary genders. A resulting structure can be suitably described by a tree structure.

Figure 2a shows an example of division of 12 primary objective functions into four primary variants. In the second step, a complete division of the primary genders (low level) into 2 disjoint subsets of secondary ‘genders’ (high level) is

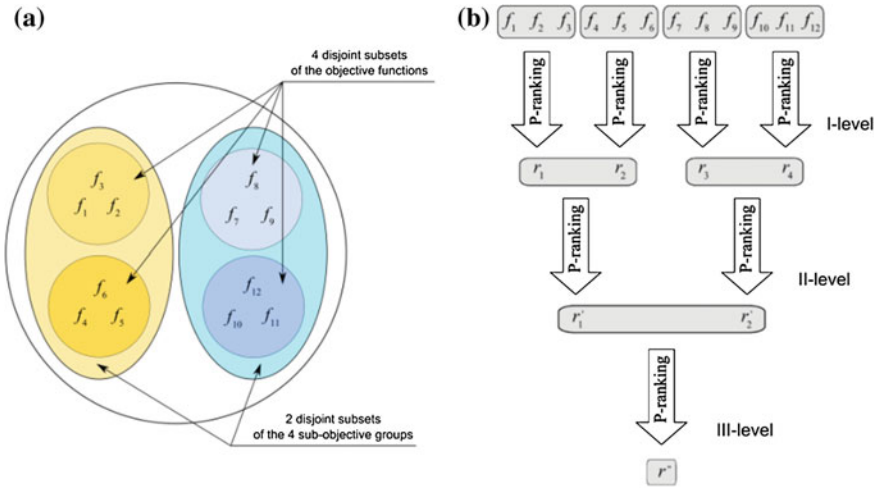


Fig. 2. Exemplary HRP 3-level: **a** distribution of objective functions and **b** ranking

performed. Suitable labels can also be applied here to name the secondary genders. The vector of s ranks (4) is applied here as the representation of primary genders, which are next subject to distribution into a set of secondary variants (some 'type', a kind of 'breed'). At both levels, the assessment of individuals is completed in the Pareto sense. At the first (I) level it is performed (as in GGA) with respect to the degree φ_i^j of sub-optimality (5) within each original variant. At the second level (II), taking into account all the members of the selected primary variants that constitute the secondary variant (and gender), the P-optimal assessment is done with respect to the same selected sub-optimality degrees φ_i^j, already computed at the first level, and next considered to be ranks [11, 13].

Figure 2b describes the optimization task consisting of 12 criteria, which are divided into four groups of Fig. 2a. Inside each of them, the Pareto ranking (I-level assessment) is carried out for all individuals in the population. As a result, a corresponding assessment vector of four (normalized) ranks is obtained. These ranks, based on the P-optimal assessment of two twofold ranks, are next used in the construction of the secondary (II) level of variants and genders.

Quite interestingly, using of HPR can always be extended to the highest level of aggregation meaning a scalar global estimation of solutions. Such an ultimate application of HPR makes thus a new proposition for defining global optimality (a challenger for GOL—the global optimality level).

The respective (achieved in HPR) scalar ranks of individuals can be used in a number of ways in the selection of parental individuals into the parental pool. This process can be easily supported, for example, by means of the method of stochastic remainder choice [13].

4 Synthesis of a Diagnostic State-Observer

To give yet another practical example of the application of the proposed GGA approach to multi-objective synthesis problem, let us now consider the problem of linear state observer synthesis [2, 7, 14, 15, 17, 20] suitable for a linear model of a ship propulsion system [13].

Fault Detection and Isolation (FDI) systems, used for diagnostic purposes, are founded on two principal operations: (a) detecting the occurrence of a fault, and (b) locating or isolating a specific fault. Such systems ensure a reliable operation of different engineering designs [17], concerning measurement, monitoring and control, for instance, where the presence of errors in the system components is disagreeable or dangerous. The detection and isolation of faults should be done as early as possible, so as to allow a human operator to take appropriate steps.

FDI system continually compare measurements of objects with predictions based on their mathematical models. The differences between the corresponding signals, called residues or residuals, allow identifying the existing faults of the system. It is clear that those differences are, in general, influenced by disturbances, noise and modeling errors. Fault detection can be, for instance, achieved by appropriate filtration of these residues, and major diagnostic decisions can also be made on the basis of their appropriate evaluation. A practical diagnosis system is in such cases founded on a residue generator [2, 7, 14, 15, 17, 20], composed of a state observer and an additional filter.

4.1 Design of Residue Generators

The state observer can be expressed in the following form [8, 13, 17]

$$\dot{\hat{\boldsymbol{x}}}(t) = (\boldsymbol{A} - \boldsymbol{KC})\hat{\boldsymbol{x}}(t) + (\boldsymbol{B} - \boldsymbol{KD})\boldsymbol{u}(t) + \boldsymbol{K}\boldsymbol{y}(t) \tag{6}$$

$$\hat{\boldsymbol{y}}(t) = \boldsymbol{C}\hat{\boldsymbol{x}}(t) + \boldsymbol{D}\boldsymbol{u}(t) \tag{7}$$

where $\hat{\boldsymbol{x}}(t) \in \mathbf{R}^n$ is a state-vector estimate, $\hat{\boldsymbol{y}}(t) \in \mathbf{R}^m$ constitutes an estimated system output, while $\boldsymbol{K} \in \mathbf{R}^{n \times m}$ stands for a matrix observer gain, $\boldsymbol{u}(t) \in \mathbf{R}^p$ is a control vector, $\boldsymbol{y}(t) \in \mathbf{R}^m$ stands for a measurement vector. The matrices appearing in an monitored model have suitable dimensions: $\boldsymbol{A} \in \mathbf{R}^{n \times n}$, $\boldsymbol{B} \in \mathbf{R}^{n \times p}$, $\boldsymbol{C} \in \mathbf{R}^{m \times n}$, $\boldsymbol{D} \in \mathbf{R}^{m \times p}$. It is presumed that the pair $(\boldsymbol{A}, \boldsymbol{C})$ is completely observable.

It is postulated that the vector fault $\boldsymbol{f}(t)$ is represented by an unknown time function and that the influence of this fault on the state evolution of the system considered and on the measurements.

The weighted residual signal $\boldsymbol{r}(t) \in \mathbf{R}^r$ can be generated as follows:

$$\boldsymbol{r}(t) = \boldsymbol{Q}(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t)) \tag{8}$$

where the matrix $\boldsymbol{Q} \in \mathbf{R}^{r \times m}$ of weights makes an additional design parameter.

The matrices $\boldsymbol{K}$ and $\boldsymbol{Q}$ allow for a simple parameterization of the designed detector. A parametrical synthesis of the considered state observer [11–13] boils down to a multi-objective optimization of the pair $(\boldsymbol{K}, \boldsymbol{Q})$

$$\underset{(\boldsymbol{K},\boldsymbol{Q})}{opt}\ \boldsymbol{J}(\boldsymbol{K},\boldsymbol{Q}) = \begin{bmatrix} \max\limits_{(\boldsymbol{K},\boldsymbol{Q})} \{\sup\limits_{s} \bar{\sigma}[\boldsymbol{W}_1(s)\boldsymbol{Q}\{\boldsymbol{C}[s\boldsymbol{I}_n - \boldsymbol{A}_0]^{-1}(\boldsymbol{F}_1 - \boldsymbol{K}\boldsymbol{F}_2) + \boldsymbol{F}_2\}]\} \\ \min\limits_{(\boldsymbol{K},\boldsymbol{Q})} \{\sup\limits_{s} \bar{\sigma}[\boldsymbol{W}_2(s)\boldsymbol{Q}\boldsymbol{C}[s\boldsymbol{I}_n - \boldsymbol{A}_0]^{-1}\boldsymbol{N}]\} \\ \min\limits_{(\boldsymbol{K},\boldsymbol{Q})} \{\sup\limits_{s} \bar{\sigma}[\boldsymbol{W}_3(s)\boldsymbol{Q}\boldsymbol{C}[s\boldsymbol{I}_n - \boldsymbol{A}_0]^{-1}]\} \\ \min\limits_{(\boldsymbol{K},\boldsymbol{Q})} \{\sup\limits_{s} \bar{\sigma}[\boldsymbol{W}_4(s)\boldsymbol{Q}\{\boldsymbol{I}_m - \boldsymbol{C}[s\boldsymbol{I}_n - \boldsymbol{A}_0]^{-1}\boldsymbol{K}\}]\} \\ \min\limits_{(\boldsymbol{K})} \{\bar{\sigma}[\boldsymbol{A}_0^{-1}]\} \\ \min\limits_{(\boldsymbol{K})} \{\bar{\sigma}[\boldsymbol{A}_0^{-1}\boldsymbol{K}]\} \end{bmatrix} \tag{9}$$

where $\bar{\sigma}[\cdot]$ denotes a maximal singular value of matrix, while $\boldsymbol{W}_1(s)$, $\boldsymbol{W}_2(s)$, $\boldsymbol{W}_3(s)$ and $\boldsymbol{W}_4(s)$ are weighting matrix functions [11–13], which allow for separating the effects of faults from disturbances and noises. The matrices $\boldsymbol{F}_1$, $\boldsymbol{F}_2$ and $\boldsymbol{N}$ represent an impact of faults and noises on the system [11–13].

The first coordinate of vector (9) is the main maximised criterion, which takes into consideration the influence of faults $\boldsymbol{f}(t)$ on the residuum $\boldsymbol{r}(t)$. The next three co-ordinates describe of the impact of disturbances, input noises and measurement noises on the state evolution. By describing the influence of static deviations from the nominal model of the plant, the last two coordinates give a robustness measure.

In our approach the analyzed multi-objective optimization problem is solved by three specific method: GGA, MSGA and NSGA2. In particular, the design of residue generators is based on the optimization of the objective function $\boldsymbol{J}(\boldsymbol{K}, \boldsymbol{Q})$ of (9).

To guarantee that genetic optimization yields exclusively permissible solutions, $\text{spectr}[\boldsymbol{A}_0] \subset \mathbf{C}_-$, we directly search only for eigenvalues (and not for the observer gain $\boldsymbol{K}$), on the basis of which the matrix $\boldsymbol{K}$ is calculated by means of the pole placement method [10, 14].

By setting the matrix $\boldsymbol{Q}$ to identity and accordingly fixing the frequency weighting matrices $\boldsymbol{W}_t(s)$, the optimization problem (9) can be reduced to the following task:

$$\underset{(\boldsymbol{K},\boldsymbol{Q})}{opt}\ \boldsymbol{J}(\boldsymbol{K},\boldsymbol{Q}) = \underset{\boldsymbol{K}}{opt}\,\boldsymbol{J}(\boldsymbol{K}) = \underset{\lambda}{opt}\,\boldsymbol{J}(\boldsymbol{K}(\lambda)) = \underset{\lambda}{opt}\,\boldsymbol{J}(\lambda) \tag{10}$$

where $\lambda \subset \mathbf{C}^n$ is the sought n-element vector of the eigenvalues of the matrix $\boldsymbol{A}_0$.

4.2 Results of Evolutionary Optimization

The design criteria applied in the optimization task (9) have been divided into three gender sets. The first gender is composed of the performance criterion (the impact of the faults on the residue). The second gender embraces the three insensitivity criteria (the influence of disturbances and noise). The third gender consists of the two robustness measures (the effects of the plant deviation from the nominal model). Thus in each cycle of our evolutionary multi-optimization process all individuals are iteratively assigned these three definite gender variants (performance, insensitivity, and robustness), and, next, the corresponding GG sets are suitably applied in the inter-gender crossover mating process.

The results of the evolutionary optimization are shown in Fig. 3, where the NSGA2 and MSGA are compared with GG and GAHP. As can be seen, the GGA and GAHP algorithms achieve greater values of the global optimality level. The figure shows the average (made of 20 trials/simulations) of a max GOL (maximum of all individuals) obtained in subsequent generations (epochs). Detailed values the optimization parameters can be found in the works [10, 13]. Verifying simulations of the performance of the state observers designed with the use of the GGA P-optimization evolutionary computations have been performed in the presence of faults, noise and other external disturbances [6]. The additive faults considered in the study are shown in Fig. 4a.

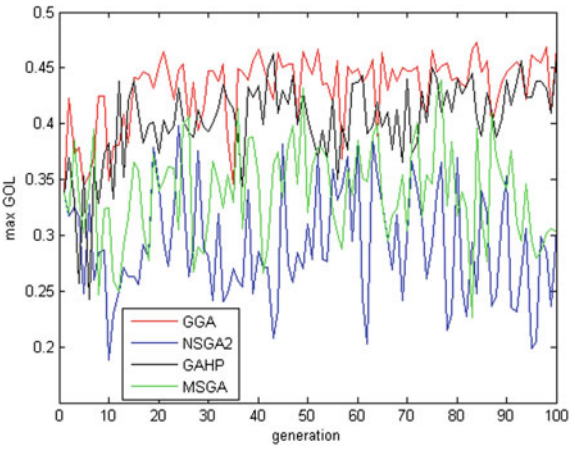


Fig. 3. Maximum global optimality level GOL during a mean simulation run

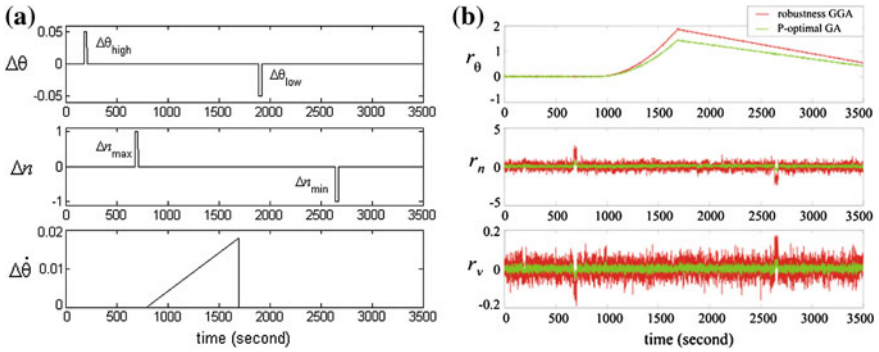


Fig. 4. Signals of the designed observer-based FDI system: **a** the additive faults; **b** the three residuals obtained for the P-optimal classical-GA solution and for the robustness-GG observer

A comparison of the total P-optimal solutions obtained by using the classical multi-objective GA algorithm with the robustness GGA solution is presented in Fig. 4b.

As can be easily seen, practically all the faults of Fig. 4b give distinctive symptoms in at least one of the residues. What is more, the residuals demonstrate changes analogous to the generic fault signal applied. It is obvious that with the use of appropriate filtration the symptom information included in the residues makes it possible to detect and isolate all the faults (though, with the assumed system model the temporary pitch-sensor fault effects are less clear as compared to the others).

5 Conclusions

An instructive feature of the proposed multi-optimization approach is the way of utilizing the Pareto-optimization results. Namely, within each gender set the Pareto-optimization is used as a helpful tool of sub-optimal judgment of the 'internal' single-gender rivals for the purpose of their uniform estimation and selection (in a greater number) to the new parental subset (and to the next generation) in each iteration cycle of evolutionary computations (or genetic algorithms). It is worth highlighting that despite of relying on this limited perspective the very notion of the set-fitting P-sub-optimality is entirely clear and practically adequate.

The method can be interpreted in terms of: (i) a new mechanism of pre-selecting both the transient and the final individuals (solutions), and (ii) a mutual inter-gender support in genetic search.

The standard concept of Pareto-optimality can still be applied to the final set of solutions on a regular basis. Another way of processing, and taking into account the 'full scope' of optimality, can be based either on the notion of the global optimality

level GOL, calculated based on the fitness, or rank functions, or on the notion of HPR/HGG, based on a developed hierarchy of genetic genders.

A major success of the gender approach can be attributed to the fact that it appropriately deals with a great number of objectives by reducing the dimensionality of the Pareto-analyzed spaces. Observe that in the full scope optimization case, due to a high dimension of the objectives space, the number of Pareto fronts is strongly limited. This means that many solutions are estimated as equivalent from the Pareto optimality viewpoint (i.e. they are of the same rank). As a result, the process of selecting individuals is not effective and the evolutionary search is overly stochastic with no indications and progress in particular directions represented by the stated criteria.

On the contrary, by introducing the gender approach we solve the above issue by means of restricting the dimensions of the objective subspaces and bringing about a greater number of Pareto fronts within each gender population analyzed in a subspace of a restricted dimension (i.e. solely in the space of the assigned gender objectives). This, in turn, brings about diversity among the individuals of the GGA subpopulations that can be easily estimated and used in effectively pushing the evolutionary exploration into the desired directions on the basis of the achievable distinctive ordering.

Our approach is also entirely different from other propositions. Although showing several instrumental consequences, the GGA method has a conceptual nature consisting in the objective space decomposition of the initial problem, as well as in the effective reduction of an originally highly-dimensional problem.

References

1. Bader, J., Zitzler, E.: A hypervolume-based optimizer for high-dimensional objective spaces. In: Conference on Multiple Objective and Goal Programming (MOPGP 2008), Lecture Notes in Economics and Mathematical Systems. Springer, New York (2009)
2. Chen, J., Patton, R.J., Liu, G.: Optimal residual design for fault diagnosis using multi-objective optimization and genetic algorithms. Int. J. Syst. Sci. **27**(6), 567–576 (1996)
3. Coello, C.C.A., Lamont, G.B., Van Veldhuizen D.A.: Evolutionary Algorithms for Solving Multi-Objective Problems. Genetic and Evolutionary Computation, 2nd edn. Springer, Berlin (2007)
4. Deb, K.: Current trends in evolutionary multi-objective optimization. Int. J. Simul. Multidiscip. Optim. **1**(1), 1–8 (2007)
5. Fonseca, C.M., Fleming, P.J.: An overview of evolutionary algorithms in multiobjective optimization. IEEE Trans. Evolut. Comput. **3**(1), 1–16 (1995)
6. Izadi-Zamanabadi, R., Blanke M.: A Ship Propulsion System Model for Fault-tolerant Control. Technical Report, no. 4262. Aalborg University, Denmark (1998)
7. Korbicz, J., Kościelny, J.M., Kowalczuk, Z., Cholewa, W. (Eds.) Fault Diagnosis. Models, Artificial Intelligence, Applications. Springer, Berlin, Heidelberg, New York (2004)
8. Kowalczuk, Z., Białaszewski, T.: Genetic algorithms in multi-objective optimization of detection observers. In: Fault Diagnosis. Models, Artificial Intelligence, Applications, pp. 511–556. Springer, Heidelberg (2004)

9. Kowalczuk, Z., Białaszewski, T.: Improving evolutionary multi-objective optimisation by nichning. Int. J. Inf. Technol. Intell. Comput. **1**(2), 245–257 (2006)
10. Kowalczuk, Z., Białaszewski, T.: Improving evolutionary multi-objective optimisation using genders. In: Artificial Intelligence and Soft Computing. Lecture Notes in Artificial Intelligence, vol. 4029, pp. 390–399. Springer, Berlin (2006)
11. Kowalczuk, Z., Białaszewski, T.: Designing FDI observers by improved evolutionary multi-objective optimization. In: Proceedings 6th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, pp. 601–606. Beijing, China (2006)
12. Kowalczuk, Z., Białaszewski, T.: Gender selection of a criteria structure in multi-objective optimization of decision systems (in Polish: Rodzajnikowy dobór struktury kryteriów w zadaniach wielokryterialanej optymalizacji systemów decyzyjnych). Pomiary Automatyka Kontrola **57**(7), 810–814 (2011)
13. Kowalczuk, Z., Białaszewski, T.: Genetic-gender approach to multi-objective optimization of detection observers with pre-selection of criteria. In: Intelligent Systems in Technical and Medical Diagnostics. Advances in Intelligent Systems and Computing, vol. 230, pp 161–174. Springer, Heidelberg (2014)
14. Kowalczuk, Z., Suchomski, P., Białaszewski, T.: Evolutionary multi-objective pareto optimization of diagnostic state observers. Int. J. Appl. Math. Comput. Sci. **9**(3), 689–709 (1999)
15. Kowalczuk, Z., Suchomski, P., Białaszewski, T.: Genetic multi-objective pareto optimization of state observers for FDI. In: Proceedings European Control Conference, (CD-ROM). Karlsrhure, Germany (1999)
16. Lis, J., Eiben, A.: A multi-sexual genetic algorithm for multi-objective optimization. In: Proceedings of the IEEE International Conference on Evolutionary Computaiton, pp. 59–64 (1997)
17. Patton, R.J., Frank, P.M., Clark, R.N. (eds.): Fault Diagnosis in Dynamic Systems. Theory and Application. Prentice Hall, New York (1989)
18. Rejeb, J., AbuElhaija, M.: New gender genetic algorithm for solving graph partitioning problems. In: Proceedings of the 43rd IEEE Midwest Symposium on Circuits and Systems, vol. 1, pp. 444–446 (2000)
19. Sanchez-Velazco, J., Bullinaria, J.A.: Sexual selection with competitive/co-operative operators for genetic algorithms. In: Proceedings the IASTED International Conference on Neural Networks and Computational Intelligence, pp. 191–196. ACTA Press (2003)
20. Suchomski, P., Kowalczuk, Z.: Robust H∞-optimal Synthesis of FDI Systems. In: Fault Diagnosis. Models, Artificial Intelligence, Applications, pp. 261–298. Springer, Heidelberg (2004)
21. Sodsee, S., Meesad, P., Li, Z., Halang, W.: A networking requirement application by multi-objective genetic algorithms with sexual selection. In: 3rd International Conference Intelligent System and Knowledge Engineering, vol. 1, pp. 513–518 (2008)
22. Song, Goh K., Lim, A., Rodrigues, B.: Sexual selection for genetic algorithms. Artif. Intell. Rev. 123–152 (2003)
23. Vrajitoru, D.: Simulating gender separation with genetic algorithms. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 634–641 (2003)
24. Yan, T.: An improved genetic algorithm and its blending application with neural network. In: 2nd International Workshop Intelligent Systems and Applications, pp. 1–4 (2010)
25. Zitzler, E., Thiele, L., Bader, J.: On set-based multiobjective optimization. IEEE Trans. Evolut. Comput. **14**(1), 58–79 (2010)

Experimental Comparison of Straight Lines and Polynomial Interpolation Modeling Methods in Ship Evolutionary Trajectory Planning Problem

Piotr Kolendo and Roman Śmierzchalski

Abstract Paper presents the application of evolutionary algorithms and polynomial interpolation in ship evolutionary trajectory planning method and its comparison to classic approach, where trajectory is modeled by straight lines. Evolutionary algorithms are group of methods that allows\ to find a collision free trajectory in real time, while polynomial interpolation allows to model smooth trajectory, which keeps continuity of velocity and acceleration values along path in opposition to straight lines approach. Paper presents the experimental researches for several collision situations at sea with application of trajectory modeled by straight lines and polynomial interpolation.

Keywords Trajectory planning · Evolutionary algorithms · Avoiding collision

1 Introduction

The problem of evolutionary path planning is a common theme in numerous applications such as: ship path planning [3, 6, 12–14], path planning for AUV's [2], mobile robots path planning [7, 16] and path planning for aerial objects [8–11]. Problem is defined as a task where given a mobile object with certain dynamical and kinematical properties and an environment through which this object is travelling, one needs to plot a path between start and end points, which avoids all environment`s static and dynamic obstacles and meets the optimization criteria. The evolutionary path planning method is non-deterministic, based on a natural selection mechanism. Its most important advantages are: build-in adaptation mechanism

P. Kolendo (✉) · R. Śmierzchalski
Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland
e-mail: pkolendo@ely.pg.gda.pl

R. Śmierzchalski
e-mail: roman.smierzchalski@pg.gda.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_24

for a dynamic environment and reaching a multi-criteria task solution in a near-real time.

In classic approach, where trajectory is modeled with straight lines, algorithm is responsible for finding turning points, where afterwards these points are connected by straight lines. This approach was presented in [6, 12]. Trajectory modeled with polynomial interpolation was presented in [3]. The main advantage of polynomial interpolation is ability of modeling smooth trajectory, which keeps continuous values of speed and acceleration alongside. According to [4] it can be claimed that the trajectory modeled this way is closest to navigators expectations. Under some assumptions trajectory modeled with polynomial interpolation can mimic real movement of ship. In [1] the complex method combined of A* optimization algorithm (which is responsible for determination optimal trajectory) and cubic splines (for its modeling) was presented. Method allows to find a smooth, collision free trajectory. This however, not able to find solution the in near real time, what significantly limit its application in on-line system of ship trajectory planner (according to summary of paper [1]). It is due to the fact, that it has to take into consideration dynamic changes in environment.

Paper presents application of a new method which combines advantages of polynomial interpolation and evolutionary algorithms. Several collision situation at sea was presented with solutions for both classical and proposed approach.

Papers is organized as follows. After the introduction the evolutionary ship trajectory planning method is presented. Third section describes cubic splines modeling method to path planning problem. Section 4 presents experimental researches for several collision situations at sea. Section 5 concludes the paper.

2 Evolutionary Ship Trajectory Planning

Evolutionary algorithms are specific methods used for ship trajectory planning. The main advantages of evolutionary algorithms, generally in path planning problem, are easy implementation in problems with large number of constraints (independently of problem characteristics) [9] and ability to find final solution in near-real time [12]. In opposition to analytic methods they allow for active search of solution space [14]. Because of that, evolutionary algorithms are widely used in problems of ship trajectory planning [5, 12, 14, 15].

According to transport plan, an own ship should cover the given path in the determined time. On the other hand, it has to move safely along the planned path, avoiding at the same time the navigational constrains and other moving objects. Path planning in a collision scenario has to stand a compromise between a deviation from a given course and ships safety. Thus the problem is defined as multi-criteria optimization task which considers the safety and economics of ships movement. Every path is evaluated based on the fitness function. In the considered case, the problem has been reduced to a single objective optimization task with weighting factors

$$Total\_Cost(S) = Safe\_Cond(S) + Econ\_Cond(S) \quad (1)$$

$$Safe\_Cond(S) = w_c * clear(S) \quad (2)$$

$$Econ\_Cond(S) = w_d * dist(S) + w_s * smooth(S) + w_t * time(S) \quad (3)$$

Figure 1 shows a single evolutionary ship path planning algorithm diagram. Individual is one single solution (path). Population is a group of paths which are in

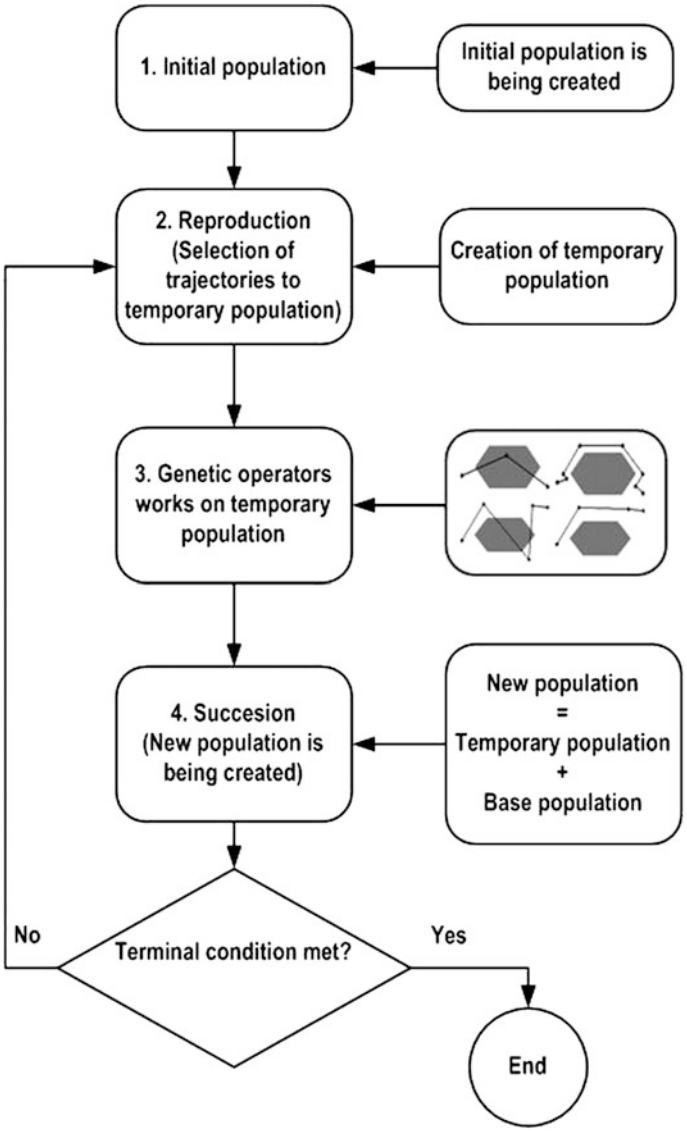


Fig. 1 Evolutionary algorithm diagram

evolution process. Generation is one algorithm iteration. Fitness function determines the fitness of individual to environment. The way of encoding of chromosomes was presented in [12]. In the first step a random population is being initiated. In the second step, using a chosen selection scheme, a specific number of individuals is randomly selected to the temporary population. Then the genetic operators such as cross-over and mutation are working on the temporary population. In the next step a new population is established. It consists of base and temporary population best individuals. The algorithm's iterations is are repeated until the termination condition is met (a certain number of iterations in this instance).

3 Trajectory Modeling

In the classic approach to evolutionary planning, trajectory is being modeled by straight lines, which connect turning points. Application of polynomial interpolation for modeling ship trajectory allows to determine smooth trajectory with continuous values of speed and acceleration alongside. This kind of trajectory is desired for control system of the ship, due to the fact that it allows to control the ship along the trajectory without overshoots and with minimal difference from reference trajectory.

In the considered case the most significant issue is to find smooth trajectory with continues values of speed and acceleration alongside. According to [3, 4] 3rd degree polynomial-cubic spline is sufficient to achieve this goal. Higher degree polynomials allow for assumption of more constraint conditions, however the smooth trajectory may be deformed because of Runge's phenomena. In [1] there is comparison of cubic splines, 5th and 7th degree polynomials, where this problem is pointed out.

In the presented method each segment of trajectory is modeled according to Eq. (4). Coefficients are calculated simultaneously for all segments of trajectory.

$$\begin{aligned} x_k(s) &= a_3 s^3 + a_2 s^2 + a_1 s + a_0 \\ y_k(s) &= b_3 s^3 + b_2 s^2 + b_1 s + b_0 \end{aligned} \tag{4}$$

where: k—is the number of segment.

Waypoints are described as follows:

$$\begin{aligned} &s \in R = \{0, \ldots, n\} \\ &s_0 = s(x_0, y_0) = 0 \\ &s_1 = s(x_1, y_1) = 1 \\ &s_2 = s(x_2, y_2) = 2 \\ &\quad \ldots \\ &s_n = s(x_n, y_n) = n \end{aligned} \tag{5}$$

where n—is the number of waypoints.

Sample trajectory modeled with cubic splines is presented below (Fig. 2).

To clarify the description, calculation of coefficients will be presented only for x- coordinate. For y-coordinate calculations are analogical.

Due to the fact that each segment preserves continuity, the constraints for coefficient determination should concern continuity in waypoints

$$\begin{aligned} x_k(s) &= a_3 s^3 + a_2 s^2 + a_1 s + a_0 \\ x_k^{'}(s) &= 3a_3 s^2 + 2a_2 s + a_1 \\ x_k^{''}(s) &= 6a_3 s + 2a_2 \end{aligned} \tag{6}$$

In order to determine polynomial coefficients there should be assumption of 2 constraints in start and end points 4 constraints in the rest of waypoints [3]. Limitations concern:

- the modeled segment k of trajectory should start in waypoint s_{n-1} and end in s_n (7),

$$\begin{aligned} x_k(s_{n-1}) &= x_{n-1} \\ x_k(s_n) &= x_n \end{aligned} \tag{7}$$

- the value of derivative should be the same on both sides of waypoint (8),

$$\lim_{s \to s_n^-} x_k'(s_n) = \lim_{s \to s_n^+} x_k'(s_n) \tag{8}$$

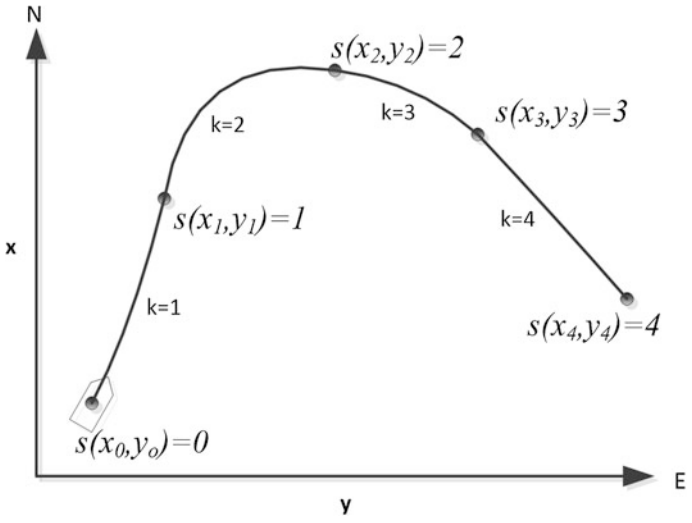


Fig. 2 Trajectory modeled with cubic splines

- the value of the second derivative should be the same on both sides of waypoint (9),

$$\lim_{s \to s_n^-} x_k''(s_n) = \lim_{s \to s_n^+} x_k''(s_n) \tag{9}$$

In order to determine a cubic spline function, it is necessary to solve the following matrix Eq. (10).

$$\begin{aligned} \boldsymbol{C} &= \boldsymbol{A}\boldsymbol{W} \\ \boldsymbol{W} &= \boldsymbol{A}^{-1}\boldsymbol{C} \end{aligned} \tag{10}$$

where $\boldsymbol{W}$ is coefficient matrix for segments 1 to k:

$$\boldsymbol{W} = \begin{bmatrix} a_{3,1} & \dots & a_{3,k} \\ a_{2,1} & \dots & a_{2,k} \\ a_{1,1} & \dots & a_{1,k} \\ a_{0,1} & \dots & a_{0,k} \end{bmatrix}^T \tag{11}$$

$\boldsymbol{C}$ is constraints matrix and $\boldsymbol{A}$ is transformation matrix. Detailed way of finding solution of this equation is presented in [3].

4 Simulations

For simulation tests several collision situations at sea, with different levels of complexity, were performed. All environments consist of a large number of dynamic constraints. The algorithm settings were as follows:

- algorithm with partially exchangeable population
- population consists of 30 individuals with 40 % rate of exchange in each population
- probability of crossover was set at 0.7
- probability of mutation was 0.5. 5 mutation operators described in [12], as standard mutation, soft mutation, adding/deletion of a gene, swap gene position, speed mutation
- terminal condition was set for 400 generations.

For trajectory modeled with polynomial interpolation it is important to set high value of mutation probability. It is significant to help the algorithm with leaving local extremes, especially in inital phase of the algorithm, were there are only unfeasible trajectories. Each collision situation was tested for several sets of algorithm settings and initial population. Figures 3, 4, 5 present collision situations

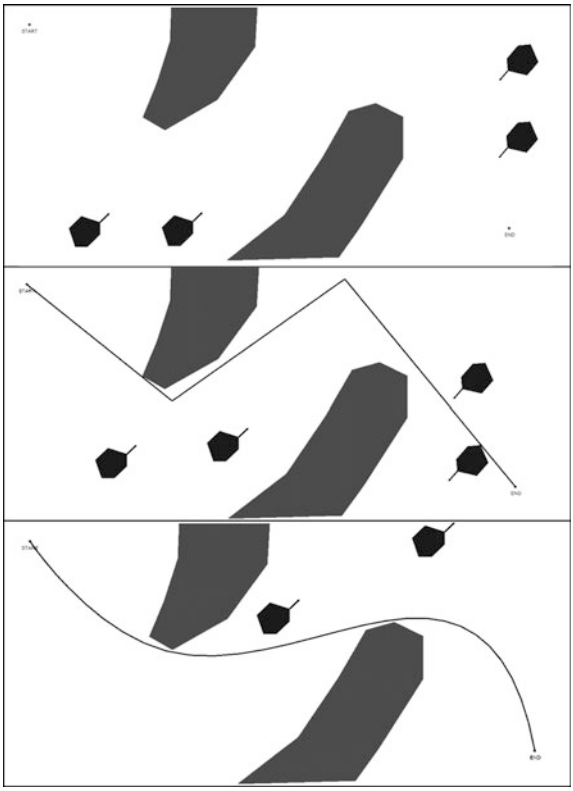


Fig. 3 1st sample collision situation at sea and final solution for trajectory modeled with straight lines and polynomial interpolation

at sea and final solutions of chosen runs, based on of straight lines and cubic splines.

The solutions designated with straight lines and polynomial interpolation in most cases are very similar as in Figs. 3, 4, 5. The difference will occur during steering the vessel alongside the trajectory. Due to the fact that trajectory modeled by polynomial is smooth and preserves described continuity, movement alongside will be performed theoretically without any overshoots in comparison to classic approach, where trajectories are made from straight lines and ships dynamic is being approximated by circle arcs (on straight line angular speed is equal to zero and has a define value at circular arc). Overshoots (besides energy loss caused by movement corrections) result forom with unprecise steering along trajectory, what in highly congested areas may result in collision (because of significant difference between reference and real trajectory).

As it can be seen in all figures, with polynomial modeled trajectory, which were found by the algorithm precise control of the ship, is possible without overshoots, from start to end point. It is very important, due to the fact that any deviation from reference trajectory may result in collision.

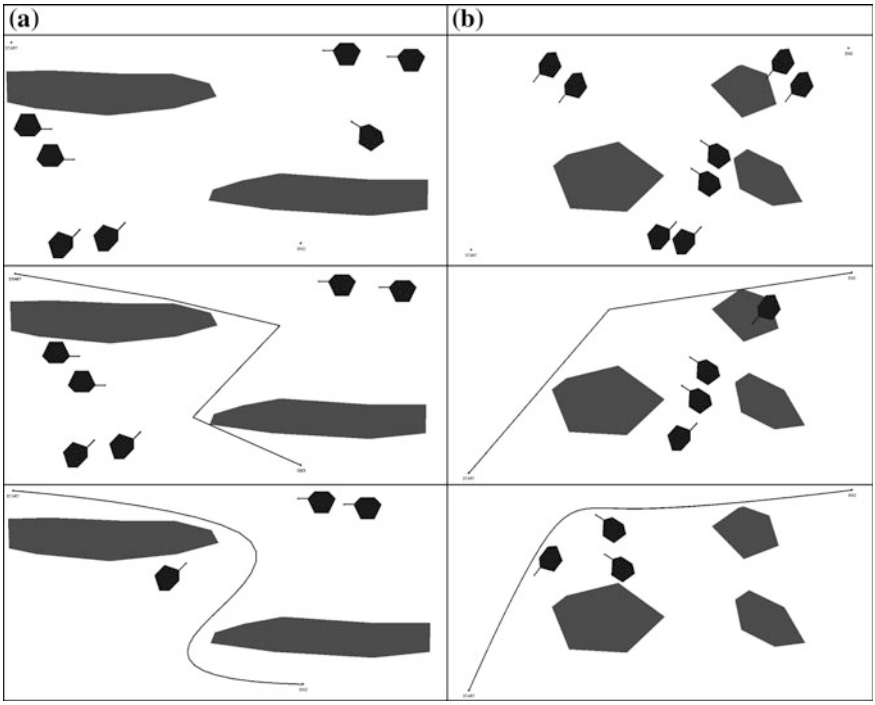


Fig. 4 The 2nd sample collision situation at sea and final solution for trajectory modeled with straight lines and polynomial interpolation (**a**), and the 3rd sample collision situation at sea and final solution for trajectory modeled with straight lines and polynomialinterpolation (**b**)

5 Results and Conclusions

Combination of evolutionary algorithms and cubic splines allows us to find smooth trajectories in near-real time, additionally preserving continuity of speed and acceleration of the ship alongside. Theoretically, this trajektory makes it possible to precisely control of the ship in congested areas and minimize the energy losses caused by overshoots. It is very important during steering the vessel in areas highly congested with static and dynamic obstacles. As it can be seen in presented situations, any deviation from reference trajectory may result in collision.

New method allows the user to find solution in near-real time, which is significant for algorithms working in on-line mode. It makes possible to verify founded trajectory with changes in environment, during movement of a vessel along trajectory.

Moreover, trajectories set in this way, under some assumptions, can mimic real movement of the ship (taking into account dynamic properties of the ship). Due to complexity of this issue, dynamics pertinent to this method will be investigated in further works.

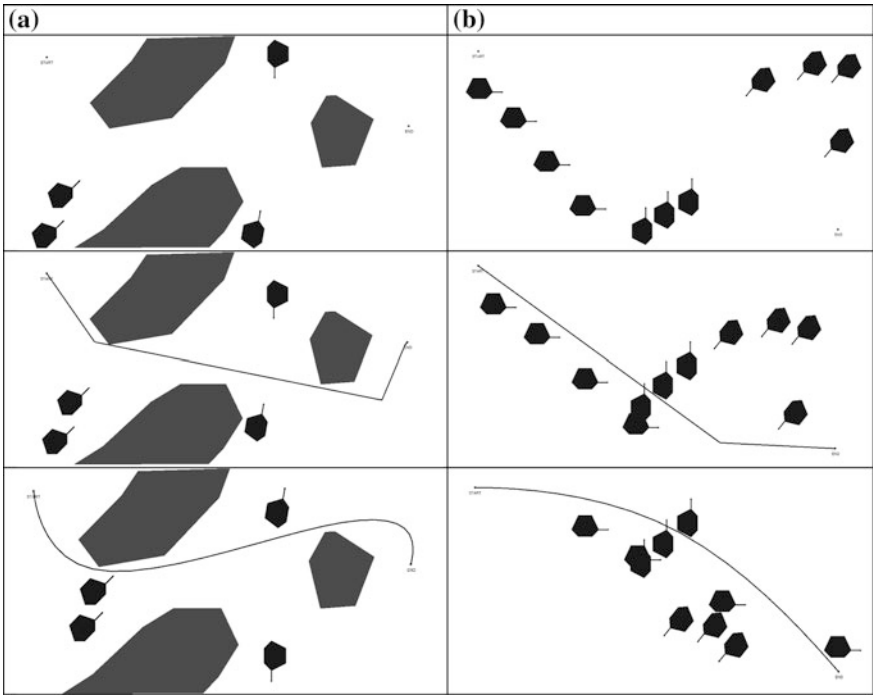


Fig. 5 The 4th sample collision situation at sea and final solution for trajectory modeled with straight lines and polynomial interpolation (**a**), and the 5th sample collision situation at sea and a final solution for trajectory modeled with straight lines and polynomial interpolation (**b**)

References

1. Corneliussen, J.: Implementaion of a Guidance System for Cybership II, (Master thesis). Norwegian University of Science and Technology, Trondheim, Norway (2003)
2. Fogel, D.B., Fogel L.J.: Optimal routing of multiple autonomous underwater vehicles through evolutionary programming. In: Proceedings of the 1990 Symposium on Autonomous Underwater Vehicle Technology, pp. 44–47 (1999)
3. Fossen, T.I.: Marine Control Systems: Guidance, Navigation and Control of Ships. Rigs and Underwater Vehicles. Norwegian University of Science and Technology, Marine Cybernetics AS Trondheim, Norway (2010)
4. Golding, B.K.: Industrial Systems for Guidance and Control of Marine Surface Vessels. Norwegian University of Science and Technology Trondheim, Norway (2004)
5. Ito, M., Zhang, F., Yosida, N.: Collision avoidance control of ship with genetic algorithm. In: Proceedings of the IEEE International Conference on Control Application (1999)
6. Kolendo, P., Śmierzchalski, R., Jaworski, B.: Comparison of selection schemes in evolutionary method of path planning. In: Lectures Notes in Artificial Intelligence: Computational Collective Intelligence: Technologies and Applications, LNAI 6923. Springer, Berlin (2011)
7. Kurata, J., Grattan, K.T.V., Uchiyama, H.: Path planning for a mobile robot by integrating mapped information. In: 3rd IFAC Conference IAV Intelligent Autonomous Vehicle, Madrid, pp. 319–323 (1998)

8. Mittal, S., Deb, K.: Three dimensional offline path planning for UAVs using multiobjective evolutionary algorithms. In: IEEE Congress on Evolutionary Computation CEC (2007)
9. Nikolos, I.K., Valavanis, K.P., Tsorveloudis, N.C., Kostaras, A.C.: Evolutionary algorithm based offline/online path planner for UAV navigation. IEEE Trans. Syst. Man Cybern. **33**(6), 898–912 (2003)
10. Pongpunwattana, A., Rysdyk, R.: Evolution-based dynamic path planning for autonomous vehicles. In: Innovations in Intelligent Machines, pp. 113–145 Springer, Berlin (2007)
11. Rathbun, D., Capozzi, B., Kragelund, S., Pongpunwattana, A.: An evolution based path planning algorithm for autonomus motion of a UAV through uncertain environments. In: Proceedings of the AIAA Digital Avionics System Conference (2002)
12. Śmierzchalski, R.: Synteza Metod i Algorytmów Wspomagania Decyzji Nawigatora w Sytuacji Kolizyjnej na Morzu. Akademia Morska, Gdynia (1998)
13. Śmierzchalski, R., Michalewicz, Z.: Path planning in dynamic environments. In: Patnaik, S., Jain, L.C., Tzafestas, S.G., Resconi, G., Konar, A., (eds.) Innovations in Robot Mobility and Control, pp. 135–153. Springer-Verlag, Berlin (2005)
14. Szłapczyńska, J.: Zastosowanie Algorytmów Ewolucyjnych oraz Metod Rankingowych do Planowania Trasy Statku z Napędem Hybrydowym, (Ph.D. Thesis). Uniwersytet Zachodniopomorski, Szczecin (2009)
15. Szłapczyński, R.: Evolutionary sets of safe ship trajectories within traffic separation schemes. J. Navig. **66**, 65–81 (2012)
16. Xiao, J., Michalewicz, Z.: An evolutionary computation approach to planning and navigation. In: Hirota, K., Fukuda, T. (eds.) Soft-Computing and Mechatronics, pp. 118–136. Physica, Heidelberg (1999)

Robust Fault Detection by Means of Echo State Neural Network

Andrzej Czajkowski and Krzysztof Patan

Abstract This paper deals with the application of Echo State Network (ESN) model to robust fault diagnosis of the Twin Rotor Aero-Dynamical System (TRAS) through modeling the uncertainty of the neural model with the so-called Model Error Modeling method (MEM). The work describes the modeling process of the plant and scenarios in which the system is under influence of the unknown fault. In such fault scenarios the ESN model together with MEM are used to form the uncertainty bands. If the system output exceeds the uncertain region the fault occurrence is signalized. All data used in experiments are collected from the TRAS through the Matlab/Simulink environment.

Keywords Echo state network · Genetic algorithm · Robust fault detection · Model error modeling

1 Introduction

Recently, it has been observed an increasing development of the Fault Diagnosis (FD) methods for the Fault Tolerant Control (FTC) system design purposes. It is directly connected to the advantages of the systems which can maintain current performance of control as close as possible to the desirable one, and preserve stability conditions in presence of faults. Faults and equipment failures directly affect the performance of the control system and can result in large economic losses and violation

This work was supported by the National Science Center in Poland under the grant UMO-2012/07/N/ST7/03316.

A. Czajkowski (✉) · K. Patan
Institute of Control and Computation Engineering, University of Zielona Góra, ul. Podgórna 50, 65-246 Zielona Góra, Poland
e-mail: a.czajkowski@issi.uz.zgora.pl

K. Patan
e-mail: k.patan@issi.uz.zgora.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_25

of the safety regulations. During the fault tolerant control system design, the basic problem is the early detection and identification of possible faults. The fault diagnosis is crucial in case when certain fault accommodation mechanisms need to be switched on when fault occurs. When the system is working in nominal operating conditions applying fault compensation terms can bring unnecessary computational burden or can even lead to unpredictable behavior. Nowadays, the most popular approach is model-based. One of the tools which are extensively exploited in this task are Artificial Neural Networks (ANN) [1, 11, 12]. Such model can reflect dynamics of the plant very closely but not perfectly. This is due to noise or modeling uncertainty. This can lead to false alarms and unnecessary control behavior. Usually, such problems are solved with the application of thresholds which allows for some inaccuracies between model and plant outputs. The cost of such solutions is lower sensitivity of detection. To improve detection accuracy without much sensitivity sacrificed it is important to consider modeling uncertainty during design of fault diagnosis block. In this work it is proposed to model that uncertainty with error model which uses autoregressive structure.

The paper is organized as follows. Section 2 presents a general description of the Echo State Networks. The robust fault detection method, and the so-called model error modeling, are described in Sect. 3. Section 4 presents a two rotor aerodynamical system, while experimental results are included in Sect. 5.

2 Echo State Network

Echo state networks are relatively new idea to architecture and supervised learning principle of the recurrent neural networks (RNNs). The idea of creating a random and large but fixed recurrent neural network and combining the nonlinear responses of the reservoir (sparsely connected neurons inside the hidden layer of the RNN) through trainable, linear combination for the desired output was presented in [8].

The discrete-time Echo State Network with N reservoir units, K inputs and L outputs is governed by the following state update equation:

$$\begin{aligned} \bar{\mathbf{x}}(k+1) &= h(\mathbf{W}\bar{\mathbf{x}}(k) + \mathbf{W}^{in}\mathbf{u}(k) + \mathbf{W}^{fb}\bar{\mathbf{y}}(k)) \\ \bar{\mathbf{y}}(k) &= g(\mathbf{W}^{out}\bar{\mathbf{z}}(k)) \end{aligned} \tag{1}$$

where $\bar{\mathbf{x}}(k) \in \mathbb{R}^N$ is the reservoir state, h is a squashed activation function (usually the logistic or the hyperbolic tangent function), $\mathbf{W} \in \mathbb{R}^{N\times N}$ is the reservoir weight matrix, $\mathbf{W}^{in} \in \mathbb{R}^{N\times K}$ is the input weight matrix, $\mathbf{u} \in \mathbb{R}^K$ is the process input, $\mathbf{W}^{fb} \in \mathbb{R}^{N\times L}$ is the output feedback matrix, $\bar{\mathbf{y}}(k) \in \mathbb{R}^L$ is the model output, g is an output activation function, $\bar{\mathbf{z}}(k) = [\bar{\mathbf{x}}(k); \mathbf{u}_c(k)]$ is the extended system state vector and it is a concatenation of the reservoir and input states.

What is very important, especially in the control theory, it is the possibility to feedback the real output of the system during the work of the plant to obtain system observer. In situation when the system states cannot be measured, those states can

be approximated by ESN in a very easy manner. Equations describing nonlinear observer based on ESN are formulated as follows:

$$\begin{aligned} \hat{\mathbf{x}}(k+1) &= h(\mathbf{W}\hat{\mathbf{x}}(k) + \mathbf{W}^{in}\mathbf{u}(k) + \mathbf{W}^{fb}\mathbf{y}(k)) \\ \hat{\mathbf{y}}(k) &= g(\mathbf{W}^{out}\hat{\mathbf{z}}(k)) \end{aligned} \tag{2}$$

where $\mathbf{y}(k)$ is the measured system output.

To design a proper model using Echo State Networks, the most important task is to tune the global parameters so as to match the dynamics of the modeled system. Based on tuning of such parameters the reservoir characteristic is configured in the following terms:

- the spectral radius of the reservoir weight matrix
- the input scaling
- the output feedback scaling
- the connectivity of the reservoir weight matrix
- the reservoir size N.

All these parameters have to be optimized jointly. Currently used practice play on tuning these with "trial and error" method. In this work it is proposed to apply genetic algorithm to facilitate the learning process. The ESN also allows for a very fast and efficient way of designing RNNs-based models. The ESN are a very convenient framework for using RNNs in practical engineering applications (e.g. [13, 16]).

3 Model Error Modelling

In this work, to carry out fault detection, the method which directly provides uncertainty descriptions for ESN models is proposed. Such problem of uncertainty included in model is referred to as robust identification [6, 14, 15]. In such framework the identification procedure should deliver not only a model of a given plant, but also a reliable estimate of uncertainty associated with the model. With such an approach it is possible to overcome many of the difficulties which lay in the idealized assumptions such as believing that the model of the system is a faithful replica of plant dynamics or that all disturbances and noises acting upon the system are known and are modeled during identification process or can be predicted/measured.

In this work the solution presented is based on idea to identify the process without robustness considerations first, and then consider robustness as an additional step [10, 15]. After that, one can estimate uncertainty of the model by analyzing residuals evaluated from the inputs. The uncertainty is a measure of unmodeled dynamics, noises and disturbances. Identification of residuals provides the so-called *error model*. Designing procedure is presented by Algorithm 1 (Figs. 1 and 2).

The uncertainty region is formed around the calculated center $\bar{y} + \tilde{y}$, and can also be called a confidence region. It consists of the upper band:

$$r_u = \bar{y} + \tilde{y} + t_\alpha v \tag{3}$$

Algorithm 1: Procedure of designing the uncertainty bands

for *each output i* **do**

Step 1. Using a model of the process, compute the residual $r_i = y_i - \bar{y}_i$, where y_i and $\bar{y}_i$ are measured and model outputs, respectively;

Step 2. Collect the data $\{\mathbf{u}(k), r_i(k)\}_{k=1}^{N}$ and for each residual identify an error model using these data. This model constitutes an estimate of the error due to under modeling, and it is called model error model (Fig. 1);

Step 3. For each error model derive the center of the uncertainty region as $\bar{y}_i + \tilde{y}_i \approx y_i$, where $\tilde{y}_i$ is the output of the error model (Fig. 2);

Step 4. Form the uncertainty region using statistical properties of the error model

end

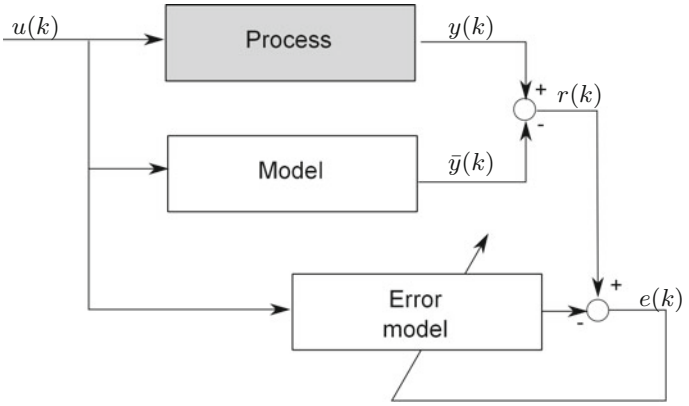


Fig. 1 MEM identification

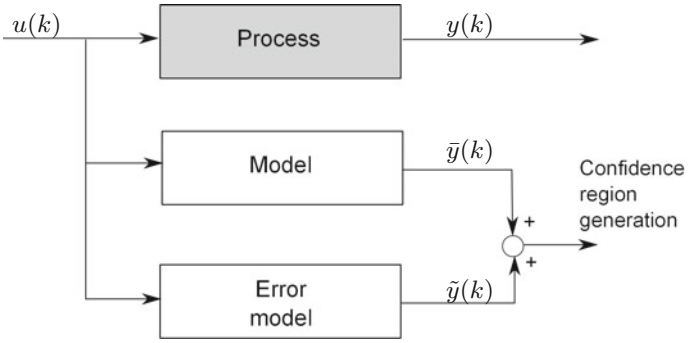


Fig. 2 MEM derivation of the center of the uncertainty region

and the lower band:

$$r_l = \bar{y} + \tilde{y} - t_\alpha v \tag{4}$$

where $\tilde{y}$ is the output of the error model on the input u, t_α is $\mathcal{N}(0, 1)$ tabulated value assigned to $1 - \alpha$ confidence level, and v is the standard deviation of $\tilde{y}$.

The error model can be realized using different techniques. The simplest way is to use the FIR filter, as discussed in [15], but also one can apply more complicated models, e.g. the ARX model or its nonlinear version. The model error modeling idea was successfully applied to design robust locally recurrent neural networks [10] or in previous works of the authors [4, 5] this idea was successfully applied to create robust state space neural model. In the present work, this methodology is used to design the robust echo state neural model using the NonLinear AutoRegressive with eXogenous input model (NLARX) available in the MATLAB software.

4 Two Rotor Aero-Dynamical System

The Two Rotor Aero-dynamical System is the laboratory stand designed for control experiments. In certain aspects its behavior resembles that of a helicopter. From the control point of view it exemplifies a high order non-linear system with significant cross-couplings. The system is controlled from a personal computer (PC). Therefore, it is delivered with both, hardware and software which can easily be mounted and installed in the laboratory. The laboratory setup consists of the mechanical unit with power supply and interface to a PC and the dedicated RTDAC/USB2 I/O board configured in the Xilinx technology. The software operates in real time under MS Windows XP/7 32-bit using MATLAB R2009/10,11, Simulink and RTW toolboxes. Real-time is supported by the RT-CON toolbox from INTECO. Control experiments are programmed and executed in real-time in the MATLAB/Simulink environment. The real-life installation is presented in Fig. 3, and the scheme of the system is presented in Fig. 4. The mathematical description of the plant symbols can be found in [7].

Fig. 3 Two rotor aero-dynamical system—laboratory setup

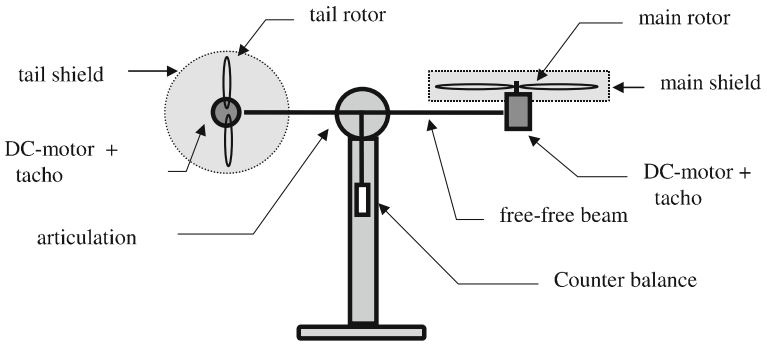


Fig. 4 Two rotor aero-dynamical system—parts scheme

5 Experiments

To apply a model of the system to any diagnostics task, the modeling phase is a very crucial one.

5.1 Neural Modelling

Incorrect model can lead to many problems, including weak detection performance or many false alarms. To build a proper model, the training data describing the process under normal operating conditions is required. The input signal should be as much informative as possible. In this paper the model is trained with the training data which were obtained during the spectral analysis of the TRAS (described in detail in [2]). The system was fed with the chirp signal and given response was analysed with Discrete Fourier Transform to obtain frequency components for which the system is the most responsive. Such approach to system modelling was with success applied to Model Predictive Control of TRAS in the previous work of the authors [3]. Taking into account determined frequency range, the training data in the form of 3 seconds long random steps were chosen. The data collected were 1000 seconds long. The sampling of 0.01 s gave 100,000 samples of data. Such high sampling is not needed so the data were resampled to 0.05 s which gave 20,000 samples. Then the collected data was divided into 4000 samples of training data and 16,000 samples of testing data.

During the following experiments the tail rotor was not used and the azimuth movement was blocked (the only task by the tail rotor is to compensate the azimuth movement, fault detection with MIMO system using cross-coupled controller will be the subject of the future research).

With correct training data it is possible to carry out the design of the model. The type of the ESN used in this paper is the so-called Leaky ESN [9]. Process of adjusting model parameters is described in Sect. 2. In many cases the adjusting process is carried out manually which often is a slow and ungrateful process. In this paper the approach based on the Genetic Algorithm is proposed to automate that process. The main task here is to define the cost function and optimization variables. The following settings have been used. Below are presented details of the genetic algorithm used:

- the initial population equal to 1000
- averaging genes based on recombination with 10 % mutation chance
- selection based on roulette rule with top 10 % of population preserved
- 500 iteration stop criterion
- the fitness function in the form of Mean Squared Errors (MSE).

Achieved set of the variables gave the performance of **0.017** for training data and **0.032** for testing data. The values of global parameters were found as follows:

- the spectral radius = 0.5612
- the input scaling = 0.3298
- the output feedback scaling = 0.0306
- the teacher scaling = 1.1047
- the reservoir size $N = 87$.

Results for training set are presented in following figures. The response of the system for the training data is presented in Fig. 5 and testing in Fig. 6.

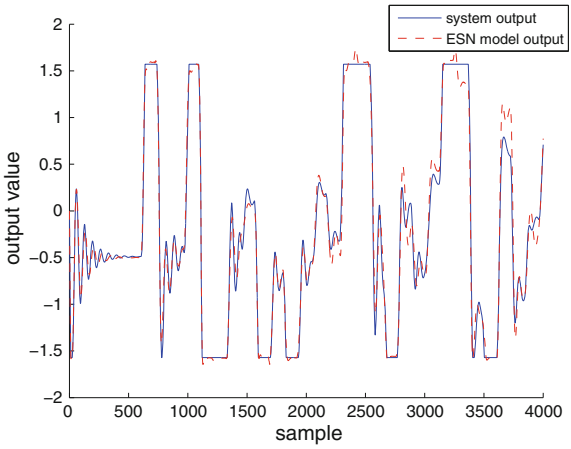


Fig. 5 Modelling results for the training set

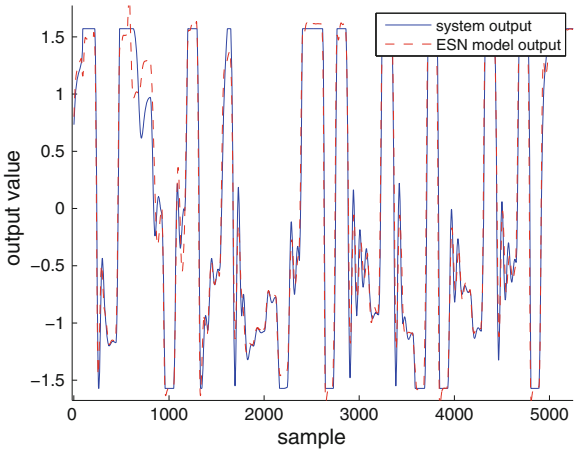


Fig. 6 Modelling results for the first 5000 samples of testing set

5.2 Fault Detection

In order to carry out fault detection the method described in Sect. 3 is applied. First step of Algorithm 1 is residual deviation. For this purpose the first experiment was performed (task of the control system was to follow random reference signals). Collected residuals were used to estimate the NLARX model. Different number of input (n_a) and output (n_b) delays were tested, and the quality of obtained models are listed in Table 1. All models were compared taking into account SSE index. The best result is marked with the box. As one can see, model with $n_a = 5$ and $n_b = 1$ was the best one. The very good quality of the model is proved in Fig. 7.

To perform robust fault detection it is necessary to form the uncertainty bands according to (3) and (4). The t_α was taken as 3 to maximize the detection accuracy and v was equal to $\mathbf{5.8513e-04}$. Due to very good modeling quality of the ESN

Table 1 Results of error modeling

Delays		SSE
n_a	n_b	
1	1	0.0010
1	3	0.0011
1	5	0.0011
3	1	$1.0241e-04$
3	3	$1.0429e-04$
5	1	$\boxed{8.0330e-05}$
5	3	$8.4804e-05$
6	1	$8.4992e-05$
6	3	$8.7905e-05$

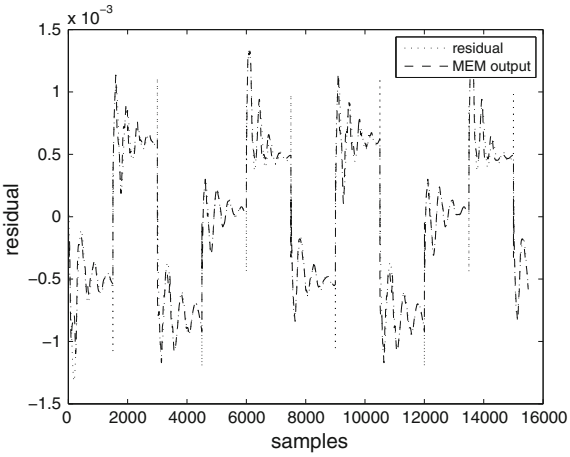


Fig. 7 Estimation of model error model

and very small error of the model, bands were so small that are almost not visible, which should result in very fast detection time. The example of the band formulation during nominal operating conditions of the plant is presented in Fig. 8. As one can see, the system output at every sampling moment stays inside uncertainty bands and false alarms are not raised.

Next, the experiments considering fault detection were carried on. The investigated fault scenario were simulated with manipulation of control signal (not visible to any part of detection system). The considered fault in the experiments was the power loss in main rotor which was simulated as a multiplicative type fault with 0.7 value.

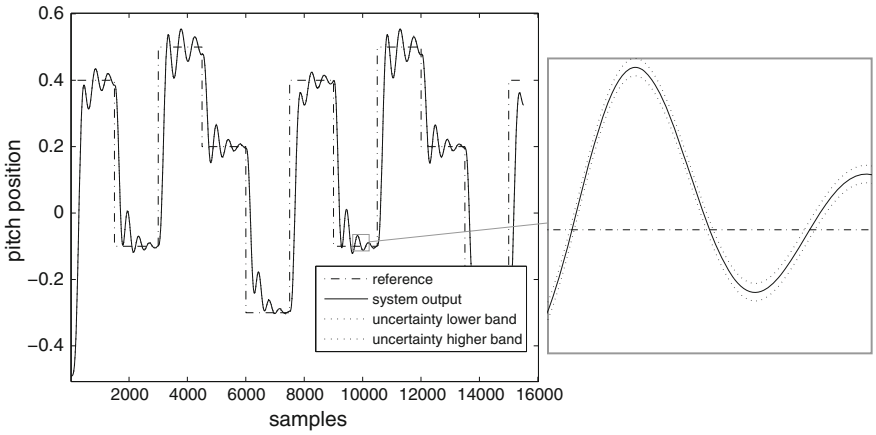


Fig. 8 Work of the fault detection system in nominal conditions

Finally, results of fault detection experiment are presented in Fig. 9. The task of controller was to sustain the pitch level of the plant at level of 0.5. At the 55 second (5500 sample) of experiment the fault was introduced. After only **0.78** s the system moved out of uncertainty bands which raised the fault alarm.

Also the analysis of the residual signal and output of the model error model which are presented in Fig. 10 gives interesting observations. Before the fault introduction the residual signal during the nominal work of the plant is nicely modeled with MEM and ESN model uncertainties are no false alarms are raised. When the real fault is introduced, the model error model is not compensating the change between model and system outputs so the real alarm can be signaled.

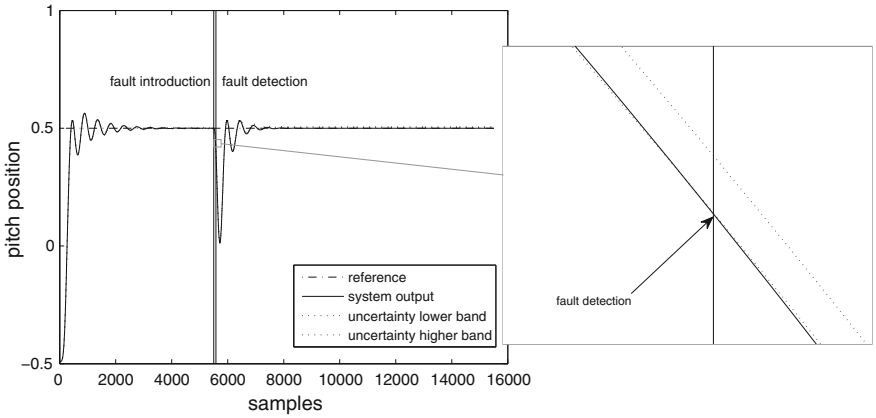


Fig. 9 Fault detection system working in case of the fault scenario

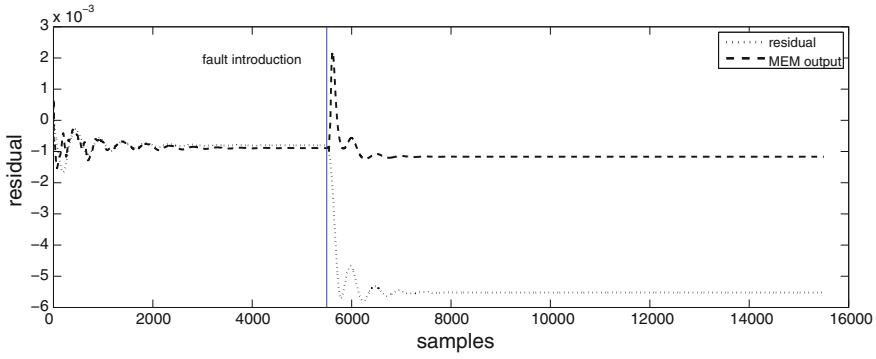


Fig. 10 Residual and MEM output in case of the considered fault

6 Conclusion

In this work the robust fault detection method by means of Echo State Network was proposed. As it was shown, this specific RNNs framework can easily and successfully be used in application of diagnostic system. The great modelling properties of RNNs are well known and thanks to ESN can be applied very easily. Also the Genetic Algorithm can be a very useful tool in case of obtaining the ESN global parameters. The proposed Model Error Modelling fault detection strategy seems to be very promising and needs to be tested outside the laboratory.

Our future work will be focused on using the proposed approach in case of both rotors of the TRAS and substituting the regular PID controller with MPC with adaptation of the model to achieve fault tolerance and to simulate more realistic control situation.

References

1. Bilski, P., Wojciechowski, J.: Artificial intelligence methods in diagnostics of analog systems. Int. J. Appl. Math. Comput. Sci. **24**(2), 271–282 (2014)
2. Czajkowski, A., Patan, K.: Designing nonlinear model of the two rotor aero-dynamical system using state space neural networks with delays. In: 18th International Conference on Methods and Models in Automation and Robotics, pp. 195–199. Międzyzdroje, Polska (2013)
3. Czajkowski, A., Patan, K.: Model predictive control of the two rotor aero-dynamical system using state space neural networks with delays. In: Korbicz, J.M.K. (eds.) Intelligent Systems in Technical and Medical Diagnostics, Advances in Intelligent Systems and Computing, vol. 230, pp. 113–124. Springer, Berlin (2013)
4. Czajkowski, A., Patan, K., Korbicz, J.: Stability analysis of the neural network based fault tolerant control for the boiler unit. Lect. Notes Comput. Sci. **7268**, 548 (2012)
5. Czajkowski, A., Patan, K., Szymański, M.: Application of the state space neural network to the fault tolerant control system of the PLC-controlled laboratory stand. Eng. Appl. Artif. Intell. **30**, 168–178 (2014)
6. Ding, L., Gustafsson, T., Johansson, A.: Model parameter estimation of simplified linear models for a continuous paper pulp digester. J. Process Control **17**(2), 115–127 (2007)
7. INTECO: Two Rotor Aero-dynamical System—User's Manual. http://www.inteco.com.pl (2012)
8. Jaeger, H.: The "Echo state" approach to analysing and training recurrent neural networks. In: GMD Report, vol. 148. German National Research Center for Information Technology (2001)
9. Jaeger, H., Lukosevicius, M., Popovici, D., Siewert, U.: Optimization and applications of echo state networks with leaky-integrator neurons. Neural Networks **20**(3), 335–352 (2007)
10. Patan, K.: Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes. Springer, Berlin (2008)
11. Patan, K.: Neural network-based model predictive control: fault tolerance and stability. IEEE Trans. Control Syst. Technol. **23**(3), 1147–1155 (2015)
12. Patan, K., Korbicz, J.: Nonlinear model predictive control of a boiler unit: a faul tolerant control study. Int. J. Appl. Math. Comput. Sci. **22**(1), 225–237 (2012)
13. Plöger, P.G., Arghir, A., Günther, T., Hosseiny, R.: Echo state networks for mobile robot modeling and control. In: Polani, D., Browning, B., Bonarini, A., Yoshida, K. (eds.) Robocup 2003: Robot Soccer World Cup Vii. Lecture Notes in Computer Science, vol. 3020, pp. 157–168. Springer, Berlin Heidelberg (2004)

14. Quinn, S.L., Harris, T.J., Bacon, D.W.: Accounting for uncertainty in control-relevant statistics. J. Process Control **15**(6), 675–690 (2005)
15. Reinelt, W., Garulli, A., Ljung, L.: Comparing different approaches to model error modeling in robust identification. Automatica **38**(5), 787–803 (2002)
16. Sheng, C., Zhao, J., Liu, Y., Wang, W.: Prediction for noisy nonlinear time series by echo state network based on dual estimation. Neurocomputing **82**, 186–195 (2012)

Part VI
Expert and Computer Systems

Towards Knowledge Compilation for Automated Diagnosis: A Qualitative, Model-Based Approach with Constraint Programming

Antoni Ligęza

Abstract The main idea of Consistency-Based Diagnosis rests in generation of diagnostic hypotheses stating which components of the system may be faulty, so that assuming them faulty explains the observations in a consistent way. Such diagnostic process is analyzed from qualitative perspective. Qualitative diagnostic inference, qualitative conflicts and qualitative diagnoses are presented in detail. Finally, we examine how qualitative knowledge can contribute to refinement of diagnostic inference and how compilation of diagnostic knowledge can be approached.

Keywords Automated diagnosis · Qualitative diagnostics · Model-based approach

1 Introduction

The main idea of Model-Based Consistency-Based Diagnosis [1] rests in generation of diagnostic hypotheses stating which components of the system may be faulty (abduction), so that assuming them faulty explains the current observations with the model in mind in a consistent way (deduction).

Existing methods of formal description of diagnostic inference are diversified. There are algebraic, graph-based, and logical expert-like and model-based diagnostic approaches. Some of the popular models include extended diagnostic matrices [2, 3], consistency-based reasoning [1, 4, 5], logical causal graphs [5–7], and many other [8, 9]. A recent survey of the so-called FDI and AI approaches is given by [10]. This paper explores mainly the AI approach and it is focused on some *qualitative and logical perspective* on diagnostic reasoning.

Research carried out within AGH University of Science and Technology statutory research 18.18.120.059.

A. Ligęza (✉)
AGH University of Science and Technology, al. Mickiewicza 30,
30-059 Kraków, Poland
e-mail: ligeza@agh.edu.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_26

The main point of interest in this work is the role of *qualitative inference* in automated diagnosis of technical systems. Perhaps one of the first models of qualitative algebra was introduced in [11]. Here, an in-depth analysis of qualitative concepts in modeling diagnostic reasoning is carried out. Moreover, a kind of *Constraint Problem Solving* approach is investigated with the ultimate goal of pruning inconsistent behaviors.

In qualitative modeling, inconsistency detection takes places when a variable (or component) is assumed to be operating in two different modes (such as *increased* and *decreased*) at the same time. Inconsistency is then eliminated by appropriate selection and refinement of diagnostic hypotheses. All potential diagnoses left must be consistent with all the other knowledge at hand.

The material presented in this paper is based in part on some previous works.[1] The concept of *Potential Conflict Structure* was first presented in [12]. It was further developed in [13, 14], and especially in the Ph.D. Thesis [15]. Basic logical material and the Reiter's theory were presented in [5]. The prototype of diagnostic procedure presented here and based on modeling the space of diagnostic hypotheses with AND/OR graphs is coming from [16]. The ideas on qualitative three-valued diagnoses have been introduced in [16], and the presentation here is based on [17]. The presented formal framework, new definitions and inference procedure based on the idea of constraint programming are new developments and they are presented mostly in Sect. 3.

2 Model-Based Diagnosis: State-of-the-Art

In this section we briefly recall a classical diagnostic example of a feed-forward arithmetic circuit. This is the multiplier-adder example presented in the seminal paper by Reiter [1]. This example was further re-explored in numerous papers, including selected readings [4] and diagnostic handbook (Chapter [5]). It was further explored in the discussion carried out in domain literature concerning the FDI and DX procedures along the [10, 18, 19]. Here we shall base on an in-depth analysis presented in [16], and also explored in [17]. The basic, intuitive schema of the system is presented in Fig. 1.

The system is composed of two layers. The first one contains three multipliers $m1$, $m2$, and $m3$, and receiving the input signals A, B, C, D and E. The second layer is composed of two adders, namely $a1$ and $a2$, producing the output values of F and G. Only inputs (of the first layer) and outputs of the system (of the second layer) are directly observable. The intermediate variables, namely X, Y and Z, are hidden and cannot be observed.

Observe that the current state of the system is defined by the input values; they are: A = 3, B = 2, C = 2, D = 3 and E = 3. It is easy to check—under the assumption of

[1] Of the author and his former Ph.D. student Barłomiej Górny, and some common work with Prof. Jan Maciej Kościelny.

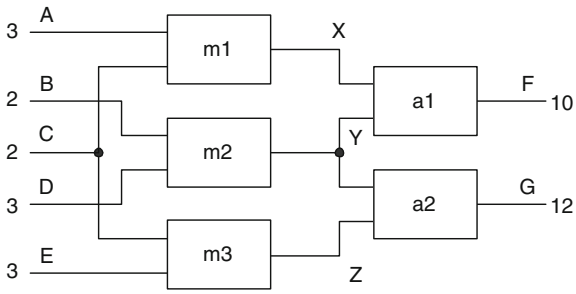


Fig. 1 Schema of the system

correct work of all the system elements—that the outputs should be F = 12 and G = 12. Note also that they *should be equal* to each other, which is due to the symmetry of the system and the symmetry of the input vales; this observation will be important for the analysis and we shall see later on why.

Now, since the current value of F is incorrect, namely F = 10, the system is faulty. At least one of its components must be faulty.[2] At this stage, for simplicity, we consider only *correct* components and *faulty* ones; no details about the type of fault are taken into consideration so far.

In order to perform diagnostic reasoning let us start with *abduction* [5, 16]; we shall try to build a conflict set specifying hypothetical faulty elements; in other word we search for a *Disjunctive Conceptual Faults; DCF for short* [16].

Note that the value of F is influenced by the inputs (observed) and the work of elements $m1, m2$ and $a1$ all of them are located in the signal path having direct influence on the value of F. In other words, one can say that there is direct *causal dependency* of influence of $m1, m2$ and $a1$ on F. If all the three elements work correctly, then the output would be correct. Since it is not, we can conclude that a DCF_1 is observed: at least one of the elements $\{m1, m2, a1\}$ must be faulty. Formally, we have

$$DCF_1 = \{m1, m2, a1\} \tag{1}$$

A further analysis leads to detection of DCF_2 [16]: under the assumed manifestations one of the elements $\{m1, a1, a2, m3\}$ must also be faulty.

In order to explain the origin of DCF_2 let us notice that if all the four elements were correct, then Z = C*E calculated by $m3$ must be equal to 6, and since G is observed to be 12, Y (calculated backwards and under the assumption that $a2$ works correctly) must also equal 6; hence, if $m1$ is correct, then X must be 6 as well, and if $a1$ is correct F would be equal to 12. Since it is not the case, at least one of the mentioned components must be faulty. So we have the following disjunctive conceptual fault:

$$DCF_2 = \{m1, a1, a2, m3\} \tag{2}$$

[2] In Model-Based Diagnosis it is typically assumed that faulty behavior is caused by a fault of a named component or a simultaneous fault of a set of such components; no faults caused by faulty links, parameter setting or the internal structure are considered.

Note that the type of DCF_1 and DCF_2 are different; this is due to their origin (or character). DCF_1 is of *causal* type—all the elements directly influence the conflicting variable. On the other hand DCF_2 is of *constraint* type; this is a kind of mathematical constraint which must be satisfied, but there is not necessary causal dependency between the components and the value of the faulty variable (F). This observation (and fault classification) will be used later on, when we pass to qualitative analysis.

Note that if both the outputs were incorrect (e.g. F = 10 and G = 14), then, in general case one can observe DCF_1, DCF_2 and DCF_3, where:

$$DCF_3 = \{m2, m3, a2\} \tag{3}$$

The DCF_3 is a *causal type* conflict.

Note also, that whether the constraint-type DCF_2 is a valid conflict may depend on the observed outputs. For example, if F = 10 and G = 10 (both outputs are incorrect but equal), then the structure and equations describing the work of the system do not lead to a conceptual fault [18, 19].

The composition of the three disjunctive conceptual faults can be presented with use of an OR-matrix for the diagnosed system (see Table 1): It defines the component elements for particular *DCF*-s for DCF_1, DCF_2, and DCF_3.

Having defined the *DCF*-s, the diagnoses are calculated as reduced elements of the Cartesian product of the currently valid ones [16]. The reduction consists in elimination of duplicates.

The AND-matrix defining the relationship between the *DCF*-s (active in the case of F being incorrect and G correct) and the manifestations is presented in Table 2. In the table F*, G*, etc. means that the output is incorrect, while F, G, etc. denotes correct output observed at the variable.

In the analyzed case, i.e. F being faulty and G correct, the final diagnoses for the considered case are calculated as reduced elements of the Cartesian product of

Table 1 An OR binary diagnostic matrix for the adder system (the lower level)

DCF	*m*1	*m*2	*m*3	*a*1	*a*2
DCF_1	1	1		1	
DCF_2	1		1	1	1
DCF_3		1	1		1

Table 2 An AND binary diagnostic matrix for the adder system (the upper level)

M	DCF_1	DCF_2	DCF_3
F*, G, (F-G)*	1	1	
F, G*, (F-G)*		1	1
F*, G*, F-G	1		1
F*, G*, (F-G)*	1	1	1

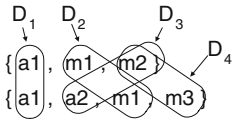


Fig. 2 Generation of potential diagnoses

Table 3 Final possible diagnoses

Manifestations	Diagnoses
F*,G, (F-G)*	$\{a1\}, \{m1\}, \{a2, m2\}, \{m2, m3\}$
F, G*, (F-G)*	$\{a2\}, \{m3\}, \{a3, m2\}, \{m1, m2\},$
F*, G*, (F-G)	$\{m2\}, \{a1, a2\}, \{a1, m3\}, \{a2, m1\},$ $\{m1, m3\}$
F*, G*, (F-G)*	$\{a1, a2\}, \{a1, m2\}, \{a1, m3\}, \{a2, m1\},$ $\{a2, m2\}, \{m1, m2\}, \{m2, m3\}, \{m1, m3\}$

$DCF_1 = \{m1, m2, a1\}$ and $DCF_2 = \{m1, m3, a1, a2\}$. There are the following potential diagnoses: $D_1 = \{m1\}$, $D_2 = \{a1\}$, $D_3 = \{a2, m2\}$ and $D_4 = \{m2, m3\}$. They all are shown in Fig. 2.

The final diagnoses for all possible general cases are presented with the following Table 3 [16].

Note that so far only binary faults were considered (i.e. a component may be faulty or not). In [16] and further in [17] an attempt to introducing qualitative diagnoses was undertaken. After calculation of possible binary diagnoses, their qualitative forms were considered, and with use of inference rules representing simple constraints inconsistent qualitative diagnoses were eliminated. In the next section an in-depth, qualitative analysis will be carried out. But this time we start from qualitative conflicts, i.e. move the qualitative analysis into the early stages of diagnostic reasoning.

3 Qualitative Approach to Knowledge Compilation

Let us start with the qualitative description of component behavior. In the basic, binary case recapitulated in Sect. 2, a component c could be just faulty or not. In the proposed qualitative approach, the component can behave in the following ways:

- $c(0)$—component c is correct; for intuition, 0 stands for *nominal behavior*
- $c(-)$—component c is incorrect, it lowers down the signal; for intuition, $-$ stands for *abnormal behavior with significant decrease of the value*
- $c(+)$—component c is incorrect, it increases the signal; for intuition, $+$ stands for *abnormal behavior with significant increase of the value*.

Such a representation will be more rich—in fact it provides more information on the type of component failure. Moreover, in some cases, it can be used to eliminate some

of the so-called *spurious behaviors* (e.g. a battery can be only normal or low; fuel consumption is typically increased when an engine is faulty, etc.).

For further use, the following logical notation will be introduced:

$$c(0|+) = c(0) \vee c(+)$$

$$c(0|-) = c(0) \vee c(-)$$

$$c(-|+) = c(-) \vee c(+)$$

Here, $c(\#)$, where $\# \in \{-, 0, +\}$ is considered as a *logical propositional statement*. Observe that only in the last case we are sure that the component is faulty; in the first two cases only a potential fault can be considered. Notation as such can be used to specify *constraints* on variable behavior.

Qualitative behavior can also be used for *inconsistency detection*, which is important in elimination of *spurious behaviors*—physically impossible ones. In the following cases an obvious inconsistency is detected:

$$c(0) \wedge c(-)$$

$$c(0) \wedge c(+)$$

$$c(-) \wedge c(+)$$

Analogous notation will be applied to characterize behavior of variables: $X(-)$ will say that the value of X is lower than expected, $X(+)$ will say that it is higher than expected, while $X(0)$ denotes the expected (correct) value. Finally, X^* will denote some conflict, i.e. $X(-)$ or $X(+)$ must hold.

3.1 Qualitative Diagnostic Approach

In this section we shall reconstruct the basic elements of Model-Based Diagnostic reasoning based on Consistency-Based Diagnosis, as presented in [1]. The study is aimed at providing an in-depth qualitative analysis of Model-Based Qualitative Reasoning.

First, consider a set of components such that under the current observations at least one of them must be faulty; this is the so-called *Conflicts Set* [1] or a *Disjunctive Conceptual Fault* [16]. For the sake of qualitative analysis a notion of *Qualitative Conflict*, or *Qualitative Disjunctive Conceptual Fault* is introduced. Let *SD* denote system description, *COMP* be the set of system components and *OBS*—the current observations.

Definition 1 A *Qualitative Conflict* (QC for short) or rather a *Qualitative Disjunctive Conceptual Fault* (QDCF) is any set of the form

$$QDCF = \{c_1(\#), c_2(\#), \ldots, c_k(\#)\}$$

such that under the current observations all the elements $c_1, c_2, \ldots, c_k$ cannot be working together correctly, and for $\# \in \{-, +, -/+\}$ the specification covers possible explanations of the observed behavior.

Formally, a set $\{c_1(\#), c_2(\#), \ldots, c_k(\#)\}$ is a $QDCF$ for $(SD, COMP, OBS)$ iff

$$SD \cup OBS \cup \{\neg AB(c_1(\#)), \neg AB(c_2(\#)), \ldots, \neg AB(c_k(\#))\}$$

is inconsistent.

Note that, from logical point of view a $QDCF$ forms a *disjunction*. In fact, it is a disjunctive statement that at least one of the components must be faulty. However, A $QDCF$ provides more information than classical *conflict set*: it gives further details on the types of possible deviations.

Consider the example of multiplier-adder from Sect. 2 and the particular case presented in Fig. 1. We have a conflict observed at the variable F, i.e. there is F(−). There are three elements having direct influence on the value of F, namely $m1, m2$ and $a1$. At least one of these elements must be faulty. But, moreover, it *must* lower down the signal value. Hence, the corresponding qualitative conflict will be of the form:

$$QDCF_1 = \{m1(-), m2(-), a1(-)\} \tag{4}$$

Since this is a *causal conflict*, the deviations are defined in a unique way, and so is the $QDCF$.[3]

Observe that, generally, under the assumptions that:

- a given conflict is minimal
- it is of causal type.

such a conflict will be defined in a unique, deterministic way.

A slightly different situation is when we consider conflicts of *constraint type*. For the case presented in Fig. 1 the second qualitative conflict is defined as:

$$QDCF_2 = \{m1(-|+), m3(-|+), a1(-|+), a2(-|+)\} \tag{5}$$

and it is not defined in a unique way. In fact, we have as many as 16 possible unique conflicts, i.e. $\{m1(-), m3(-), a1(-), a2(-)\}$, $\{m1(-), m3(-), a1(-), a2(+)\}$, $\{m1(-), m3(-), a1(+), a2(-)\}$, etc. until $\{m1(+), m3(+), a1(+), a2(+)\}$. Conflicts of such type define *de facto* constraints.

[3]Recall that we consider only minimal conflict sets and minimal diagnoses. In the other case, conflicts such as $\{m1(-), m2(-), a1(+)\}$ would also be possible, but removing a single element such as $m2(-)$ would not lead to regaining consistency.

Table 4 Definition of composition of qualitative values

	I1(−)	I1(0)	I1(+)
I2(−)	O'(−) (1)	O(−)	O'(?) (2)
I2(0)	O(−)	O(0)	O(+)
I2(+)	O'(?) (3)	O(+)	O'(+) (4)

Let us explain this in detail. $QDCF_2$ is specified to explain the inequality of the output values F and G. In fact, one cannot be sure what kind of influence the elements will have on the output values. As we shall see with an example, even if the output value is lowered down, there can be some necessity of some component to increase its value, for example to keep some observed inequality (this is the case of F(–), G(–) and F < G).

In general the diagnoses are calculated as minimal hitting sets (in fact: reduced elements of the Cartesian Product) of the $QDCF$-s.

Definition 2 A (minimal) qualitative diagnosis $D = \{d_1(\#\}, d_2(\#), \ldots, d_n(\#)$ is any minimal hitting set for all the $QDCF$-s, satisfying the following conditions:

- D is internally consistent (i.e. it does not contain a pair $d(-)$ and $d(+)$)
- D is consistent with observations, i.e.

$$SD \cup OBS \cup \{d(-|+)|d \in D\} \cup \{d(0)|d \in (COMP \backslash D)\}$$

is consistent.

In order to define consistency with observations one can use a table defining the qualitative result of composition of two input qualitative values on the third output value. Consider two input variables I1 and I2, and one output value, namely O. In Table 4, we present an *extended form* of the classical table [16].

The specific cases (1)–(4) are explained below.

1. $O'(-)$ denotes the output variable value in case of simultaneous decrease of both of the inputs; in such a case it is rational to assume that the following constraint holds:
$$O'(-) \leq O(-) \tag{6}$$

2. $O(?)$ can be $O(-)$, $O(0)$, and $O(+)$; in the first case it is rational to assume that the following constraint holds:
$$O'(-) \geq O(-) \tag{7}$$
while in the third case it is rational to assume that the following constraint holds:
$$O'(+) \leq O(+) \tag{8}$$

3. $O(?)$ can be $O(-)$, $O(0)$, and $O(+)$; in the first case it is rational to assume that the constraint given by (7) holds, while in the third case the constraint given by (8) should hold.
4. $O(+)$ denotes the output variable value in case of simultaneous increase of both of the inputs; in such a case it is rational to assume that the following constraint holds:

$$O'(+) \geq O(+) \tag{9}$$

The meaning of the constraints is intuitive. In general, two uni-directed influences should have an impact stronger or at least equal to the impact of any single influence selected from the pair of them. Similarly, a combination of opposite-directed influences should have impact weaker than or at least so strong as the impact of any of them observed alone. All the above constraints can be used to eliminate spurious behaviors.

4 Example

Let us consider a qualitative analysis performed with use of Qualitative Disjunctive Conceptual Faults of the classical Reiter example. Consider the situation as presented in Fig. 1.

As presented in Sect. 3 two QDCF can be observed; these are $QDCF_1 = \{m1(-), m2(-), a1(-)\}$, and $QDCF_2 = \{m1(-|+), m3(-|+), a1(-|+), a2(-|+)\}$.

Now, the qualitative diagnoses are constructed as reduced and consistent elements of the Cartesian Product of $QDCF_1$ and $QDCF_2$. Note that in fact there are as many as 3×16 such potential diagnoses. However, the analysis and diagnoses elimination is much faster than expected! Let us scan the elements of $QDCF_1$ and combine them with the ones of $QDCF_2$.

The case of $m1(-)$. Since $m1$ belongs both to $QDCF_1$ and $QDCF_2$, due to the minimality requirement only single element diagnoses are to be considered. In fact we have one consistent diagnosis $D_1 = \{m1(-)\}$.

The case of $m2(-)$. Note that we have only to consider combinations with $m3$ and $a2$ (the cases of $m1$ and $a1$ explored separately). Note that the influence of $m2(-)$ on G is negative, so there must be an extra fault compensating for that (the value of G is correct). This leads to the following two qualitative diagnoses, $D_3 = \{m2(-), m3(+)\}$ and $D_4 = \{m2(-), a3(+)\}$.

The case of $a1(-)$. Since $a1$ belongs both to $QDCF_1$ and $QDCF_2$, again, due to the minimality requirement only single element diagnoses are to be considered. In fact we have one consistent diagnosis $D_2 = \{a1(-)\}$.

5 More Complex Example

Let us consider a qualitative analysis performed with use of Qualitative Disjunctive Conceptual Faults of the classical Reiter circuit. Consider a situation different from the one presented in Fig. 1. Namely, let us assume that F = 10 and G = 11, i.e. both of the outputs are faulty, and moreover F ≠ G. Roughly speaking, F is *more faulty* than G.

In such a case three QDCF can be observed; these are $QDCF_1 = \{m1(-), m2(-), a1(-)\}$, $QDCF_2 = \{m1(-|+), m3(-|+), a1(-|+), a2(-|+)\}$ and $QDCF_3 = \{m2(-), m3(-), a2(-)\}$.

As before, the qualitative diagnoses are constructed as reduced and consistent elements of the Cartesian Product of $QDCF_1$, $QDCF_2$ and $QDCF_3$. Note that in fact there are as many as $3 \times 16 \times 3$ (nominally: 144) such potential diagnoses. However, the analysis and diagnoses elimination is much faster than expected!

The case of $m1(-)$. Since $m1$ belongs both to $QDCF_1$ and $QDCF_2$, due to the minimality requirement $m1(-)$ must be a component of the final diagnosis, and it must be combined with elements of $QDCF_3 = \{m2(-), m3(-), a2(-)\}$. In consequence, one obtains the following three qualitative diagnoses: $D_{11} = \{m1(-), m2(-)\}$, $D_{12} = \{m1(-), m3(-)\}$ and $D_{13} = \{m1(-), a2(-)\}$. It is interesting to observe the components of D_{11}: $m2(-)$ influences both F and G in a negative way, but F is also influenced by $m1(-)$; this is consistent with the constraint given by (6).

The case of $a1(-)$. Since $a1$ belongs both to $QDCF_1$ and $QDCF_2$, again, due to the minimality requirement $a2(-)$ must be a component of any qualitative diagnosis and it must be combined with elements of $QDCF_3 = \{m2(-), m3(-), a2(-)\}$. In consequence, one obtains the following three qualitative diagnoses: $D_{21} = \{a1(-), m2(-)\}$, $D_{22} = \{a1(-), m3(-)\}$ and $D_{23} = \{a1(-), a2(-)\}$. It is interesting to observe the components of D_{21}: $m2(-)$ influences both F and G in a negative way, but F is also influences by $a1(-)$; this is consistent with the constraint given by (6).

The case of $m2(-)$. Since the case of $m1$ and $a1$ have been explored, we have only to consider the case of $m2(-)$ (being a component of $QDCF_1$ and $QDCF_3$) with all the components of $QDCF_2$. Note that the influence of $m2(-)$ on G is negative, so there must be an extra fault compensating for that (the value of G is higher than the one of F). This leads to the following two qualitative diagnoses: $D_{31} = \{m2(-), m3(+)\}$ and $D_{41} = \{m2(-), a3(+)\}$. The diagnoses $\{m2(-), m1(-)\}$ and $\{m2(-), a1(-)\}$ have already been explored (see: D_{11} and D_{21} above). Note that in development of D_{31} and D_{41} the weakening constraints given by (7) play an important role.

6 Towards Knowledge Compilation

Knowledge compilation is a procedure consisting in transforming declarative knowledge into ready-to-use deterministic procedures. In case of diagnostic knowledge it means that an approach as described in this paper can be applied a priori, for each

Table 5 A table presenting all possible qualitative failure states

No.	F	G	F ~ G	Comment
1	−	0	F < G	F—incorrect; G—correct; F ~ G—incorrect
2	+	0	F > G	F—incorrect; G—correct; F ~ G—incorrect
3	0	−	F > G	F—correct; G—incorrect; F ~ G—incorrect
4	0	+	F < G	F—correct; G—incorrect; F ~ G—incorrect
5	−	−	F < G	F—incorrect; G—incorrect; F ~ G—incorrect
6	−	−	F > G	F—incorrect; G—incorrect; F ~ G—incorrect
7	−	+	F < G	F—incorrect; G—incorrect; F ~ G—incorrect
8	+	−	F > G	F—incorrect; G—incorrect; F ~ G—incorrect
9	+	+	F > G	F—incorrect; G—incorrect; F ~ G—incorrect
10	+	+	F < G	F—incorrect; G—incorrect; F ~ G—incorrect
11	−	−	F = G	F—incorrect; G—incorrect; F ~ G—correct
12	+	+	F = G	F—incorrect; G—incorrect; F ~ G—correct

assumed scenario, and retained for further use. Below, an outline of the steps necessary for diagnostic knowledge compilation is given.

Let us consider the complete table identifying possible combinations of qualitative conflicts Table 5.

Note that Table 5 can potentially contain as many as 27 rows (3 values for F times 3 values of G times 3 values for the comparison of F vs. G). However, the pattern (F(0), G(0), F = G) represents correct behavior (no conflicts observed) and eight other patterns where F = G, are internally inconsistent. And so are the other 6 patterns (F(−), G(+), F > G), (F(+), G(−), F < G), (F(0), G(−), F < G), (F(0), G(+), F > G), (F(−), G(0), F > G), (F(+), G(0), F < G).

Now, for any pattern of active conflicts defined by Table 5 one has the set of *Qualitative Disjunctive Conceptual Faults*, each for a single conflict. The patterns of the QDCF are defined as by Eqs. (4) and (5).

Having all such QDCF-s, it is straightforward to generate all *Qualitative Diagnoses*. Obviously, these are only potential diagnoses, but the set of them covers all possible causes of the observed failure.

7 Conclusions and Further Work

In this paper an in-depth qualitative analysis of diagnostic inference was carried out. It seems that using qualitative reasoning can provide a valuable contribution both to better understanding the nature of faults and their propagation, as well as to more efficient diagnostic procedures; in fact, some spurious behaviors can be eliminated with use of qualitative constraints.

Several new points have been raised. These include *Qualitative Conflicts* and qualitative constraints. The conflicts have been classified into *causal* ones and *constraint* type. As a consequence, the definition of qualitative potential diagnoses and their generation has been restated.

As for future work, the following directions seems promising: (i) research towards efficient decomposition of the system, so as to reduce the combinatorial explosion; in order to do that (ii) definition of new check/measurement points seems one of most obvious ways; it should take into account that (iii) reduction of *diagnosis entropy* can be a valuable indicator for rational choice of new measurements. As for diagnostic knowledge compilation it seems worth considering applications in on-line, real-time systems—to increase their robustness, and perhaps in other embedded/autonomous systems working under temporal constraints in challenging and critical environments, such as space or military applications.

References

1. Reiter, R.: A theory of diagnosis from first principles. Artif. Intell. **32**, 57–95 (1987)
2. Kościelny, J.M.: Methodology of process diagnosis, chap. 3. In: Korbicz, J., Kościelny, J., Kowalczuk, Z., Cholewa, W. (eds.) Fault Diagnosis. Models, Artificial Intelligence, Applications, pp. 57–114. Springer, Berlin (2004)
3. Kościelny, J.M.: Models in process diagnosis, chap. 2. In: Korbicz, J., Kościelny, J., Kowalczuk, Z., Cholewa, W. (eds.) Fault Diagnosis. Models, Artificial Intelligence, Applications, pp. 29–43. Springer, Berlin (2004)
4. Hamscher, W., Console, L., de Kleer, J. (eds.): Readings in Model-Based Diagnosis. Morgan Kaufmann, San Mateo (1992)
5. Ligęza, A.: Selected methods of knowledge engineering in system diagnosis, chap. 16. In: Korbicz, J., Kościelny, J., Kowalczuk, Z., Cholewa, W. (eds.) Fault Diagnosis. Models, Artificial Intelligence, Applications, pp. 633–668. Springer, Berlin (2004)
6. Fuster-Parra, P.: A model for causal diagnostic reasoning. Extended inference modes and efficiency problems, (Ph.D. Thesis). University of Balearic Islands, Spain, Palma de Mallorca (1996)
7. Ligęza, A., Fuster-Parra, P.: And/or/not causal graphs—a model for diagnostic reasoning. Appl. Math. Comput. Sci. **7**(1), 185–203 (1997)
8. Korbicz, J., Kościelny, J., Kowalczuk, Z., Cholewa, W. (eds.): Fault Diagnosis. Models, Artificial Intelligence, Applications. Springer, Berlin (2004)
9. Davis, R., Hamscher, W.: Model-Based Reasoning: Troubleshooting, pp. 3–24. Morgan Kaufmann Publishers, San Mateo (1992)
10. Travé-Massuyès, L.: Bridges between diagnosis theories from control and AI perspectives. In: Korbicz, J., Kowal, M. (eds.) Intelligent Systems in Technical and Medical Diagnosis. pp. 3–28. Springer (2014)
11. Travé-Massuyès, L., Piera, N.: The order of magnitude models as qualitative algebras. In: Sridharan, N. (ed.) Proceedings of the International Joint Conference on Artificial Intelligence, pp. 1261–1266. Morgan Kaufmann Publishers Inc. (1989)
12. Ligęza, A.: A note on systematic conflict generation in Ca-en-type causal structures. LAAS Report, No. 96317, pp. 1–22 (1996)
13. Górny, B., Ligęza, A.: Model-based diagnosis of dynamic systems: systematic conflict generation. In: Magnani, L., Nersessian, N.J., Pizzi, C. (eds.) Model-Based Reasoning, Scientific Discovery, Technological Innovations, Values, pp. 273–291. Kluwer Academic Publishers (2002)
14. Ligęza, A., Górny, B.: Systematic conflict generation in model-based diagnosis. In: Edelmayer, A.M. (ed.) 4th IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes, vol. 2, pp. 1103–1108 (2000)
15. Górny, B.: Consistency-based reasoning in model-based diagnosis, (Ph.D. thesis). AGH (2001)

16. Ligęza, A., Kościelny, J.M.: A new approach to multiple fault diagnosis. Combination of diagnostic matrices, graphs, algebraic and rule-based models. The case of two-layer models. Int. J. Appl. Math. Comput. Sci. **18**(4), 465–476 (2008)
17. Ligęza, A.: A constraint satisfaction framework for diagnostic problems. In: Kowalczuk, Z. (ed.) Diagnosis of Processes and Systems, pp. 255–262. Pomeranian Science and Technology Publishers (PWNT), Gdańsk (2009)
18. Cordier, M.O., et al.: AI and automatic control approaches of model-based diagnosis: links and underlying hypotheses. In: Edelmayer, A.M. (ed.) 4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, pp. 274–279 (2000)
19. Cordier, M.O., et al.: A comparative analysis of AI and control theory approaches to model-based diagnosis. In: Horn, W. (ed.) 14th European Conference on Artificial Intelligence, pp. 136–140. IOS Press (2000)

Development of Expert System Shell with Context-Based Reasoning

Dominik Wachla, Piotr Przystałka, Mateusz Kalisch, Wojciech Moczulski and Anna Timofiejczuk

Abstract The paper focuses on the expert system shell which is proposed as a tool that can be used for a wide spectrum of industrial applications. A new architecture of the system enables reasoning by means of multi-domain knowledge representations and multi-inference engines. Moreover, the extended functionality of the system is developed using a context based approach. The system is implemented applying a data mining software which makes possible to acquire domain-specific knowledge and its direct application in the expert system shell. In this study, the preliminary verification is presented using the data registered by the SCADA system of the water supply network. The case study results are useful to illustrate the merits and limitations of the proposed approach.

Keywords Expert systems · Context-based reasoning

1 Introduction

Expert systems are a product of the *Artificial Intelligence* [5, 11, 14] and they are used in different domains of human activities. They were and they are designed for solving specialized tasks requiring expert knowledge. The expert systems consist of a knowledge base, an inference engine, a user interface and an explanation facility module [11, 14]. The current research conducted in this field focus on developing expert system shells [25]. The expert system shell is an expert system where knowledge base is empty and must be defined by an expert from domain of interest. One can distinguish expert system shells by means of the symbolic- and the instance-base knowledge representation and reasoning [25]. In case of an expert system shell with the instance-base knowledge representation and reasoning, the inference mechanism is based on hierarchical reasoning realized by multiply classifiers such as decision

D. Wachla (✉) · P. Przystałka · M. Kalisch · W. Moczulski · A. Timofiejczuk
Silesian University of Technology Institute of Fundamentals of Machinery Design, Konarskiego 18a Str., 44-100 Gliwice, Poland
e-mail: dominik.wachla@polsl.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_27

trees, neural networks, support vector machines and others which form a knowledge base of the system. This approach has numerous advantages, e.g.:

- there exist many tools supporting classifiers induction and data exploration, thus the knowledge acquisition [15] is relatively easy
- the reasoning can be combined by means of connecting multiple classifiers and it can include a context (Sect. 2) as well
- in many cases data describing processes and systems are available.

The main problem with instance-based expert system shells is interaction with the end-user by means of statements. The statement [5–7] is a sentence expressed in a natural language (Sect. 5). In the symbolic-based expert systems shells, i.e. the rule-based expert systems, this problem does not exist because rules are constructed using statements [5]. Then, statements are shown directly during a dialogue with the end-user as a result of reasoning or as a part of reasoning explanation. To overcome this problem, a dictionary of statements (Sect. 5) can be applied. In this solution, classes of a classifier are connected together with appropriate statements using labels.

Taking into account all of the above, the instance-based expert system shell was proposed. The novelty of our approach consists in application of context [4, 10] and statements [5–7]. Introductory investigations (Sect. 6) seem to prove that application of context and statements allows to obtain not only better classification results, but also the description of the system reasoning is more accurate and more user-friendly as well.

2 Context Definition

In the classification tasks all available features of process variables are usually divided into three groups: primary, contextual and irrelevant features [23]. The primary features are directly applied inside a classification algorithm to obtain the final result of classification. It is also possible to distinguish two types of the primary features, such as context-sensitive and context-insensitive features. The contextual features can not be used directly by the classifier because it does not contain any diagnostic information about the current state of the monitored process. However, they can be useful when they are combined with context-sensitive features. The irrelevant features are not useful for classification, neither when combined with other features nor when they are considered alone. The context can be known and connected with one or more process variables as well as it can be hidden (e.g. when it is not known). It is possible to use evolutionary algorithm [20, 21] or other methods [13] to extract an unknown context from the data. The contextual feature is a continuous or discrete variable connected with a specific object. In case of the discrete contextual variable, a contextual value is equal to one of all available contextual variants connected with this variable. The contextual variant can be obtained from the continuous contextual feature by the classifier or can be specified by an expert.

The context can be represented by a text message describing its purpose. The part of the message should be connected with a contextual variable and another part with contextual variant. The example of contextual message can be *Temperature is normal* or *Temperature of the air is high*. The first part of the message (*Temperature of the air is*) is connected with a contextual variable and the second part (*normal* or *high*) is connected with different variants of the same contextual variable. The contextual message helps experts and users to use them, because it is easier to work with descriptions than labels used by the classifiers.

Turney in [22, 23] presents five various approaches of usage of context variables in machine learning methods. These methods are based on *Contextual normalization*, *Contextual expansion*, *Contextual classifier selection*, *Contextual classification adjustment* and *Contextual weighting*. For example, in the paper [22], *Contextual normalization* was proposed in order to achieve the high performance of fault diagnosis of the gas turbine engine. In this case, the contextual variables were connected with weather conditions such as an external temperature of the air, barometric pressure and humidity measured by the sensors installed outside of the engine.

3 System Requirements and Assumptions

The development of the diagnostic expert system shell with multi-domain knowledge representations and multi-inference engines is realized within the frame of DISESOR project which is a decision support system designed mainly for fault diagnosis of machinery and other equipment operating in underground mines as well as for monitoring potential threats that can occur in such kind of industry. However, the authors also view the proposed environment as a tool that can be employed for a wider spectrum of industrial applications, and therefore it is considered to apply this software for solving analogous problems and tasks. It is assumed that the expert system shell module will be used for on-line and off-line diagnosing of technical objects and for monitoring processes. Another task of the module is to support domain experts in taking decision either on terms of the technical condition of the objects or on risk managing e.g. in situations when the process is going to an undesirable direction. The main assumption in case of operation of this module is to allow the user to acquire intuitively the knowledge and store the knowledge with use of different types of representations. The expert system shell is designed in such a way that the reasoning can be based on elementary methods such as classic, possibility and probability theory logics, etc.

Using a dictionary of statements and contexts is another assumption of the proposed expert system shell. The main objective is to present results of reasoning in a form that is readable for a user, i.e. in a form of statements [5]. In view of main assumptions of the DISESOR project, the dictionary of statements and context is going to be organized as a database unit.

4 Major Functionalities of the Proposed Expert System Shell

The case diagram of the proposed expert system shell is given in Fig. 1. The system consists of two layers. The first layer is called a management layer and it is used to supervise the whole system. There is included the main use case for the mode selection that can be used to switch a system into one of three different modes. The on-line mode mainly serves in situations when the scheduler of the reasoning process is created and executed in real time or when the end user demands this process with the use of a web-service application. In this mode a knowledge engineer is also able to observe the parallel execution of the scheduler logic. The off-line mode is often applied for ad-hoc reasoning on historical data. The last mode is necessary for editing the knowledge base.

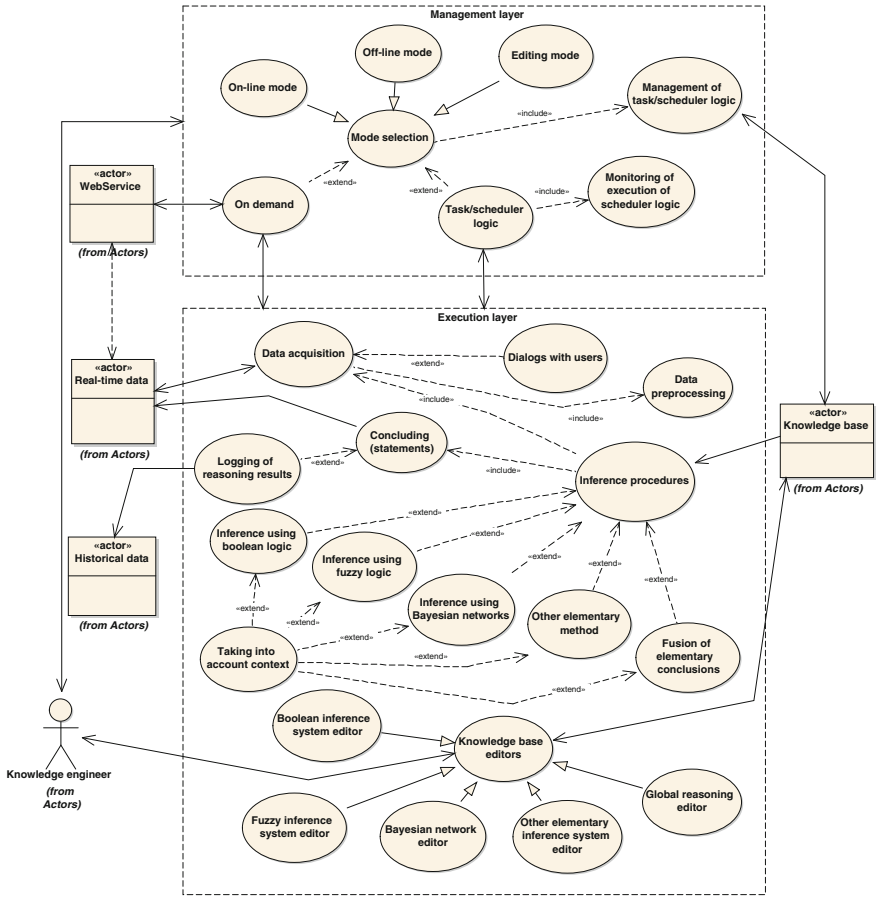


Fig. 1 A use case model of the proposed system

The engine of the system is implemented in the second layer (the execution layer). The user is able to clearly edit the knowledge base using different knowledge representations which are related to classic, possibility and probability theory logics. In the current version of the system inference engines are mainly developed using Boolean logic, fuzzy logic and Bayesian networks. Moreover, the engine of the system is designed in such a way as to allow for applying other elementary approaches belonging to classic, soft computing and artificial intelligence methods (e.g. artificial neural networks, neuro-fuzzy systems, rough sets, fuzzy-rough sets, etc.). The next functionality of the system is that, the user can also define the vocabulary of statements (and contextual statements) in order to prepare the description of a monitored object.

As it was stated above, the execution layer applies multi-domain knowledge representations for reasoning by means of the multi-interference engine. Inference procedures are executed using selected reasoning mechanisms in order to prepare the conclusion. It is also assumed that a context may be taken into account e.g. in the condition part of the rules or input nodes of the Bayesian network (it is an extended use case). In the next step the elementary conclusions are subject to the fusion process. The output of the system (statements) is obtained for measured and user data. The last use case of the system is the possibility to record the results of reasoning in order to realize the explanation interface.

It was decided to apply RapidMiner software as the engine of an expert system shell. RapidMiner is a partially free software developed to solve data mining problems [1]. This software has a clear user interface based on a drag and drop methodology. The application allows users to modify a source code and add plug-ins created by other software engineers to extend functionality of the program. The authors decided to utilize this software since there are implemented learning methods which can be used for knowledge acquisition. An additional factor is that the architecture presented in Fig. 1 can be obtained in a direct way by creating new plug-ins for this software.

5 Dictionary of Statements and Contexts

In expert systems, statements [5] are used to interact with the system end-users. The statement is a declarative sentence resulting from observed facts or representing an opinion [6, 7]. It means that the statement is expressed in a natural language, i.e. in a form that is understandable for man. The truth of the statement can be defined by means of the boolean value, i.e. 1 (*truth*) or 0 (*false*) or by an approximate value coming from range $\langle 0; 1\rangle$ which is called *the degree of truth* or *the degree of belief* [6, 7]. Three examples of a statement are presented below [5–7]:

$$\textit{Apple is green} \tag{1}$$

$$\textit{Temperature of the plain bearing bushing is less than } 40\,^{\circ}C \tag{2}$$

$$\textit{Temperature of the plain bearing bushing is greater or equal } 40\,^{\circ}C \tag{3}$$

A statement s can be formally defined by the following quadruple [5–7]:

$$s = \langle o, a, v, b \rangle \tag{4}$$

where o represents an object, a is an attribute describing the object, v is a value of the attribute and b is a belief factor describing the truthfulness level of the statement.

The statement can be also defined as [6]:

$$s = \langle c, b \rangle \tag{5}$$

where c is the statement content expressed in a form understandable for man, i.e. as a sentence of a natural language.

Statements can be univariate or multivariate, e.g. statements (2) and (3) [6]. Taking into account the (5), an n-variate statement can be defined as follows [6]:

$$s = \langle \underline{\mathbf{c}}, \underline{\mathbf{b}} \rangle = \langle c_{1:n}, b_{1:n} \rangle \tag{6}$$

where $\underline{\mathbf{c}}$ is an n-dimensional vector of the statement contents presenting each option of the statement, $\underline{\mathbf{b}}$ is an n-dimensional vector of values of the belief factor for all options of the statement. In a similar way, the statement notation (4) can be extended by means of the notation (6):

$$s = \langle o, a, \underline{\mathbf{v}}, \underline{\mathbf{b}} \rangle = \langle o, a, v_{1:n}, b_{1:n} \rangle \tag{7}$$

The analysis of many different statements shows that the statement content can not be constructed directly from the triple $\langle o, a, v \rangle$. This is due to the syntax and grammar rules of the natural language in which the content of the statement is expressed. On the other hand, the statement content can be partitioned into few parts describing separately the object, the object attribute, the value of the attribute and the expression presenting the nature of a relation between the object and the attribute. For example, in the statement (1) the object is *apple* and the content describing the object is also *Apple*. The attribute describing the object is conjectural and it is a *color*, thus the content describing the attribute is empty. Subsequently, the value of the attribute is *green* and the content of this value is also *green*. Finally, the relation between the object and the attribute is defined by the verb *to be*. The analysis of exemplary statements allowed to discover five schemes of the statement content construction by means of the content partitioning rules presented above.

The definition of the context (Sect. 2) states that the context is a special case of the statement. It means that the presented method of the statement decomposition and description can be applied to the context as well.

Based on the expert system shell assumptions (Sect. 3) and requirements of the statement content description, a database structure of a dictionary of statements and contexts was defined. Figure 2 presents a simplified version of this structure. For the purpose of describing multivariate statements, fundamental tables and the relation between them were identified.

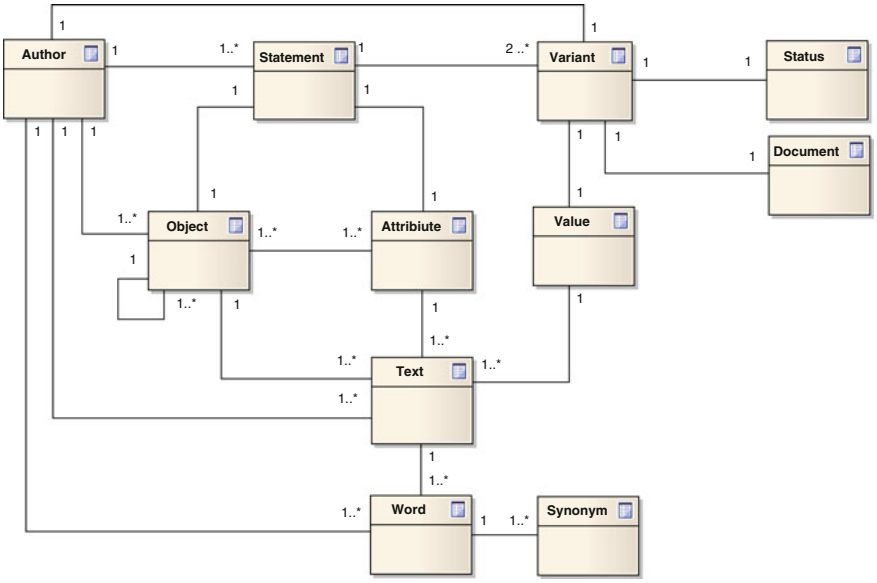


Fig. 2 A simplified database structure of a dictionary of statements and contexts

The main table of the dictionary database is table *Statement*. The *Statement* table includes a label of the statement, identifiers of an object and an attribute specified by the statement content and other fields important to describing the statement in a general way. Tables *Object* and *Attribute* contain data describing objects and attributes in details. By means of the use of the dictionary of statement in the expert system shell, the table *Variant* is essential, as well. Among other things, a single record of the table *Variant* includes an identifier, a scheme of the statement content reconstruction from elements describing the object, the attribute and the value, respectively. Furthermore, the table includes an expression presenting the nature of a relation between the object and the attribute in the statement content and identifier of the statement, the variant belonging to and identifer of the value of the variant.

The table *Text* is a table where elements of the statement content describing the object, the object attribute and the attribute value are stored to form a complete content of the statement in future. The functionality of the dictionary of statements and contexts is extended to work with synonyms. It is realized by means of two tables, i.e. the table *Word* and the table *Synonym*. The table *Word* is used to store user-selected words coming from the statement content while the table *Synonym* is used to define relations between words from the table *Word* and it constitutes definitions of synonyms.

6 Case Study

The subject of a case study in this paper is focused on the problem of leak detection in large-scale water distribution systems. Water distribution networks are critical to industrial and individual customers. In such systems, there are uncontrolled leaks which are caused by various factors. The most common origins of leaks are assembly errors, mechanical damages of pipes caused by overloads, fatigue, or normal wear and tear, material defects in parts of pipelines, corrosion, etc. [3], [19]. The consequence of such damage is a direct hazard to human health and life, e.g. as a result of local contamination of water. Another very important aspect is the economic loss [8]. There are studies showing that water losses may range from 3 to 7 % of the total consumption of the network in highly-developed countries as well as 50 % and more in networks with low technical standards [19]. In Poland, the average water losses are estimated to be below 19 %, however there is noticeable large variation in these estimates depending on the region [3, 12]. According to the report of the Supreme Audit Office from 2011 [17], the lowest and highest water losses within 2008–2010 in various regions of Poland were between 11 and 69 %. In relation to the same report, it is considered that the losses at 34 % of the total abstraction of water are treated as a waste.

The above analysis confirms that there is a need for continuous monitoring of such systems in terms of the detection and location of leaks and contamination of water. Leak detection in on-line diagnostic systems is currently carried out based on methods of analysis of water flow variability in the supply of a network at night [19]. Despite the high level of efficiency such an approach is limited due to a time delay in the leakage detection. There are still developed methods that can provide a solution to the problem of rapid detection and exact location of the leak at an early stage of their occurrence (e.g. in situations where leaks do not manifest themselves on the surface of the ground). The detailed description of the most important approaches can be found in [9, 19]. The complex methodology deals with this problem was also proposed and described in papers [16, 18] and monograph [24].

The related methods require continuous updating of the structure of a model and its parameters due to the changing characteristics of the system at the micro and macro time of maintenance (e.g. seasonal variations caused by changes of the seasons, the expansion of the water supply system, replacement of network elements, etc.). The result is that these methods are difficult to apply in on-line diagnostic systems that must operate automatically. Furthermore, the approaches that are now developed focus their attention only on the detection of occurred leaks without taking into account the reasons that lead to such a state. In the present study, the authors attempt to apply the proposed expert system shell for designing the diagnostic assay that can be employed to detect anomalies in flow signals which may be symptoms of faults such as leakages.

The structure of the water supply network under consideration is shown in Fig. 3. This is a network covering industrial and individual consumers in the selected district in a city of the southern region of Poland. The total length of pipelines of the

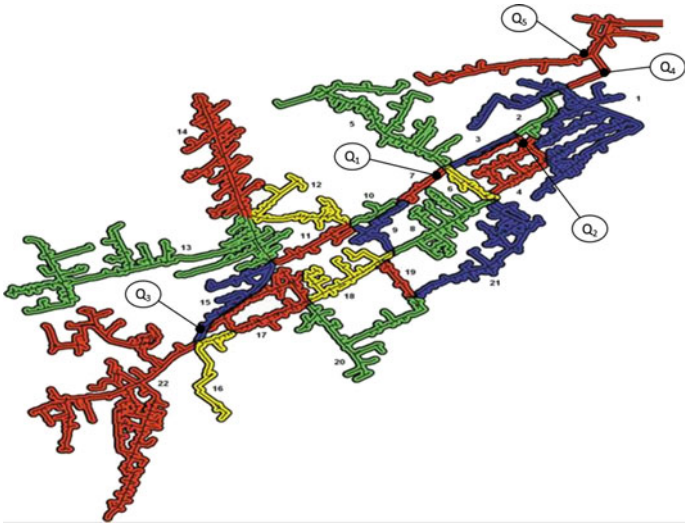


Fig. 3 The considered structure of the water distribution network

network exceeds 25 km. The network provides water to over a thousand customers. The locations of flow meters are marked in Fig. 3. The water supply pipe is monitored by the flowmeter labelled Q_4. The rest of flowmeters are used to collect instantaneous measurements of water flows in the critical points of the network. The colours and numbers in this figure show the predefined subareas of the network identified by the domain expert for leakage location purposes, but this is not studied in this paper. Measurement data collected by SCADA system for a selected district represent the operation of water supply system from 2010-01-04 to 2010-02-11 ($\Delta t = 15$ min.). In this period of time a leakage was observed that was caused by a pipe damage.

The first step in designing a diagnostic test for anomaly detection is focused on the preparation of statements. In the natural way, the input statements s_{1-5} are elaborated using process variables Q_{1-5} (the analysed network is an object of these statements, flow water variables are attributes and flow rates are values of attributes). The output statement s_6 is described as follows:

- o: the water distribution network in a southern Polish city
- a_6: instantaneous measurements of distribution of water flows in the network
- v_{61}: normal state with a belief factor b_{61}, v_{62}: anomalous state with a belief factor b_{62}
- c_{61}: instantaneous measurements of distribution of water flows in the network are the symptom of the normal state, c_{62}: instantaneous measurements of distribution of water flows in the network are the symptom of the anomalous state.

A deeper analysis of flow measurements for faultless state in time and amplitude domains allows to identify the contextual variable that can be used in the reasoning

process. This analysis is done taking into account time series of flow measurements (Fig. 4a) and a 2D histogram of flow rates prepared for Q_5 (Fig. 4b).

Thanks to this it is possible to observe the context corresponding to the form of flow rates as a function of a period of twenty four hours. For instance, from this plot it should be noted that the context is changed during the whole day (1: the flow is decreased about midnight, 2: the flow has a small magnitude and dispersion at night, etc.). This analysis can be summarized in order to formulate the contextual statement S_7 as follows:

- o: default object
- a_7: time of the day (contextual variable)
- v_{71}: about midnight (23:00–1:45), v_{72}: at night (2:00–4:45), v_{73}: at early morning (5:00–8:45), v_{74}: from the morning to the late afternoon (9:00-18:45), v_{75}: in the evening (19:00–22:45)
- c_{71}: the time is about midnight, c_{72}: it is the night, c_{73}: it is the early morning, etc.

In the next step it is necessary to acquire the knowledge that is needed by inference engines implemented in the expert system shell. In this paper, four kinds of the knowledge representation are applied and compared. In particular, classic and soft computing techniques such as decision tree (DT), artificial neural network (ANN), naive Bayes (NB) and Bayesian network (BN) were utilized. The structures of these models and their behavioural parameters were tuned using trial and error procedure. In a decision tree method the best results were achieved for C4.5 algorithm (for default parameter values). A multilayer neural network with eleven and five sigmoidal neurons in the first and second hidden layer, respectively, was enough to obtain the smallest classification error. A backpropagation with momentum technique was used to train this model (the learning rate was equal to 0.3, the momentum factor was set to 0.2, the total number of iterations was equal to 100). A naive Bayes model was created using the kernel density estimation mode with the minimum bandwidth set to 0.1 and the number of kernels set to 20. A Bayesian

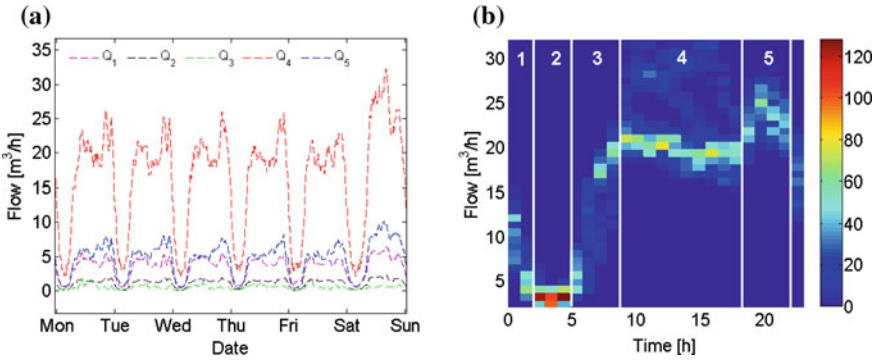


Fig. 4 Time and amplitude domain data analysis of flow measurements

network was designed applying simulated annealing algorithm with default values of the behavioural parameters.

The collected data was divided into two separated subsets (training and test datasets). The training dataset corresponds to a faultless state was used to create each model. In this data artificial faults were introduced in order to have representative and well balanced subsets of training patterns. Cross-validation technique (X-Validation) with the stratified sampling method (the number of validations was equal to 10) was used in order to estimate the accuracy of each classification model. On the other hand, the test dataset with real samples related to either faultless or faulty states was applied to compute the true and false detection rates. The inputs of the considered types of models were process variables Q_{1-5} and their features (such as time domain energy, kurtosis, entropy, etc.). The output of each model was the diagnostic signal relating to the symptom of anomalies. The following cases were studied. The first one was based on classification models without applying the contextual variable. The second case dealt with classifiers in which the additional input was declared by means of the contextual variable. In the last one the contextual variable was used as a switch for meta-classification models.

The most important results obtained by the authors in this case study are summarized in Table 1. It can be seen that at cross-validation stage the highest values of accuracy were obtained for DT and ANN based classifiers, however it was no matter if the context was applied or not. A much more important conclusion can be stated taking into account results for the test data with a real leakage. As one can see, the accuracy increased about 5 % for ANN models in the cases of using the contextual variable as the additional input or as a switch and it increased about 7−8 % for NB and BN models in the 3rd case only. Hence, it can be stated that the context based approach may lead to minimize the probability of overfitting in these kinds of models.

Some exemplary results obtained for a meta-classification neural model with a contextual switch are presented in Fig. 5 in order to illustrate the high detectability features of the proposed approach. The first plot shows water flow rates registered from 2010-01-24 (Mon) to 2010-02-05 (Fri). In the maintenance event log it was noted that the detection time of a leakage was on 2010-02-03 21:00:00 (Wed), the

Table 1 The performance results for anomaly detection

		Case No. 1	Case No. 2	Case No. 3
DT	X-validation	0.93	0.93	–
	Test	0.87	0.86	0.82
ANN	X-validation	0.94	0.93	–
	Test	0.79	0.84	0.84
NB	X-validation	0.87	0.87	–
	Test	0.78	0.78	0.85
BN	X-validation	0.87	0.87	–
	Test	0.77	0.77	0.85

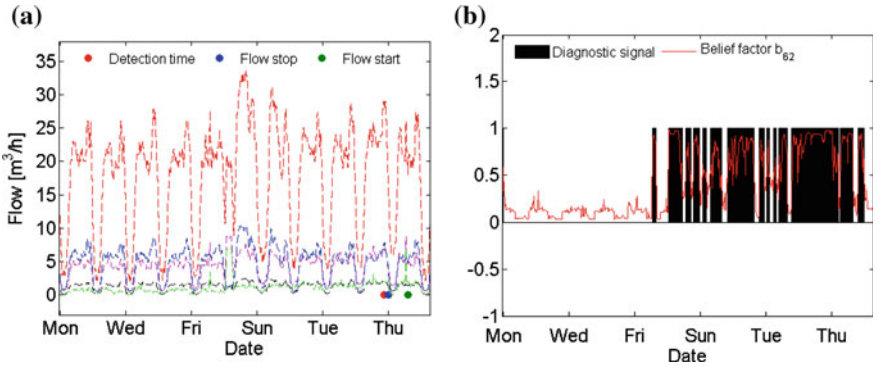


Fig. 5 Exemplary results obtained for a meta-classification neural model with a contextual switch

time of water supply cut off was on 2010-02-03 23:40:00 and the restoration time of water supply was on 2010-02-04 14:00:00 (Thu). On the other hand, the elaborated ANN model was able to early detect the first anomalies at about 29-01-2010 12:30:00 (Fri), that means above five days before it was detected by the technical supervision personnel of the water and sewage limited liability company (Fig. 5b). Moreover, there were not generated any false detection alarms.

7 Summary

In this paper the expert system shell with context-based reasoning was presented. The proposed solution is based on hierarchical reasoning realized by multiple classifiers connected together. This type of knowledge representation and reasoning is known as the instance-base and the expert system uses this form of knowledge representation and reasoning is called the instance-base expert system.

The novelty of our approach consists in application of contexts and statements. We define the context as a special feature (e.g. the day of the week) that it does not contain any relevant information about the examined problem but it can be used to control the system reasoning. We assumed that results of the system reasoning will be better when the context is used. In fundamental application, the context is applied to decide which of classifiers should be used during the system reasoning. Conducted investigations show that the application of the context is a very promising option.

Presentation of results and explanations of the system reasoning in more accurate and user-readable form is a generic problem of instance-based expert systems. Due to this problem, the application of statements was proposed as well. To use statements within the proposed expert system shell, a dictionary of statements was developed. A connection between appropriate statements from dictionary and adequate classes of a classifier was realized by means of labels.

Currently, the proposed expert system shell is implemented using the RapidMiner [2] functionality. Further investigations concerning application of the context and statements are conducted as well.

Acknowledgments The research presented in the paper was partially financed by the National Centre of Research and Development (Poland) within the frame of the project titled "Zintegrowany, szkieletowy system wspomagania decyzji dla systemów monitorowania procesów, urządzeń i zagrożeń" (in Polish) carried out in the path B of Applied Research Programme—grant No. PBS2/B9/20/2013. The part of the research was also financed from the statutory funds of the Institute of Fundamentals of Machinery Design.

References

1. Akthar, F., Hahne, C.: Combining Pattern Classifiers: Methods and Algorithms. Wiley, New York (2004)
2. Akthar, F., Hahne, C.: RapidMiner 5, Operator Reference (2012). http://www.rapid-i.com
3. Bargiel, T., Pawełek, J.: Straty wody w systemach wodociągowych - charakterystyka, wielkość, wykrywanie i ograniczenia. In: III Konferencja Naukowo-Techniczna Błękitny San (2006)
4. Brézillon, P.: From expert systems to context-based intelligent assistant aystems: a testimony. Knowl. Eng. Rev. **26**(1), 19–24 (2011)
5. Cholewa, W.: Expert systems in technical diagnostics. In: Korbicz, J., Kowalczuk, Z., Kościelny, J., Cholewa, W. (eds.) Fault Diagnosis, pp. 591–631. Springer, Berlin (2004)
6. Cholewa, W.: Multimodal Statement networks for diagnostic applications. In: Proceedings of ISMA2010—International Conference on Noise and Vibration Engineering, pp. 817–830, Leuven, Belgium, 20–22 Sept 2010
7. Cholewa, W., Rogala, T., Chrzanowski, P., Amarowicz, M.: Statement networks development environment REx. In: Jędrzejowicz, P., Nguyen, N., Hoang, K. (eds.) Computational Collective Intelligence. Technologies and Applications, Lecture Notes in Computer Science, vol. 6923, pp. 30–39. Springer, Berlin (2011)
8. Eliades, D., Polycarpou, M.: A fault diagnosis and security framework for water system. IEEE Trans. Control Syst. Technol. **18**(6), 1254–1265 (2010)
9. Geiger, G., Werner, T., Matko, D.: Leak detection and locating—a survey. In: 35th Annual PSIG Meeting (2003)
10. Gonzalez, A.J., Ahlers, R.: Context-based representation of intelligent behavior in training simulations. Trans. Soc. Comput. Simul. Int. **15**(4), 153–166 (1998)
11. Gonzalez, A.J., Dankel, D.D.: The Engineering of Knowledge-based Systems: Theory and Practice. Prentice-Hall, Englewood Cliffs (1993)
12. Hotloś, H.: Analiza strat wody w systemach wodociągowych. Ochrona Środowiska **1**, 17–24 (2003)
13. Jakubczyc, J.A.: Contextual classifier ensembles. In: Abramowicz, W. (ed.) BIS. Lecture Notes in Computer Science, vol. 4439, pp. 562–569. Springer, Berlin (2007)
14. Liebowitz, J.: The Handbook of Applied Expert Systems, 1st edn. CRC Press Inc, Boca Raton (1997)
15. Moczulski, W.: Methods of acquisition of diagnostic knowledge. In: Korbicz, J., Kowalczuk, Z., Kościelny, J., Cholewa, W. (eds.) Fault Diagnosis, pp. 675–718. Springer, Berlin (2004)
16. Moczulski, W., Ciupke, K., Przystałka, P., Tomasik, P., Wachla, D., Wiglenda, R., Wyczółkowski, R.: Metodyka budowy systemu monitorowania wycieków w sieciach wodociągowych. In: Diagnostyka Procesów i Systemów. DPS 2011. X Międzynarodowa konferencja naukowo-techniczna, Zamość, pp. 409–420 (2011)

17. Poddębniak, T.: Informacja o Wynikach Kontroli. Prowadzenie przez Gminy Zbiorowego Zaopatrzenia w Wodę i Odprowadzania Ścieków, Nr. ew. 128/2011/P/10/140/LKI. Tech. rep., Najwyższa Izba Kontroli (2011)
18. Przystałka, P., Moczulski, W.: Optimal placement of sensors and actuators for leakage detection and localization. In: Astorga, Z., Carlos, M., Molina, A. (eds.) 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Mexico City Mexico, vol. 8, pp. 666–671 (2012)
19. Puust, R., Kapelan, Z., Savic, D., Koppel, T.: A review of methods for leakage management in pipe networks. Urban Water J. **7**(1), 25–45 (2010)
20. Timofiejczuk, A.: Identification of diagnostic rules with the application of an evolutionary algorithm. Maint. Reliab. **1**, 11–15 (2008)
21. Timofiejczuk, A.: Signal feature encoding in an inference diagnostic system. Maint. Reliab. **1**, 22–27 (2009)
22. Turney, P.D.: Exploiting context when learning to classify. In: Proceedings of the European Conference on Machine Learning, pp. 402–407, ECML '93. Springer, London (1993)
23. Turney, P.D.: The management of context-sensitive features: a review of strategies. In: Proceedings of the ICML-96 Workshop on Learning in Context-Sensitive Domains, pp. 60–65 (1996)
24. Wyczółkowski, R.: Metodyka Detekcji i Lokalizacji Uszkodzeń Sieci Wodociągowych z Wykorzystaniem Modeli Przybliżonych. Wydawnictwo Politechniki Śląskiej, Gliwice (2013)
25. Yang, J., Ye, C., Zhang, X.: An expert system shell for fault diagnosis. Robotica **19**, 669–674 (2001)

Fault Detection Method Using Context-Based Approach

Mateusz Kalisch

Abstract The paper describes the context based and model-free fault detection method. The main purpose of the research is to present that there is the possibility of development of diagnostic schemes using ensemble learning and context based approach to obtain the high efficiency of the fault detection system. The achieved results confirm the effectiveness of the proposed approach and also show its limitations.

Keywords Fault detection · Context based approach

1 Introduction

Fault diagnosis is one of the most important directions of research in modern automatic control and robotics because of the increasing complexity of recent industrial objects [2, 13]. Fault diagnosis system can be developed by means of model-free (based on data), model-based and knowledge-based approaches [7]. In this paper the first approach was discussed. The base idea of data-driven technique is to use these historical data and classifiers for on-line process monitoring. It is possible to distinguish various methods of classification like e.g. probabilistic methods, methods based on the similarity between objects in the feature space or methods based on black box models [8]. All available classifiers can be used in approach based on the information fusion and meta-classification. The main idea in this approach is the application of the simple classifiers working together to solve a problem with better results than it can by achieved by the single classifier [6].

Studies and results presented in this article are connected with diagnostic expert system shell which is designed i.a. by the author of the paper in collaboration with

M. Kalisch (✉)
Institute of Fundamentals of Machinery Design, Silesian University of Technology, 18a Konarskiego Street, 44-100, Gliwice, Poland
e-mail: Mateusz.Kalisch@polsl.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_28

researchers and engineers from Institute of Innovative Technologies EMAG, Silesian University of Technology, University of Warsaw and Sevitel company.

The rest of the paper is organized as follows. In Sect. 2 the context based approach is described. The next section includes the detailed description of the proposed method. In particular, there are contained investigations on classification methods. Section 4 contains a case study description and the more interesting results of verification experiments. The last section is devoted to concluding remarks and future works.

2 Context in Machine Learning

In a classification task it is possible to distinguish three types of features: primary, contextual and irrelevant ones [16]. Primary features are useful for classification, without regard for the other features. The contextual features can not be used directly by a classifier, but can be useful when they are combined with other features. The irrelevant features are not useful at all. The primary features can be also divided into context-sensitive and context-insensitive features. Speech recognition is an example of a method which can use the contextual features (e.g. speaker's sex, nationality or age) to improve efficiency of classification [18]. In the case of machine diagnosis the context variable can be connected with e.g. weather condition. In paper [17], the author used the contextual variables such as external temperature, barometric pressure and humidity for gas turbine engine diagnosis. Another type of context is unknown context which can be identified from data by means of a method based on e.g. evolutionary algorithm [14, 15].

The contextual variable is a continuous or discrete variable connected with specific object. In case of discrete contextual variable, contextual value is equal to one of all available contextual variants describing this variable. The contextual variant can be obtained from a continuous contextual variable by the classifier. The context can be connected with a text information where first part of the message is connected with the contextual variable and object related to this variable. The second part of the message is connected with the contextual variant.

In the literature, there are described some of concepts of usage of the context with machine learning algorithms [10]. In [16, 17], Peter Turney described five strategies which show how context can be used: *Contextual normalization*, *Contextual expansion*, *Contextual classifier selection*, *Contextual classification adjustment* and *Contextual weighting*. In the paper the author used *Contextual classifier selection* approach. In this case, context variable is used as decision variable, and for each variant of the contextual variable separated classifier is prepared. The classifier related to one contextual variant uses data connected with the same variant (in both cases: learning and verification process).

3 Method of Fault Detection Using Context

In the next part of the article, two model-free fault detection approaches were described. One of the most often used model-free fault detection method is presented in Fig. 1. It can be seen, that faults are detected and distinguished using primary and redundant process variables. In this method classifier uses the subset of process variables ($U' \cup Y'$) as its input and it is dedicated for generating diagnostic signals (S).

Different classification methods can be used by the algorithms corresponding to the diagram presented in Fig. 1 [4, 7, 12], such as soft computing approaches (e.g. neural networks, Bayesian networks, fuzzy systems, etc.) or classical methods (e.g. k-nearest neighbor, decision trees, etc.).

The first concept of fault detection (Fig. 2) is based on a single classifier. Its main task is to obtain a diagnostic signal corresponding to fault or faultless states of the device. The second concept (Fig. 3) uses an ensemble classifier based on the bank of single classifiers (Fig. 4). Each classifier in the bank is connected with different variants of the context. In this case the context variable is used as a switching signal for selecting the current classifier. If the specific sample in the dataset is connected with the first contextual variant, then the algorithm chooses a classifier that has been learned on the dataset connected with the same context. For sample connected with another context, it is targeted to another classifier. Only one classifier can work at the same time.

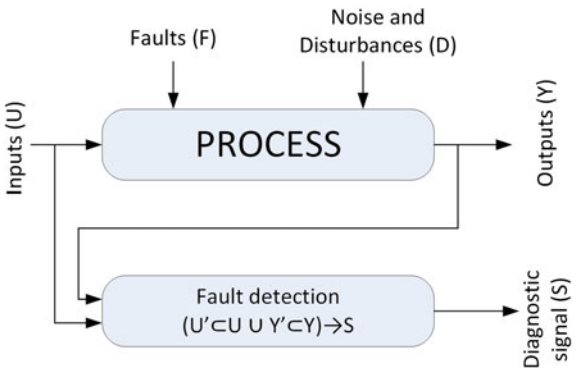


Fig. 1 A diagram of model-free fault detection [7]

Fig. 2 Scheme of fault detection based on a single classifier

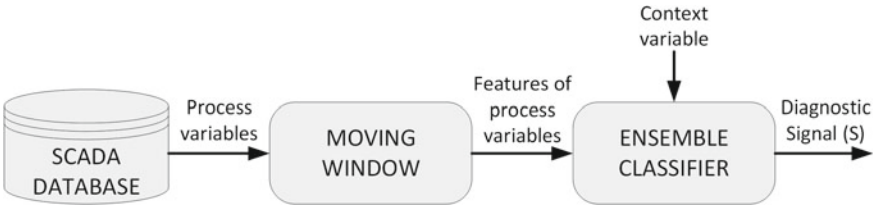


Fig. 3 Scheme of fault detection based on ensemble classifier

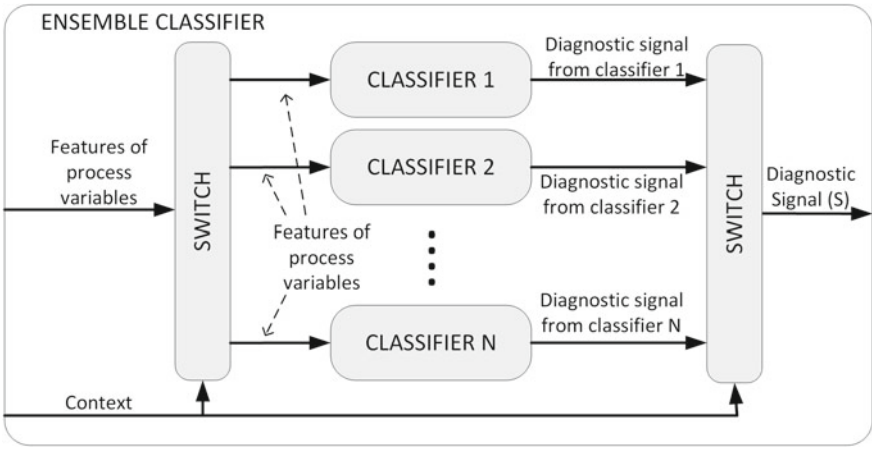


Fig. 4 Scheme of ensemble classifier

3.1 Used Classifiers

In this paper the author compared three different classifiers based on various approaches: *naive Bayes*, *decision tree* and *artificial neural network*. Each of these classifiers returns a label of a chosen class and the degrees of belief for all predicted classes. The best solution is when one of the class is characterised by the belief level equal to 1 and the rest of them are equal to 0. It gives us 100 % certainty that a new element should be classified as this particular class. In the next subsections a more precise description of the selected methods was given.

Naive Bayes This is a simple probabilistic classification method which is based on bayesian theory. However, the naive Bayes classifier considers each of existing feature independently

$$P(d_i|V_1, \cdots, V_n) = \frac{P(V_1, \cdots, V_n|d_i)P(d_i)}{P(V_1, \cdots, V_n)} \tag{1}$$

Taking into account this assumption, the Bayesian equation (1) can be transformed into (2), where the denominator of the equation is replaced by a constant C and the conditional probability is calculated by the multiplication.

$$P(d_i|V_1, \cdots, V_n) = C \cdot P(V_1|d_i) \cdot ... \cdot P(V_n|d_i) \cdot P(d_i) \tag{2}$$

The degrees of beliefs for the classification results are equal to probability values obtained from the Bayesian equation.

Decision Tree This is the classifier based on the tree-like graph created by nodes and connections between them, where the end nodes are called *leafs* and the rest of them have conditions. The result of a decision tree application depends on a chosen leaf. In the algorithm different split evaluation criteria (e.g. ratio gain in C4.5, information gain in ID3, the Gini impurity measure in CART, etc.) can be used [1, 9]. The confidence levels about the classification results are calculated separately for all leafs of the tree during the learning process. Sometimes, when learning data are very complex, the results of the decision tree may be uncertain since some of the leafs may be connected to more then one class. The class which is described by more elements then others (in specific leaf) is chosen as the main class for this leaf. The ratio between the number of elements for available classes is used to calculate the probability for each class in the leaf.

Artificial Neural Network This is a feedforward neural model in which multiple layers of neurons with nonlinear activation functions allow the network to learn nonlinear or linear relationships between input and output vectors ([3]). In this paper a multiple-layer network consists of three layers including n^1 neurons in the input layer, n^2 and n^3 neurons in the first and the second hidden layer, respectively. In this case, the neural computation can be represented by the following equation:

$$y = \mathbf{LW}^3\mathbf{f}^2\left(\mathbf{LW}^2\mathbf{f}^1\left(\mathbf{LW}^1\mathbf{u} + \mathbf{b}^1\right) + \mathbf{b}^2\right) + b^3 \tag{3}$$

where $\mathbf{LW}^{\{1,2,3\}}$ corresponds to weight matrices of the input layer and the first/second hidden layer, $\mathbf{b}^{\{1,2,3\}}$ are vectors of biases, $\mathbf{u}$ is the input signal, $\mathbf{f}^{\{1,2\}}$ are nonlinear transform operators based on tangential activation functions.

4 Verification Studies

The proposed schemes of fault detection and isolation were implemented using RapidMiner software. It is an open source software created for solving data mining problems. The verification studies were conducted on data generated using the Wind Turbine simulator [11] in order to investigate selected classification schemes.

4.1 Benchmark Problem

The author of the paper applied the benchmark model of a wind turbine elaborated by [11] and implemented in MATLAB®Simulink® software. The benchmark was developed in order to aid engineers and scientists in evolving fault detection and isolation methods, and robust controllers for the wind turbines. The benchmark can be divided into several parts connected with the wind model, blade and pitch model, drive train model, generator/converter model and controller. The benchmark allows to use several process variables such as measured wind speed at hub position (V_{hub} [m/s]), rotor speed (ω_r [rad/s]), generator speed (ω_g [rad/s]), feedback pitch angle (β_f), pitch angle (β [deg]), generator torque (τ_g [Nm]) and electrical power (P_g [W]). In this benchmark model several faults are considered. These faults cover sensor, actuator, and process faults in different parts of the wind turbine. The list of available faults implemented in the benchmark was grouped and presented in Table 1.

In this paper faults F1, F2, F3, F4, F5 and F8 are only investigated in order to show clearly advantages and limitations of the proposed approach.

Table 1 The set of considered faults

Sensor faults	
F1	Fixed value on Pitch 1 position sensor 1
F2	Scaling error on Pitch 2 position sensor 2
F3	Fixed value on Pitch 3 position sensor 1
F4	Fixed value on Rotor speed sensor 1
F5	Scaling error on Rotor speed sensor 2 and Generator speed sensor 2
Actuator faults	
F6	Changed pitch system response pitch actuator 2—high air content in oil
F7	Changed pitch system response pitch actuator—low pressure
System faults	
F8	Offset in Converter torque control
F9	Change Dynamic Drive train

4.2 Data Preprocessing and Feature Selection

The author prepared different datasets and each dataset contained the same fragment of power generation process but faults occurred in different periods of time in order to reduce the overfitting problem. The number of samples in each dataset was equal to 120,000. Half of them were connected with the faulty state and the second half were related to the faultless state. Only six faults were used during tests, so the number of samples connected with each fault was equal to 10,000. The available list of the process variables is widened by two another attributes. The first one contained information about the current context of a device, the second contained information about a state of each sample in the dataset (fault-free state or faulty state) which was used during supervised learning process of the classifiers. In the benchmark model several signals are measured by redundant sensors. For this kind of signals two physically redundant measurements were subtracted from each other to generate one residual signal. All the obtained residues were added to the matrix of process variables. In the next step the algorithm calculates scalar features of all available signals (raw signals of sensors and residues). The authors choose a few time domain features often used in fault diagnosis [5], such as: average, maximum and minimum values, standard deviation, root mean square, shape factor, kurtosis, time-domain energy, skewness and entropy. To reduce the number of features during classification task (learning and verification) some methods were used in order to calculate relevance of each attribute and choose a few the most relevant. During this process six algorithms of attribute selection available in RapidMiner software were used: Information Gain, Information Gain Ratio, Correlation, Chi squared Statistic, Gini Index and Uncertainty. All these methods of feature selection require a training dataset with labels prepared for supervised learning. As a result of this analysis the author chose 27 attributes of 210 available: The mean signal of residuum of ω_r and measured value of P_g, the maximum value of measured values of β_3 (sensor 1), ω_g (sensors 1 and 2), ω_r (sensor 1), τ_g and residues of β_3 and ω_g, the standard deviation of measured value of P_g, the shape factor of measured values of β_1 (sensor 1), β_3 (sensor 1) and ω_r (sensor 1), the kurtosis of measured values P_g and τ_g, the time domain energy of measured value of ω_r, the skewness of measured values of β_3 (sensor 1), ω_g (sensors 1 and 2), ω_r (sensors 1 and 2), P_g, τ_g and residues of β_3 and ω_g, the last signal is the entropy of residuum of ω_r.

4.3 Classification Schemes Implementation

The RapidMiner software allows to create data mining processes with the use of a visual programming language. This tool gives the opportunity for developing different classification schemes using the so-called drag and drop methodology. In this way the classification processes can be viewed as dataflow graphs (Fig. 5).

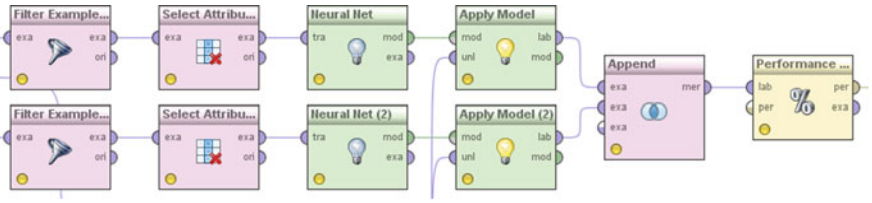


Fig. 5 Example of RapidMiner process

Figure 5 presents a fragment of the scheme of learning and verification process using context based approach. *Filter examples* operators divide initial dataset into two datasets separated by the context. A top line of the process works with the first context of the wind turbine process and the bottom one works with the second context. In the second step of the process, the operators *Select Attributes* choose only the most relevant features described above in the article. Each classifier (*Neural Net*) is trained with dataset connected with the specific context. The trained classifiers are used in *Apply model* operator but with using the testing dataset. The testing data are also divided into two separated groups depending on the context. All results of each classifier are merged into one dataset (*Append*) and then the accuracy is calculated in the *Performance* operator.

4.4 Context Based Approach

The real wind turbine can work in four different states (zones) which depends on the wind speed (Fig. 6). The first step occurs when wind is too weak to power the wind turbine generator and it doesn't produce energy. During the second step, the power generated by the wind turbine depends on the speed of the wind. In the third step the wind is too strong for the wind turbine and the controller has to change the angles of the blades to keep the highest possible level of the power generated by the turbine and reduce speed of the generator. In the last step the controller is not able to keep the maximum level of the generated power because the strength of the wind is too high. The controller has to adjust a position of the blades towards the wind in such a way, that the turbine and the generator will stop. The wind turbine cannot work properly when the wind speed is too high and it can cause very serious damages of the wind turbine.

A model of the controller implemented in the simulator uses a measured value of the power generated by the turbine to decide in which zone the turbine is currently working. The controller in the simulator considered only two available states (zone 2 and zone 3). Information about current state of the controller during simulation is added to the dataset of measured values. The fault detection process uses this variable as a contextual variable to control an ensemble classifier.

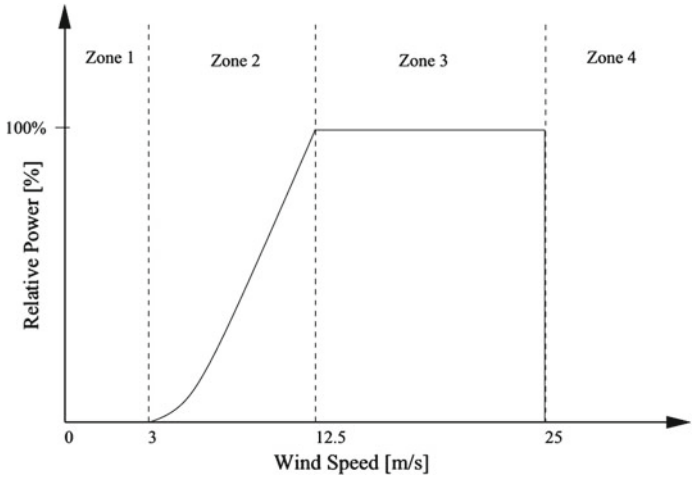


Fig. 6 The relation between wind speed and power generated by the wind turbine [11]

4.5 Results of Verification Studies

Verification studies show the comparison between each possible combination of training and testing datasets. Each classifier was learned as a single classifier and ensemble classifier then created classifiers were tested by another dataset where faults occurred in other periods of time. The author prepared six different datasets so the number of combinations and tests is equal to 30. The final results of verification tests are presented in Table 2 as the average value and standard deviation of the accuracy. The accuracy value describes a ratio between the number of correctly classified samples and all available samples. This measure is reliable because of well balanced training and testing datasets.

Results show that most of the ensemble classifiers based on two separated classifiers reach better accuracy of fault detection than a single classifier. Naive Bayes classifier achieved the biggest increase but the general result was the lowest of all the tested classifiers. The author considered two different options for classifiers based on artificial neural network. In the first case the neural network had only 1 hidden layer,

Table 2 Results of verification tests

	Accuracy [%]			
	Naive Bayes	Decision tree	Neural net (one layer)	Nerual net (two layers)
Single classifier	$67,3(\pm 6,6)$	$84,5(\pm 2,3)$	$77,0(\pm 5,7)$	$75,9(\pm 5,7)$
Ensamble classifier	$71,6(\pm 4,4)$	$83,4(\pm 3,1)$	$79,0(\pm 4,2)$	$77,8(\pm 3,7)$

and in the second case the neural network had two hidden layers. As it is shown in Table 2, the general result of a single classifier is a little bit lower for both artificial neural networks. The standard deviation shows that all base results of ensemble classifiers were closer to the average value than results of a single classifier. Decision tree was the only classifier which reached better result as a single classifier than ensemble classifier. The decision tree is the classifier based on rules. This kind of the classifier is able to fit to different states of the device, especially to faults like F1, which is connected with fixed measured value of a sensor. All the decision trees (the single classifiers and the ensemble classifiers) use the same parameters which should prevent an overfitting problem. In case of ensemble classifier, each base classifier is learned by smaller dataset connected with specific period of time where the samples are similar to each other. It is possible that the decision tree working with this kind of the data tends to be more overfitted and probably reaches lower result for the verification dataset than the single classifier learned by the full dataset.

Table 3 shows results of the classifier based on the neural network with two hidden layers. Column *All* presents general efficiency of classification (the ratio of correctly classified samples to incorrectly classified samples). The value in the column *F0* shows how many samples from fault-less state were correctly classified to this state. It is possible to count the ratio of false alarms, for single classifier it is equal to $34,33$ and $17,80$ for the ensemble classifier. The single value connected with all the fault states presents general efficiency of fault detection. The values presented below show efficiency of detection of each fault considered during the experiment. The single classifier is able to reach higher accuracy for faults which are easy to recognize, such as F1 an F3. For more difficult faults the accuracy values are much lower. The ensemble classifier obtains better results for the more difficult faults but at the same time its accuracy connected with faults F1 and F4 is a little bit lower than results reached by the single classifier.

Figure 7 shows fragments of prediction vectors obtained from the neural net classifiers working as single classifier and ensemble classifier. The first plot shows when fault occurred in the presented fragment of a process, where value 1 means faulty state and value 0 fault-free state. The second graph presents the changes of context in the same part of dataset. The third and fourth plots show diagnostic signals obtained using the single classifier and the ensemble classifier. In the presented case differ-

Table 3 Example of detailed results of classifier based on neural network

	All	F0	F1	F2	F3	F4	F5	F8
Single classifier	70,17	65,77	74,58					
			98,99	28,79	99,07	99,79	99,98	20,84
Ensemble classifier	81,25	82,20	80,30					
			84,9	60,52	99,57	93,73	99,66	43,41

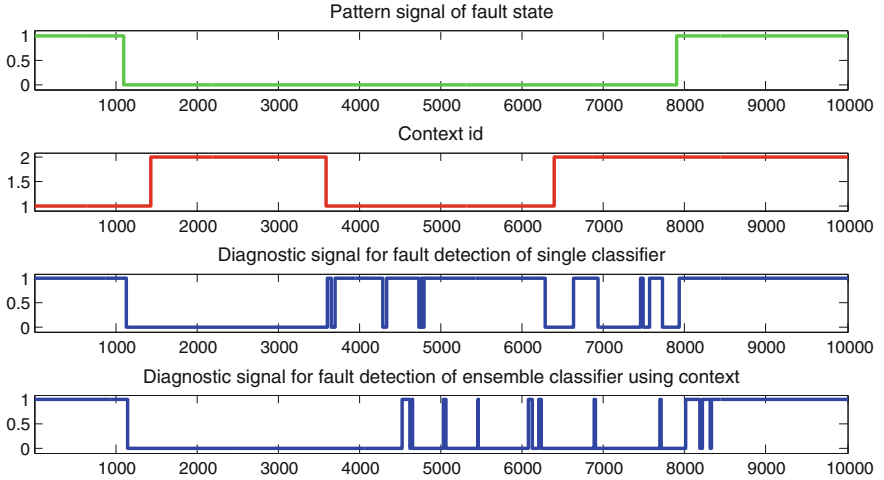


Fig. 7 Selected results obtained for neural net classifier working as single classifier and ensemble classifier

ences between single classifier and ensemble classifier are clear. For the single classifier it is more difficult to recognize fault-free state, especially when the first context occurred. Prediction of the ensemble classifier is more fitted to real situations. There are only a few short false alarms when fault does not occur. The ensemble classifier generally reached better results than the single classifier, however, in few places the single classifier clearly led to superior results.

5 Conclusions

The results presented in the paper show that proper usage of the contextual variable can increase the accuracy of classification in case of fault detection. Another advantage of this method is that two or more separated classifiers can be learned by smaller datasets than a single classifier because each classifier is connected with other part of learning data. More classifiers using different datasets can be learned in parallel, so the time remaining for learning process can be significantly reduced. This method has also one disadvantage. Even when learning data are very well balanced (the number of samples connected with faulty state is equal or similar to the number of samples of fault-free state), after splitting datasets by a contextual variable, each sub-dataset can be poorly balanced. In this paper samples with a faulty state contained samples connected with six various faults. Each fault generates different symptoms in the process variables. It is possible that during learning process one specific fault occurs in a section of the data connected only with the first contextual variant but during verification the same fault occurs in a section connected

with the second contextual variant. It means that the classifier connected with the second contextual variant is not prepared for this type of data and the final result of fault detection can be incorrect.

5.1 Future Work

In the next step the author is going to propose another examples of the ensemble classifiers based on the context which will reduce known problems and issues especially related to problem of unbalanced data after contextual splitting. Another goal of the future works is to test the current and all next concepts of fault detection and isolation based on the context approach with different data such as real data collected in mining industry.

Acknowledgments The research presented in the paper was partially financed by the National Center of Research and Development (Poland) within the frame of the project untitled. "An integrated shell decision support system for systems of monitoring processes, equipment and hazards" carried out in the path B of Applied Research Programme—grant No. PBS2/B9/20/2013. The part of the research was also financed from the statutory funds of the Institute of Fundamentals of Machinery Design.

References

1. Akthar, F., Hahne, C.: Rapidminer 5, Operator Reference (2012). http://www.rapid-i.com
2. Caccavale, F., Villani, L.: Fault Diagnosis and Fault Tolerance for Mechatronic Systems: Recent Advances. Springer Tracts in Advanced Robotics. Springer, Berlin (2003)
3. Haykin, S.: Neural Networks: A Comprehensive Foundation. 2nd edn. Prentice Hall International, Englewood Cliffs (1999)
4. Isermann, R.: Fault-diagnosis Systems: An Introduction From Fault Detection to Fault Tolerance. Springer, Berlin (2006)
5. Jigar, P., Vaishali, P., Amit, P.: Fault diagnostics of rolling bearing based on improve time and frequency domain features using artificial neural networks. Int. J. Sci. Res. Dev. **1**, 781–788 (2013)
6. Kalisch, M., Przystałka, P., Timofiejczuk, A.: Application of selected classification schemes for fault diagnosis of actuator systems. In: Ganzha, M., Maciaszek, L., Paprzycki, M. (eds.) Proceedings of the 2014 Federated Conference on Computer Science and Information Systems. Annals of Computer Science and Information Systems, vol. 2, pp. 1381–1390. IEEE (2014)
7. Korbicz, J., Kościelny, J.M., Kowalczuk, Z., Cholewa, W. (eds.): Fault Diagnosis. Models, Artificial Intelligence, Applications. Springer, Berlin (2004)
8. Kuncheva, L.: Combining Pattern Classifier: Methods and Algorithms. Wiley-Interscience, New Jersey (2004)
9. Lile, A.: Analyzing e-learning systems using educational data mining techniques. Mediterr. J. Soc. Sci. **2**(3), 403–419 (2011)
10. Munoz, D., Bagnell, J.A., Vandapel, N., Hebert, M.: Contextual classification with functional max-margin markov networks. In: CVPR, pp. 975–982. IEEE (2009)
11. Odgaard, P., Stoustrup, J., Kinnaert, M.: Fault tolerant control of wind turbines—a benchmark model. In: 7th Ifac Symposium on Fault Detection, Supervision and Safety of Technical Processes, pp. 155–160 (2011)

12. Patton, R., Uppal, F., Lopez-Toribio, C.: Soft computing approaches to fault diagnosis for dynamic aystems: a survey. In: IFAC Symposium Safeprocess, pp. 298–311 (June 2000)
13. Patton, R.J., Frank, P.M., Clark, R.N.: Issues of Fault Diagnosis for Dynamic Systems. Springer, Berlin (2000)
14. Timofiejczuk, A.: Identification of diagnostic rules with the application of an evolutionary algorithm. Maint. Reliab. **1**, 11–15 (2008)
15. Timofiejczuk, A.: Context based diagnostics of rotating machinery. In: 7th International Conference on Acoustical and Vibratory Surveillance Methods and Diagnostic Techniques, pp. 1–9, Chartres, France (2013)
16. Turney, P.: The Management of context-sensitive features: a review of strategies. In: Proceedings of the Icml-96 Workshop on Learning in Context-sensitive Domains, pp. 60–65 (1996)
17. Turney, P.D.: Exploiting context when learning to classify. In: Proceedings of the European Conference on Machine Learning, ECML '93, pp. 402–407. Springer, London (1993)
18. Widmer, G.: Tracking context changes through meta-learning. In: Machine Learning, pp. 259–286. Kluwer Academic Publisher, Boston (1996)

Automatic Graph-Based Local Edge Detection

Jagoda Lazarek and Piotr S. Szczepaniak

Abstract In this paper, we present a method of edge detection in the absence of prior knowledge of the analyzed image. The method is based on a graph representation of the image, more precisely, on graph representations of the image parts. The process of edge detection is based on a flow network and graph-cut techniques. "Source" and "sink" are indicated automatically. The results of edge detection are based on an artificial example and real images.

Keywords Image processing · Edge detection · Segmentation · Graphs · Flow networks · Graph-cuts

1 Introduction

Edge detection is usually one of the first steps in image processing methods and its result constitutes an input for object detection and recognition methods. This makes it a key operation that affects the quality of further high-level processing procedures [1, 6, 12–16]. There are many known methods that deal with the described task, for example Canny, Sobel, Roberts, Prewitt or Laplacian edge detector, as well as LoG—Laplacian of Gaussian and DoG—Difference of Gaussian [11]. Due to its importance, this stage of processing has become the focus of much research, which aims not only to develop better methods of edge detection as such, but also to provide a basis for the research concerned with further stages of processing.

In addition to a bitmap representation of the image, processing methods can operate on its graph-based representation. Graph-based representations are useful because they allow one to describe the relationship between neighboring pixels. This attribute is helpful in detecting edges and segmentation.

J. Lazarek (✉) · P.S. Szczepaniak
Institute of Information Technology, Lodz University of Technology,
Wólczańska 215, 90-924 Łódź, Poland
e-mail: Jagoda.Lazarek@p.lodz.pl

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_29

There are many methods of image segmentation which are based on the graph representation of the image. The idea is based on the theory of network flow, especially graph cut techniques. Among them, we can identify a common feature—interactivity. The user indicates the pixels as "pixel of object" and "pixel of background". These methods are knowledge-driven. The user can provide hard constraints for segmentation. There are no restrictions on how the user should indicate the pixels belonging to the object and the background. The user can select individual pixels or separated parts of the image. The remaining part of the analysis, i.e. optimal boundary finding, is based on the graph cut method. Thus, the user can obtain the expected result of segmentation. The usage of graph-cuts is presented for example in [2–4]. Other techniques combining graph-cuts with shape priors are described in [9].

However, sometimes the user has no prior knowledge and is not able to provide hard constraints such as those described above. In this case, we can only apply soft constraints, for example the relations between neighboring pixels, global or local thresholds etc.

Our solution is designed to solve the latter case, i.e. if no prior knowledge about the processed image is available, or more specifically, if the goal of edge detection or segmentation is unknown . Our method is based on a local approach to graph-cut technique and the automatic selection of "source" and "sink" pixels. A detailed description and the results of experiments on an artificial example and real images are presented below. We also refer to our previous work, which concerns the interactive segmentation method based on graph cuts.

The paper is organized as follows. We start with a description of the considered problem in Sect. 1. A brief description of flow networks and graph cuts is provided in Sect. 2. Then, in Sect. 3, we refer to our previous work, which involved the use of graphs in segmentation. The remainder of the paper is devoted to the description and testing of the proposed method. In Sect. 4, we describe the algorithm and the parameters of the method. Section 5 presents the tests conducted using simple patterns and an artificial example. In Sect. 6, the results of edge detection on real images are shown and the influence of parameters is described. The paper closes with a summary in Sect. 7.

2 Basic Theory of Graphs, Flow Networks and Graph-Cuts

In this section, we recall the basic definitions related to the graphs and flow networks, as well as graph-cut techniques that we use to detect the edges.

Definition 1 A graph is an ordered pair $G = (V, E)$ comprising a set of vertices V together with a set of edges E, where each edge is a pair of elements from the set V.

Definition 2 A flow network $G = (V, E)$ is a directed graph in which each edge $(u, v) \in E$ has a nonnegative capacity c ($c(u, v) \neq 0$). If $(u, v) \notin E$ then $c(u, v) = 0$. We distinguish two vertices in the flow network: a source s and a sink t. A flow in

network G is real-valued function $f : V \times V$ that satisfies the following properties [5]:

- **Capacity constraint**: For all $u, v \in V$ we require $f(u, v) \leq c(u, v)$
- **Flow conservation**: For all $u \in V - \{s, t\}$ we require $\sum_{u,v \in V} f(u, v) = \sum_{u,v \in V} f(v, u)$.

The capacity constraint ensures that the flow from u to v is not greater than the capacity of the edge (u, v). Flow conservation means that the value of a flow entering the node equals the sum of the flows leaving the node.

2.1 Maximum Flow Problem

In a given network G, with source s and sink t, the maximum flow problem can be formulated as finding the maximum amount of data (e.g. water) that can be sent from s to t, through the network G (via graph edges with capacities equal to edge weights). To describe the basic algorithm (Ford-Fulkerson method) that is used to solve the problem, we need to introduce the definitions of residual network, augmenting path and minimum cut of a graph.

Residual Network Given a flow network $G = (V, E)$ and a flow f, the residual network of G induced by f is $G_f = (V, E_f)$ [5], where

$$E_f = \{(u, v) \in V \times V : c_f(u, v) > 0\} \tag{1}$$

where $c_f(u, v)$ we call residual capacity, which is defined by the following equation

$$c_f(u, v) = c(u, v) - f(u, v) \tag{2}$$

The edges in E_f are either edges in E or their reversals, and thus the following constraint for the number of edges: $|E_f| \leq 2|E|$.

Augmenting Paths Given a flow network $G = (V, E)$ and a flow f, an augmenting path p is a simple path from s to t in the residual network G_f. By the definition of the residual network, we may increase the flow on an edge (u, v) of an augmenting path by up to $c_f(u, v)$ without violating the capacity constraint on whichever of (u, v) and (v, u) is in the flow network G [5]. We call the maximum amount by which we can increase the flow on each edge in an augmenting path p the residual capacity of p, given by

$$c_f(p) = min\{c_f(u, v) : (u, v) \; is \; on \; p\} \tag{3}$$

Cut of Flow Network A cut (S, T) of a flow network $G = (V, E)$ is a partition of V into S and $T = V - S$ such that $s \in S$ and $t \in T$. If f is a flow, then the net flow across the cut (S, T) is defined to be $f(S, T)$. A capacity of cut (S, T) is $c(S, T)$. We distinguish a minimum cut of a network, which is a cut whose capacity is minimum over all cuts of the network [5].

Ford-Fulkerson Method In each iteration of the Ford-Fulkerson method [8], we find an augmenting path p and use p to increase the flow f along it by residual capacity $c_f(p)$. When no augmenting paths exist, the flow f is a maximum flow.

Edmunds-Karp Algorithm Ford-Fulkerson [7] method with breadth-first search strategy for finding an augmenting path is called Edmunds-Karp algorithm. Computational complexity of this algorithm is $O(VE^2)$.

2.2 Graph Construction

To construct the graph, we need to connect neighboring pixels. We can choose between a 4- and an 8- neighborhood system. In the network, we distinguish two vertices: a source s and a sink t (which are called terminals), as well as a sourceSeed (object pixel) and a sinkSeed (backrgound pixel). SourceSeed and sinkSeed may be, single pixels or areas (connected or disjoint). All vertices of a graph are connected to both s and t. Those connections are called links or edges. Capacities (weights) of those edges are $c(s, sourceSeed) = \infty$, $c(sourceSeed, t) = 0$ and $c(s, sinkSeed) = 0$, $c(sinkSeed, t) = \infty$. For the other pixels u (neither sourceSeed nor sinkSeed), weights of edges $c(s, u)$ and $c(u, t)$ are calculated. We also need to determine the weights for the edges between adjacent pixels $c(u, v)$. Edge weights are attributes which indicate the strength of connections between nodes in the network (here: pixels) and are used for calculating the cost of network cuts. The schemas of the network and the cut result are presented in Fig. 1.

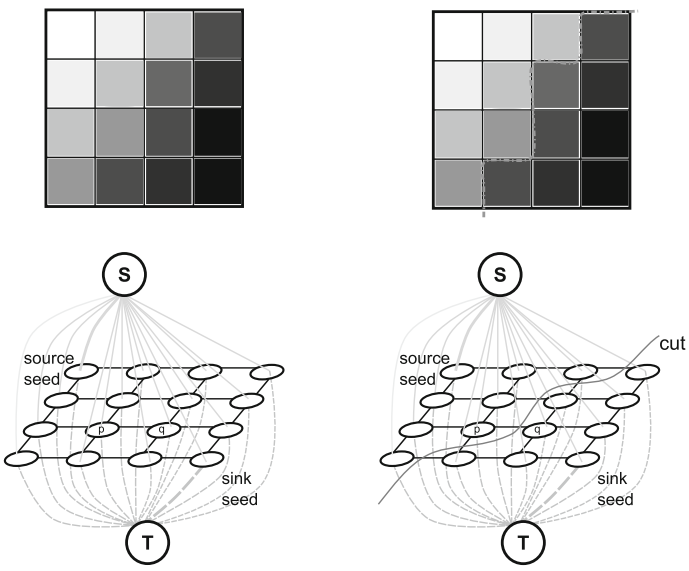


Fig. 1 Flow network and cut result

3 Segmentation Based on Flow Network

As mentioned in the Introduction, interactive segmentation based on a graph-cut technique can provide satisfactory results. Such an approach is possible when the user has prior knowledge about the analyzed image and is able to identify the target of segmentation, i.e. to provide hard constraints. In our prior work [10], the max-flow method was applied to the segmentation of medical images. An example of the analyzed image with the results of segmentation is presented in Fig. 2. In Fig. 2a, pixels belonging to the source are marked on the right ventricle and pixels belonging to the sink are marked on the left ventricle and on the septum. In Fig. 2c, pixels belonging to the source are marked on the left ventricle and pixels belonging to the sink are marked on the right ventricle and on the septum.

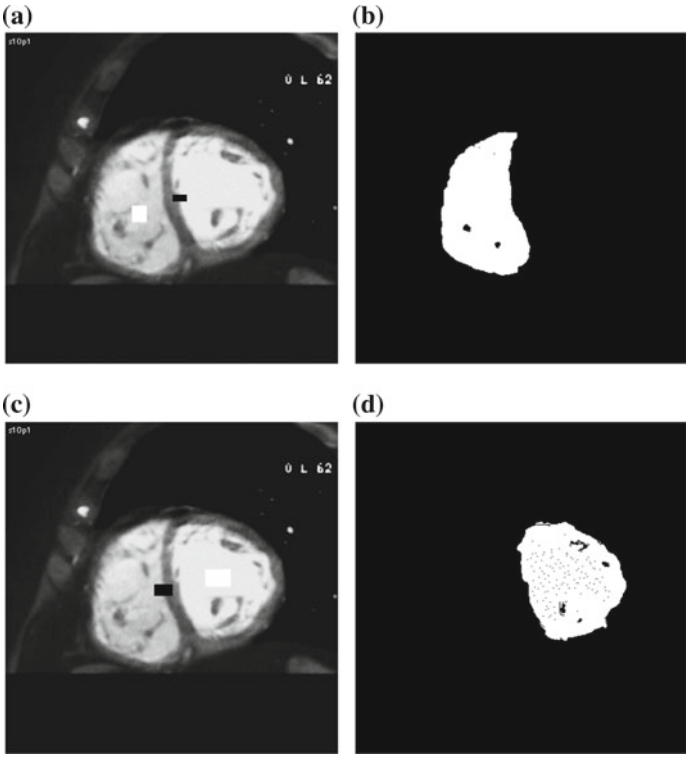


Fig. 2 Segmentation of the ventricles of the human heart: **a** Source (*white rectangle*), sink (*black rectangle*) marks; **b** Result—*right* ventricle is segmented; **c** Source (*white rectangle*), sink (*black rectangle*) marks; **d** Result—*left* ventricle is segmented

Proper results were obtained, as we had prior knowledge of the processed image, and we were able to identify the pixels belonging to the source and sink. In the following part of this paper, the method that solves the problem of finding the edge in the absence of knowledge of the analyzed image is presented.

4 The Proposed Edge Detection Method Based on a Local Approach to Graph Representation of the Image

The proposed method is intended for use in the absence of prior knowledge of the processed image. This means that we are not able to identify the object whose edges are to be detected. In short, our goal is to find the edges and minimize the number of artifacts. To this end, we decided to process the image locally. First of all, we must begin by dividing the image into frames of size $n \times n$, and then process them separately. Splitting the image into tiles of dimensions n x n is not always possible, a scheme for processing the boundary pixels of the input image was not designed. On this stage of the method development, the boundary pixels are ignored in the output.

Each frame is treated as an independent graph, for which the "source" and "sink" seeds are selected automatically. Then, the minimum s-t cut method is applied. It should be mentioned that we generate and analyze graphs only for the frames for which the absolute difference between the maximum and minimum pixel values is greater than the established threshold. The steps of the algorithm are shown in Fig. 3.

4.1 Graph Construction

In the construction phase of the graph, we can choose between a 4- and an 8- neighborhood system. To build a proper graph, we need to assign weights between adjacent pixels (two of them are the "source" and "sink" seeds), as well as between terminals S and T.

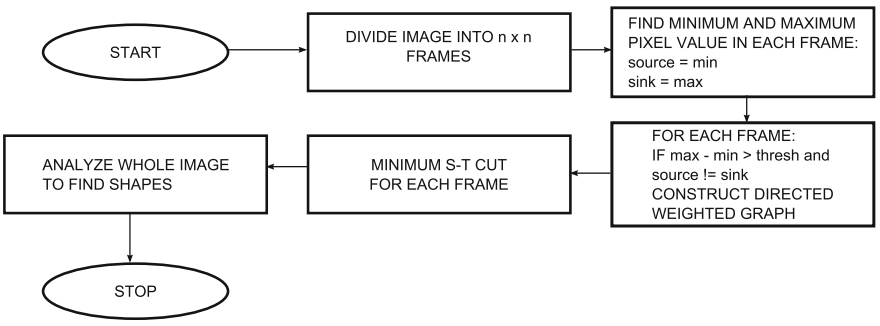


Fig. 3 The algorithm of edge detection

Table 1 Types of edges and their weights

Edge type	Weight
$\{S, sourceSeed\}$	maxValue
$\{sourceSeed, T\}$	0
$\{sinkSeed, T\}$	maxValue
$\{S, sinkSeed\}$	0
$\{u, v\}$	$255 - \lvert img(u) - img(v)\rvert$
$\{S, u\}$	$\lambda * (255 - \lvert min - img(u)\rvert)$
$\{u, T\}$	$\lambda * (255 - \lvert max - img(u)\rvert)$

where
- S, T—source and sink terminals
- sourceSeed, sinkSeed—source and sink seeds
- u, v—other pixels (neighter sourceSeed nor sinkSeed)
- img(u), img(v)—intensity of pixels u or v
- maxValue—value which indicates the strongest connection between nodes, to be chosen by user

The weights depending on the type of the edge are presented in Table 1. "Source" and "sink" seeds are chosen automatically. The darkest pixel in a frame is mapped as the "source" seed and the brightest pixel is mapped as the "sink" seed. The maxValue in our implementation is the maximum value of integer.

5 Tests

In order to validate the proposed method, the basic research was performed—on very simple patterns, as illustrated in Fig. 4, and the designed artificial example, shown in Fig. 5.

5.1 Experiments on Basic Patterns

For each pattern we tested the proposed method as well as Canny and Sobel edge detector. We used 7×7 filter size for both additional detectors and high threshold value on level of 50 for the Canny detector.

It is easy to notice that the proposed method yields visually better results of edge detection. We are able to find a single edge, while Canny and Sobel detectors generate a lot of lines, among which it is difficult to distinguish the right edge.

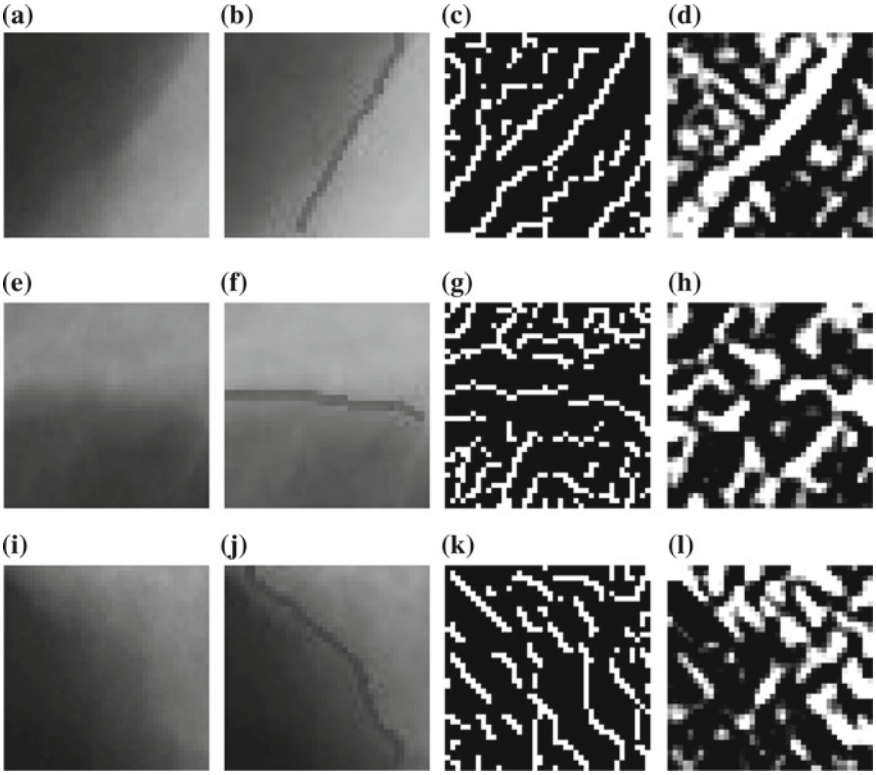


Fig. 4 Patterns and detected edges based on different methods: **a** No. 1, **e** No. 2, **i** No. 3, **b, f, j** Our method, **c, g, k** Canny, **d, h, l** Sobel

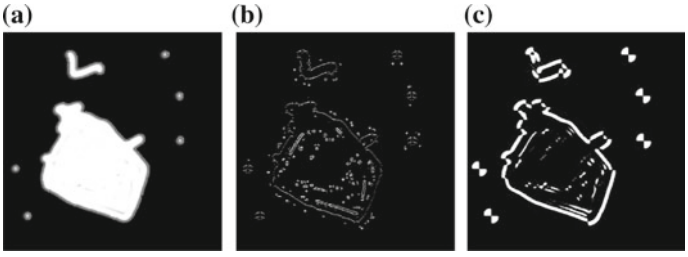


Fig. 5 The artificial example—the results of edge detection, **a** Our method, **b** Canny method, **c** Sobel method

5.2 Artificial Example

To check the proposed method on more complex image we designed an artificial example (image size: 396×331 px) and applied the method. We used the following values of parameters: frame size –20, threshold –50, $\lambda - 5$.

We also compared the results obtained by means of our method with those provided by Canny and Sobel methods. We used 7×7 filter size for both additional detectors and high threshold value on level of 50 for the Canny detector, Fig. 5. We used a 4-neighborhood system.

Properties of the Proposed Method The proposed method is able to detect a white line on a black background as well as detect a black line on a white background. It can also deal with noisy images (Fig. 6).

The obtained results were satisfactory. All edges were found properly. Therefore, further testing was performed on more difficult, real-world images, which are presented in Sect. 6. The dependency between the time of processing and the size of a frame in our method was also checked. The result, which is presented in Fig. 7, follows directly from the computational complexity of Edmunds-Karp algorithm (2.1).

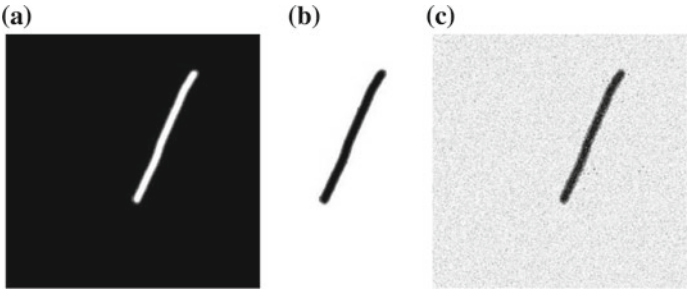


Fig. 6 Detected edges of: **a** the white line on the black background—frame size: 20, threshold: 10, λ: 20, **b** the *black line* on the white background—frame size: 20, threshold: 10, λ: 20, **c** the *white line* on the black background (noisy image)—frame size: 20, threshold: 200, λ: 10

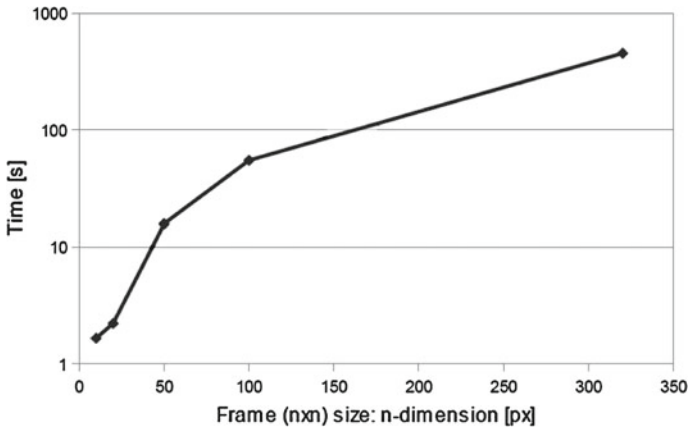


Fig. 7 The graph of time-size dependency for artificial example processing

Algorithm Implementation and Test Environment Algorithm was implemented sequentially in Java, according to the schema presented in Fig. 3. Tests have been done with the Asus laptop with Intel Core i7-3610QM CPU 2.30 GHz, 4 GB RAM, 64-bit version of Windows. It is possible to use a parallel implementation due to the disjoint processing of each of the graphs.

6 Experimental Results

Finally, we checked the proposed method on real images. We tested the influence of frame size, threshold and λ value on results of edge detection, Figs. 9, 10. We used a 4-neighborhood system. The results were also compared with Canny and Sobel methods. We used 7×7 filter size for both additional detectors and high threshold value on level of 150 for the Canny detector, Fig. 8.

The λ has an impact on the relative importance of "region" in relation to "border". Too low λ value causes gaps in boundaries, as shown in Figs. 9a, b and 10a, while too high a value of λ results in a big number of additional artifacts—Fig. 9d. Very small parts of the image, such as single pixels have a tendency to be separated. Frame size has an influence on the accuracy of edge detection—Figs. 9c, 10b. The frame size shown in Fig. 10b allows one to obtain more adequate results and reduces the number of artifacts, because of a wider context of processing. However, too big a frame size, which is almost equal to the image size, as illustrated in Fig. 10d, results in a very long processing time (compare Fig. 7). The results are not better in comparison to Fig. 10b and it can be observed that a lot of details have been missed. The threshold level gives the possibility to reduce the number of artifacts and shorten

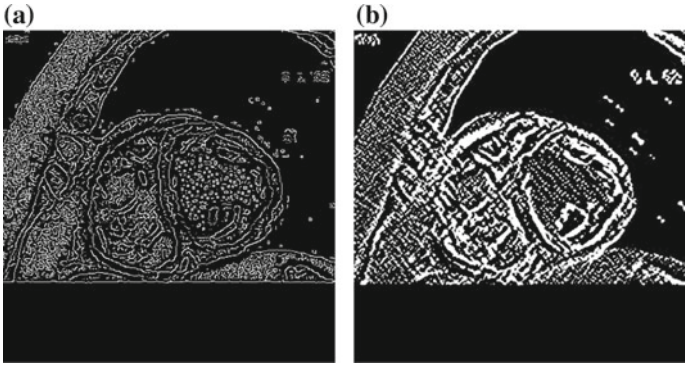


Fig. 8 The results of edge detection of human heart image (400×400 px), **a** Canny method, **b** Sobel method

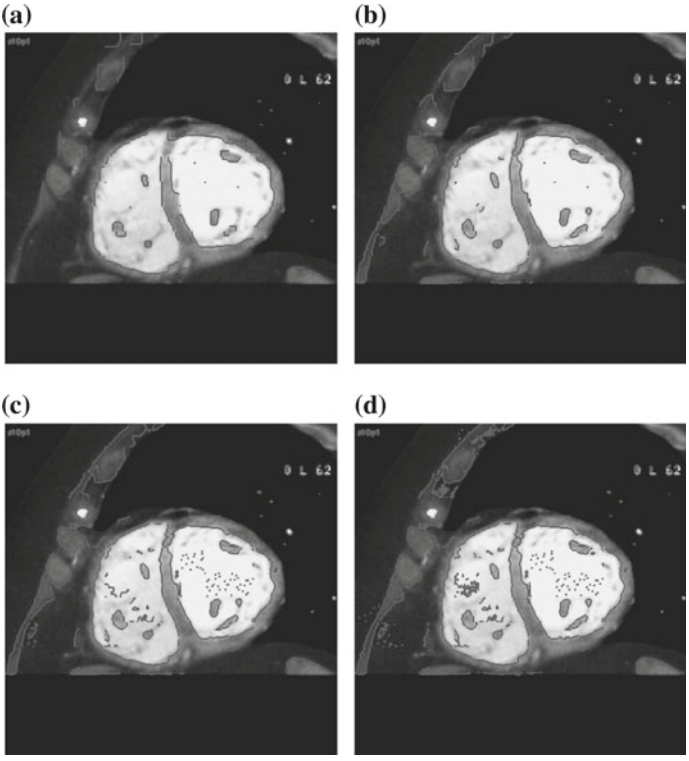


Fig. 9 The results of edge detection based on our method with different parameters (image size—400×400 px). **a** Frame size: 30, threshold: 25, λ: 2. **b** Frame size: 30, threshold: 25, λ: 10. **c** Frame size: 30, threshold: 25, λ: 100. **d** Frame size: 30, threshold: 25, λ: 1000

the processing time, because the frames which do not meet this condition are not processed (compare Fig. 10c with Fig. 9c). It is also crucial to compare the results given by Canny and Sobel detectors to those produced by our method. Canny and Sobel detectors generate much more artifacts than the proposed method, see Fig. 8. Thus, we can conclude that our method provides a more convenient way to use the results of processing in further image analysis stages, e.g. we can find consistent shapes (Fig. 5a) and then recognize them.

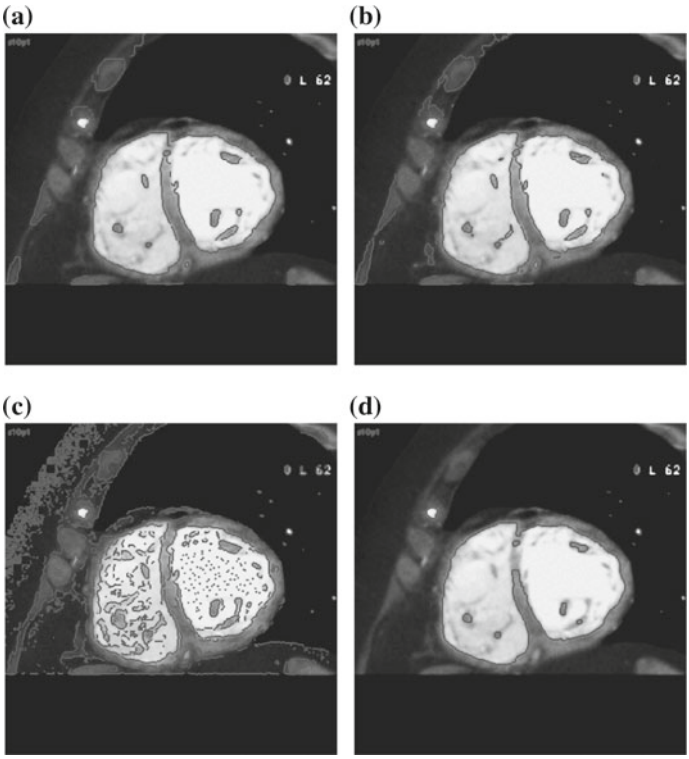


Fig. 10 The results of edge detection based on our method with different parameters (image size—400 × 400 px). **a** Frame size: 50, threshold: 25, λ: 10. **b** Frame size: 50, threshold: 25, λ: 100. **c** Frame size: 10, threshold: 10, λ: 100. **d** Frame size: 350, threshold: 25, λ: 10

7 Conclusions

To conclude, we proposed the method which is based on a flow network and the graph cut theory. Thanks to the local approach and automatic way of "source" and "sink" seeds indication, the method gives the possibility to find edges even if no prior knowledge about the image is available. The obtained results were satisfactory and fulfilled previous assumptions, in terms of the quality of processing (in comparison to Canny and Sobel detectors) and processing time. We showed that results given by the proposed method may generate fewer artifacts. Our tests were performed using simple patterns, an artificial example, as well as real images. All of them provided promising results.

We also proved that it is preferable to split the image into pieces and process them separately rather than the whole picture, and that it does not affect the quality of the edge detection result.

References

1. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
2. Boykov, Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In: ICCV, pp. 105–112 (2001)
3. Boykov, Y., Kolmogorov, V.: Computing geodesics and minimal surfaces via graph cuts. In: ICCV, pp. 26–33 (October 2003)
4. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. **26**(9), 1124–1137 (2004)
5. Cormen, T., Leiserson, C., Rivest, R., Stein, C.: Introduction to Algorithms. Third Edition, Massachusetts Institute of Technology (2009)
6. Davies, E.R.: Computer and Machine Vision: Theory, Algorithms, Practicalities. Elsevier, Academic Press (2012). ISBN: 9780123869081
7. Edmonds, J., Karp, R.: Theoretical improvements in algorithmic efficiency for network flow problems. J. ACM **19**, 248–264 (1972)
8. Ford, L., Fulkerson, D.: Flows in Networks. Princeton University Press, Princeton (1962)
9. Freedman, D., Zhang, T.: Interactive graph cut based segmentation with shape priors. In: CVPR, pp. 755–762 (June 2005)
10. Kosma, L., Lazarek J., Szczepaniak P.S.: Application of max-flow method for segmentation of medical images. Geneva Inventions (poster) (2014)
11. Malina, W., Smiatacz, M.: Digital Image Processing (in Polish: Cyfrowe Przetwarzanie Obrazow). Akademicka Oficyna Wydawnicza EXIT, Warszawa (2008)
12. Smith, S.M., Brady, J.M.: SUSAN—new approach to low level image processing. Int. J. Comput. Vis. **23**, 45–78 (1997). ISSN 920–5691
13. Sonka, M., Hlavac, V., Boyle, R.: Image Processing, Analysis and Machine Vision. Chapman & Hall Computing, Cambridge (1993)
14. Szczepaniak, P.S.: Obliczenia Inteligentne. Szybkie Przeksztalcenia i Klasyfikatory. Akademica Oficyna Wydawnicza EXIT, Warszawa (2004)
15. Szczepaniak, P.S., Tadeusiewicz, R.: The role of artificial intelligence, knowledge and wisdom in automatic image understanding. J. Appl. Comput. Sci. JACS **18**(1), 75–85 (2010)
16. Tadeusiewicz, R., Flasinski, M.: Image Recognition (in Polish: Rozpoznawanie Obrazow). Wydawnictwo Naukowe PWN (1991)

Harmony Search to Self-Configuration of Fault-Tolerant Grids for Big Data

Jerzy Balicki, Waldemar Korłub and Maciej Tyszka

Abstract In this paper, harmony search algorithms have been proposed to self-configuration of fault-tolerant grids for big data processing. Some tasks related to big data processing have been considered. Moreover, two criteria have been applied to evaluate quality of grids. The first criterion is a probability that all tasks meet their deadlines and the second one is grid reliability. Furthermore, some intelligent agents based on harmony search have been developed to support a middleware layer of grids.

Keywords Harmony search · Self-configuration · Distributed systems · Fault-tolerant grids · Big data

1 Introduction

The fault-tolerant grid deals with failures of its elements during the execution of tasks that communicate each other. An adequate resource allocation problem can be studied regarding failure/repair behaviors and fault-tolerant overhead [36]. The crucial model consists of fault-tolerant nodes where each node has many duplicated servers associated with it. One server is the primary, which performs user tasks, and some associated servers are applied for backup [16]. In another model, to tolerate

J. Balicki (✉) · W. Korłub · M. Tyszka
Faculty of Telecommunications, Electronics and Informatics, Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland
e-mail: balicki@eti.pg.gda.pl

W. Korłub
e-mail: waldemar.korlub@pg.gda.pl

M. Tyszka
e-mail: tyszka.maciej@gmail.com

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8_30

failures of nodes, a node uses other nodes as backups. When a node fails, all requests initially allocated to the node are served by one of its backups. To study the resource allocation for such grids, an approximate model can be applied. Some allocation algorithms take into account the failure/repair rates of the nodes and the fault-tolerant overheads. Although, these algorithms incur overhead, they can improve the grid performance significantly over an intuitive allocation algorithm [40]. In result, some new algorithms are still constructed [18].

Harmony search HS algorithms belong to the class of metaheuristics modeling physical phenomena related to the process of playing on musical instruments. It is worth to notice that the number of publications on these algorithms has increased rapidly over the last decade [39]. Harmony search algorithm is inspired by the process of selecting the best sound by jazz musicians while improvising [13]. Similarly, a conductor of orchestra searches the best harmony of several instruments or a compositor creates the best melody for different music lanes [1]. An inspiration to construct a harmony algorithm was based on the observation that jazz musicians ad hoc select appropriate melodic lines to get the best sound while performing the song during improvised music sessions [43]. So, we suggest applying harmony search for self-configuration of some fault-tolerant grids.

Fault-tolerant grids can be developed for big data BD. Large volumes of data are published daily by many companies to the web. We observe a migration of database capacities from terabyte (10^{12} bytes) to petabyte (10^{15} B). However, 10 terabytes is a large capacity for a financial transaction system, but small one to test a web search engine. Additionally, BD is uncooperative to work with using some relational database management systems RDBMS like DB2, INGRES, MySQL, *Oracle,* or *Sybase*. Big data requires thousands processors from systems like supercomputers [34], grids [10] or clouds [9].

In an experimental grid called *Comcute*, two kinds of tasks have been considered to implement a middleware layer. Agents for data management send data from source databases to distribution agents. Then, distribution agents cooperate with web computers to calculate results and return them to management agents. Both types of agents can autonomously move from one host to another to improve quality of grid resource using. Moreover, agents based on harmony search have been introduced to optimize big data processing regarding some fault-tolerant criteria. These schedulers can optimize using of resources. They cooperate with distributors and managers to give them information about optimal workload in a grid [12].

In this paper, harmony search algorithms are discussed in Sect. 2. Moreover, specific tasks for big data are described in Sect. 3. *Map-Reduce* model for BD processing is presented in Sect. 4. Then, intelligent agents for improvement of fault-tolerant measure are studied in Sect. 5. Moreover, some outcomes for numerical experiments are submitted in Sect. 6.

2 Harmony Search Algorithms

A generation process of new solutions that is based on harmony memory HM reminds a simulated annealing algorithm. On the other hand, the similarity between HM and a population refers to the genetic algorithm, wherein the improvisation refers to a mutation. The selection from the harmony memory reminds a genetic crossover, wherein the harmony algorithm takes into account all memory, and not just two individuals in the genetic algorithm. A diagram of the basic version of the harmony search algorithm is given in Fig. 1 [2].

Let some continuous decision variables are limited by $l = [l_1, \ldots, l_j, \ldots, l_{J_{\max}}]^T$ and $u = [u_1, \ldots, u_j, \ldots, u_{J_{\max}}]^T$, where $l_j \in R$, $u_j \in R$, $l_j \le u_j$, for $j = \overline{1, J_{\max}}$. An algorithm determines a solution for one-criterion optimization problems as follows [4]:

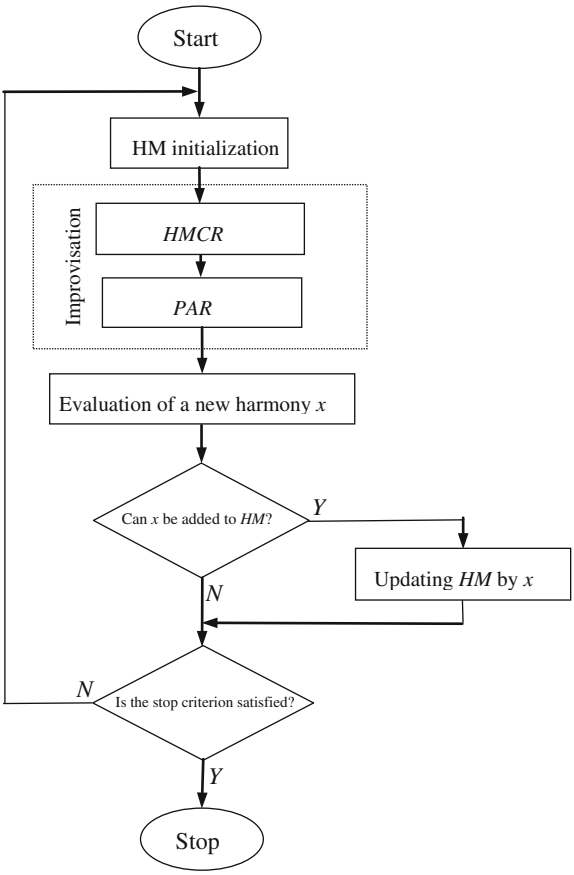


Fig. 1 A diagram of the basic version of the harmony search algorithm [27]

$$\min_{x \in X} f(x) \tag{1}$$

where
$f(x)$—a value of an objective function for solution $x \in X, f: R^{J_{\max}} \rightarrow R$;
$x = [x_1, \ldots, x_j, \ldots, x_{J_{\max}}]^T$—a vector of decision variables, $l_j \leq x_j \leq u_j$ for $j = \overline{1, J_{\max}}$;
$J_{\max}$—a number of decision variables;
$\boldsymbol{X}$—a set of decision variables.

An initialization of the harmony memory occurs after setting harmony memory size *HMS* and harmony memory considering rate *HMCR*. *HMCR* is the probability of a random event that the value of the decision variable during improvization is drawn from the memory HM (Fig. 1). An uniform distribution is assumed to draw. Moreover, pitch adjusting rate *PAR* is the rate of the randomly selected decision variable. A stop criterion is based on *NGmax*—number of generations (improvizations). Finally, *BW*—bandwidth of generations is the width of the interval to modify the value of the decision variable that is randomly selected from memory. The new value of the decision variable is modified by adding the value from the range [$-BW$, BW]. In memory HM, there are stored *HMS* randomly generated solutions with $J_{\max}$ coordinates and the corresponding fitness function values *Fitness*(x) [22].

3 Parallel Processing of Big Data

Tasks related to SQL-like queries are massive parallel, and they run in an environment based on services provided by some public cloud or grid computing platforms. It ensures a short time of a query processing. For instance, the query for multi-terabyte datasets at *BigQuery* in *Google Cloud* can be performed during few seconds. *BigQuery* service offers real-time and easy insights about scalable datasets. Moreover, this *RESTful* web service enables interactive analysis cooperating with *Google Storage*. *Google Cloud* with *BigQuery* is a sort of *IaaS Infrastructure as a Service* [14].

Tasks in big data are based on such areas like: capturing, storage, searching, sharing, analytics, and visualizing. So, we assume there are V tasks, where V is a big number of tasks ($V > 1000$) [21]. In the 4Vs model, it is considered: high *volume*, extraordinary *velocity*, great data *variety*, and *veracity*. Moreover, few terabytes of data are estimated to be captured per day from different sensors like smartphones, tablets, microphones, cameras, computers, radars, satellites, radio-telescopes and the others. Moreover, data are captured from social networks like *Twitter*. A wide *variety* of data is related to a huge range of data types and sources [24].

Some tasks can develop databases such as *MongoDB* that is the *NoSQL* database supporting data stored to different nodes. Scalability is its ability to handle an increasing amount of transactions and stored data at the same amount of time.

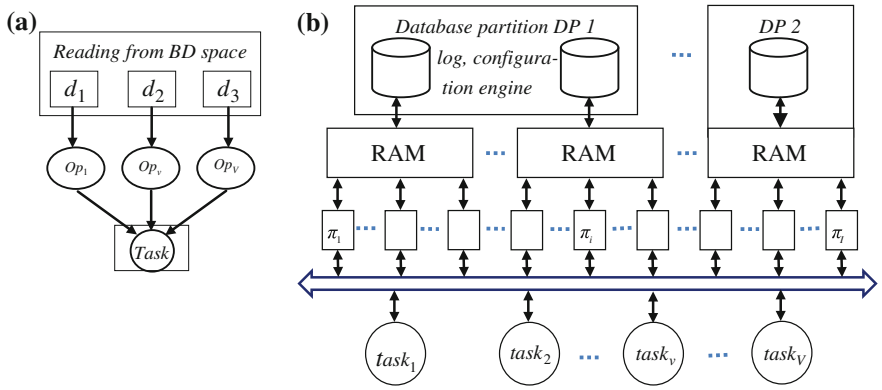


Fig. 2 Parallelism in BD: **a** reading from hard disks; **b** shared-nothing cluster architecture

We can use a massively parallel cluster with lots of CPUs, GPUs, RAM units and hard discs to obtain a high performance by data-based parallelism [25]. Complex query is divided into some parallel operations for different data (Fig. 2a). Three cases of data-based parallelism permit us to prepare two alternative architectures that support big data parallelism. Big data sets are spread over some partitions that run on some separate servers with own table spaces, logs, and configurations. A query is performed in parallel on all partitions. Such architecture can support *Google* search engine and *NoSQL* key-value stores (*Bigtable*) [33].

A crucial feature of BD is related to intensive reading from hard discs and then processing, instead of processing and then intensive writing. If we consider no sharing of memory or discs across nodes (Fig. 2b), this system requires data partitioning of database like in servers: DB2 DPF, MySQL*cluster* or *Teradata* [42].

4 Intelligent Agents in Fault-Tolerant Grids

Grid architecture has an intrinsic tendency for communication failures as interconnections between individual facilities are beyond control of any of the involved institutions and are based on general-purpose Internet connections. Fault-tolerance and partitions handling are two important aspects for successful realizations of grid systems. One programming paradigm that can provide useful features for fault-tolerant grid implementation is agent oriented programming [17].

Multi-agent systems are designed to handle complexity in distributed environments. Internal agent architectures like BDI (belief-desire-intention) or the ones based on behavioral trees are meant for handling rapid changes in the surroundings of agents [41]. Those abilities can be harnessed to cope with sudden communication failures. Another trait of agents, which is useful in such scenarios, is their proactiveness. It is the ability to not only react to changes happening in the environment

but also to take actions on their own [39]. This trait is especially important when the system becomes partitioned as agents in different temporarily disconnected facilities can carry on computations on their own without external guidance.

Heterogeneity of the grid can be handled by a multi-agent system as it provides a unified communication interface for exchanging messages and introduces a common execution environment that is not bound to any particular software or hardware stack [20]. Traits of agents mentioned above are also beneficial for BigData analysis [24]. Proactiveness is especially important in case of BD analysis, as there are often no clues about expected results known in advance so the decisions made during the actual processing of data may greatly influence final outcomes [20].

Multi-agent systems are well suited for BD acquisition also because of agent's mobility, which means the ability to move between different facilities [11]. By doing that agents can get closer to the source of data or closer to the data they are about to process. It reduces bandwidth requirements and communication delays. Agents can improve efficiency of data mining compared to centralized approaches [24]. It was applied in different domains showing promising results for further research, e.g. banking and finance domain [21] or resource allocation in distributed environments [24].

For real-time data analysis, Marz proposed the lambda architecture [24]. Twardowski and Ryżko further show that the lambda architecture can be defined in terms of a heterogeneous multi-agent system (Fig. 3) [35]. Qualities of agents prove useful when it comes to integration of individual tools that are used as basic building blocks in the lambda architecture [38]. As the AOP paradigm can be beneficial for both, the application layer—implementing the processing of BD—and the middleware layer—the underlying grid platform providing processing power—it is worthwhile to merge those two layers into a single agent-driven execution platform [37].

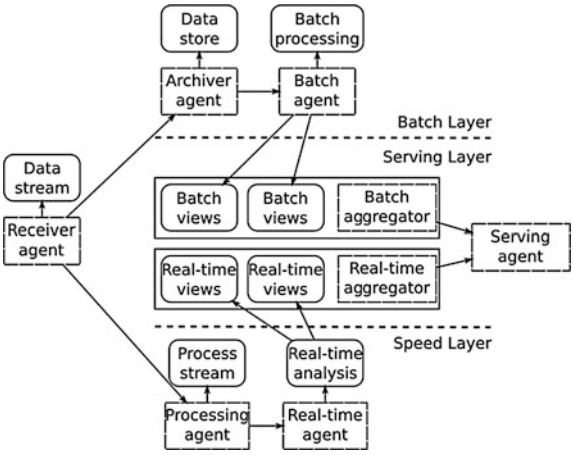


Fig. 3 Multi-agent real-time processing utilizing lambda architecture [35]

5 Map-Reduce Model for Fault-Tolerant Grid

MapReduce emerges as a popular programming model for data-intensive scalable computing. As one of the core components, task schedule comes to be a very hot topic in recent studies. However, the computing on fault-tolerant applications has not been covered yet. So, we at first point out the importance of fault-tolerant computing and propose a novel model to find out an optimal task allocation scheme, which allows us to obtain the optimal job availability. Some experiments show that the reduction of job unavailability by our method is about one order of magnitude as compared to the systematical allocation [5, 7].

Cloud computing is one of the most important techniques in nowadays distributed systems, and it draws broad attention in both, industry community and research area. Being different from super-computing systems, inexpensive hardware commodities have been widely deployed, which is helpful to the scalability of cloud system. But it also brings a large number of hardware failures. Moreover, many machines constantly restart to update systems, which cause huge software failures [12].

The popular cloud computing model *MapReduce* also has to overcome the failures. When a job consists of hundreds or thousands tasks, the possibility of a few failed tasks is very likely. In doing so, several fault-tolerant applications have been executed in the platform, which can use the result despite of some failed tasks. To support such fault-tolerant computing (FTC), an open source implementation of *MapReduce*, Hadoop has already provided the interface, by which the job can tolerate a given percentage of failed tasks. It may be observed that existing researchers have focused on the availability of an individual task. However, optimizing the availability of an individual task is not an effective approach for ensuring the high availability of these multi-task jobs [28].

In additional, *Hadoop* implicitly assumes the nodes are homogeneous, but it doesn't hold in practice. These motivate us to propose an optimal multi-task allocation scheme towards heterogeneous environments, which can tolerant a given percentage of failures to total tasks [26]. In this case the *reduce* function's responsibility is to sum the each *key's values* (Fig. 4) [32].

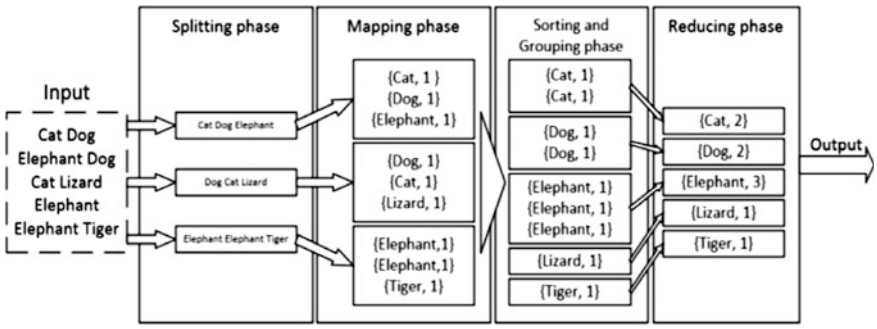


Fig. 4 *MapReduce* execution flow illustrated with the number of occurrences of each word

MapReduce is applied to solve several problems like large-scale machine learning or clustering problems for the *Google News*. Moreover, an extraction of data is used to produce reports of popular queries and extraction of geographical locations from a large corpus of web pages for localized search. In 2004, *Google* changed an indexing system that produces data used for web search service to system that uses *MapReduce*. The new indexing system takes input documents that have been retrieved from a crawling system store as a set of files, which are processed by *MapReduce* operations. This system gives many benefits like simpler, smaller, and easier to understand code, simplicity to change entire indexing process. Moreover, it is easier to operate because of automatic resolving problems like machine failures, slow machines and networking hiccups [23].

In 2008, *New York Times* released service web-based archive of scanned issues from 1851 till 1980. To create that archive company had to convert 405,000 large TIFF images, 3.3 million SGML files, and 405,000 XML files. Output consisted of 810,000 PNG images and 405,000 *JavaScript* files. To speed up this process they applied *Amazon Elastic Compute Cloud* and *Hadoop*. Regarding to hundreds of machines, the process took less than 36 h [15].

6 Agents Based on Harmony Search for Reconfiguration

Intelligent agents can optimize a grid resource management for big data queries. Some intelligent agents are based on harmony search AHSs that reconfigure some local parts of grids. AHS uses the principle of harmony search to be applied as a multi-objective scheduler for efficient using big data by the *Comcute*. This scheduler optimizes two criteria related to fault-tolerance. The first one is a probability that all tasks meet their deadlines, and the second one is the reliability of the system. The whole grid is divided into several sub-grids with at most 15 nodes [10]. We assume that each computer and each link between them are assumed to fail independently with exponential rates.

We do not take into account of the repair and recovery times for failed computers to assess the logical correctness of an allocation. Instead, we shall allocate modules to computers on which failures are least likely to occur during the execution of tasks [3]. To guard against the unlikely failures of these computers, one can assign copies of a module to multiple computers, but this subject is not the scope of this paper. The rationale behind the above assumption is that repair and recovery times are largely implementation-dependent. Moreover, repair and recovery routines usually introduce too high time overheads to be used on-line for time-critical applications [6].

Let the task T_v be executed on computers taken from the set of available computer sorts $\Pi = \{\pi_1, \ldots, \pi_j, \ldots, \pi_J\}$. The overhead performing time of the task T_v by the computer π_j is represented by t_{vj}. Let the computer π_j be failed independently due to an exponential distribution with rateλ_j. Computers can be allocated to nodes

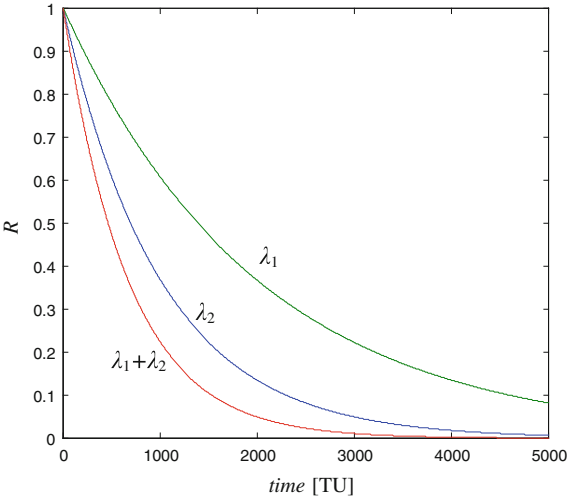


Fig. 5 The reliability of the two-computer system

and also tasks can be assigned to them in purpose to maximize the reliability function R [19]:

$$R(x) = \prod_{v=1}^{V} \prod_{i=1}^{I} \prod_{j=1}^{J} \exp(-\lambda_j t_{vj} x_{vi}^{m} x_{ij}^{\pi}), \tag{2}$$

where

$$x_{ij}^{\pi} = \begin{cases} 1 & \text{if } \pi_j \text{is assigned to the } w_i, \\ 0 & \text{in the other case.} \end{cases}$$

$$x_{vi}^{m} = \begin{cases} 1 & \text{if task } T_v \text{ is assigned to } w_i, \\ 0 & \text{in the other case,} \end{cases}$$

$$(x^m, x^\pi) = [x_{11}^m, \ldots, x_{1I}^m, \ldots, x_{vi}^m, \ldots, x_{VI}^m, x_{11}^\pi, \ldots, x_{1J}^\pi, \ldots, x_{ij}^\pi, \ldots, x_{I1}^\pi, \ldots, x_{Ij}^\pi, \ldots, x_{IJ}^\pi]^T.$$

Figure 5 shows the relation between the measure of system reliability R and time of using this system for the chosen two-computer system.

7 Deadlines for Tasks in a New Grid Configuration

Let the distributed program P_n may begin its running after λ_n and complete before δ_n. A task flow graph characterizes the logical structure of program performance. The precedence constraints among modules and the timing constraints can be presented on the task flow graph [29]. Figure 6 shows an example of the task flow

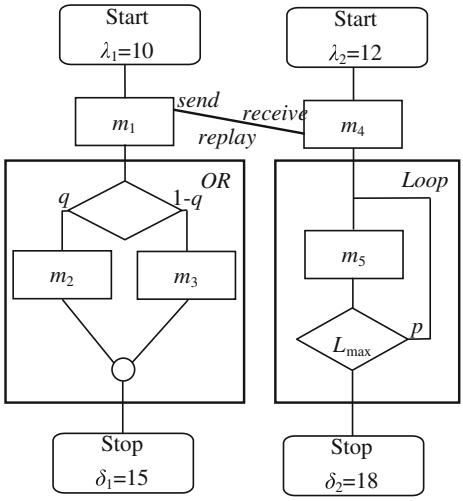


Fig. 6 The task flow graph for two programs divided into five modules

graph for two programs divided into five modules. Task m_2 is performed with the probability q in a sub-graph denoted as *OR* (Fig. 6) and task m_3—with the probability $(1 - q)$. Task may be performed at the most $L_{\max}$ times in a sub-graph denoted as *Loop*, and each repetition of this module is performed with the probability p.

The task flow graph is split on some instances to schedule tasks if the sub-graph *OR* appears. For example, the first instance incorporates the module m_2 instead of the sub-graph *OR* and this instance emerges with frequency q. The second instance incorporates the module m_3 instead of the sub-graph OR and it appears with frequency $(1 - q)$. For the sub-graph *Loop*, $L_{\max}$ instances are designed, and module is run k times for each instance ($k = 1, 2, \ldots, L_{\max}$). The instance, where module runs k times, can be met with the probability $(1 - p)\, p^{k-1}$. There are $2L_{\max}$ instances for the task graph from Fig. 6. The instance, where m_2 appears and m_5 runs k times, occurs with the probability:

$$p_i = q(1-p)p^{k-1} \tag{3}$$

An allocation of modules to computers (x^m, x^π) creates possibility to schedule tasks for each computer. Times of task completions $(C_1,\ldots,C_v,\ldots,C_V)$ can be calculated for scheduled allocation modules to computers x. Let d_v represent the completion deadline for the vth task.

If $C_v \le d_v$, then the time constraint is satisfied what can be written as $\xi(d_v - C_v) = 1$. If the deadline is exceeded then $\xi(d_v - C_v) = 0$. The state of deadline constraints regarding the ith instance of the flow graph with the set of tasks marked M_i is determined as below [30]:

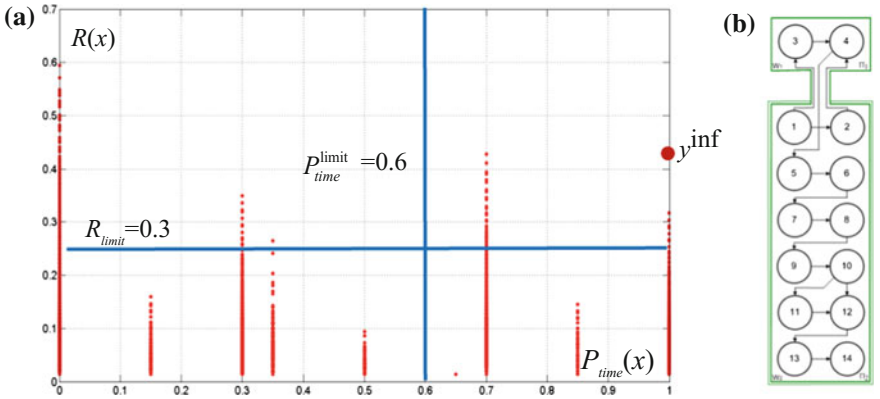


Fig. 7 Finding a compromise configuration by *AHS1* for a subgrid in the *Comcute* system: a criterion space (**a**) and a compromise configuration (**b**)

$$S_i = \prod_{m_v \in M_i} \xi(d_v - C_v(x)) \tag{4}$$

If at least one task exceeds the deadline, then deadline constraint for the *i*th instance is not satisfied. Probability that all tasks meet their deadlines for *K* instances of the flow graph is calculated from [31]:

$$P_D(x) = \sum_{i=1}^{K} p_i \prod_{m_v \in M_i} \xi(d_v - C_v(x)) \tag{5}$$

One *AHS* can find suboptimal configuration for at most 15 nodes, 50 tasks and 15 alternatives of resource sets *ARS* on *PC/Windows 7/Intel Core i7-2670 QM 2,2 GHz*. Figure 7 shows a practical example of a compromise configuration for a subgrid in the *Comcute* that was found by *AHS1*.

8 Concluding Remarks and Future Work

Shared-nothing cluster architecture for big data can be extended by cooperation with volunteer and grid computing. Moreover, intelligent agents in the middleware layer of grid can significantly support efficiency of the proposed approach. Multi-objective harmony search is a relatively a new paradigm of artificial intelligence and can be used for finding Pareto-optimal configuration of the grid. Agents based on harmony search can solve the NP-hard optimization problem of grid resource via improving the level of fault-tolerance. Harmony search agents for grid

configuration complete a new approach with respect to the actual state of art in the studied field.

Our future works will focus on testing harmony search to find fault-tolerant configurations of grids. Moreover, quantum-inspired algorithm can support big data, too [8]. Finally, genetic programming based agents can also be compared to harmony search algorithms for self-configuring grids.

Acknowledgements This research is supported by Department of Computer Architecture, Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology under statutory activity grant.

References

1. Afshari, S., Aminshahidy, B., Pishvaie, M.R.: Application of an improved harmony search algorithm in well placement optimization using streamline simulation. J. Petrol. Sci. Eng. **78**, 664–678 (2011)
2. Ahmed, A.M., Bakar, A.A. Hamdan, A.R.: Harmony search algorithm for optimal word size in symbolic time series representation. In: Proceedings of Conferences on Data Mining and Optimization, Malaysia, pp. 57–62 (2011)
3. Ajith, A.P., Murthy, C.S.R.: Algorithms for reliability-oriented module allocation in distributed computing systems. J. Syst. Softw. **40**, 125–138 (1998)
4. Al-Betar, M.A., Khader, A.T., Zaman, M.: University course timetabling using a hybrid harmony search metaheuristic algorithm. IEEE Trans. Syst. Man Cybern: Part C: Appl. Rev **42**, 66–681 (2012)
5. Apache Hadoop, http://hadoop.apache.org/, Accessed (2015)
6. Balicki, J.: Negative selection with ranking procedure in tabu-based multi-criterion evolutionary algorithm for task assignment. In: Alexandrov, V.N., et al. (eds.) Proceedings the 6th International Conference on Computational Science. Lecture Notes in Computer Science, 3993, pp. 863–870. Reading, England (2006)
7. Balicki, J.: Multi-criterion optimisation of distributed system performance by evolutionary task assignments. J. Res. Pract. Inf. Technol. **33**, 173–185 (2001)
8. Balicki, J.: An adaptive quantum-based multiobjective evolutionary algorithm for efficient task assignment in distributed systems. In: Mastorakis, N. et al. (eds.) Recent Advances in Computer Engineering. Proceedings of the 13th WSEAS International Conferences on Computers, pp. 417–422, Rhodes, Greece (2009)
9. Balicki, J.: Genetic Programming with negative selection for volunteer computing system optimization, In: Paja, W.A., Wilamowski, B.M (eds.) Proceedings on the 6th International Conferences on Human System Interactions, pp. 271–278, Gdansk, Poland (2013)
10. Balicki, J., Korłub, W., Szymańsk, J., Zakidalski, M.: Big data paradigm developed in volunteer grid system with genetic programming scheduler. In: Rutkowski, L. et al, (eds.) Artificial Intelligence and Soft Computing. Lecture Notes in Computer Science, 8467. Proceedings of the 13th International Conferences on Artificial Intelligence and Soft Computing ICAISC, Part II, pp. 771–782, Zakopane, Poland (2014)
11. Cao, L., Gorodetsky, V., Mitkas, P.A.: Agent mining: the synergy of agents and data mining. IEEE Intell. Syst. **24**, 64–72 (2009)
12. Comcute, http://comcute.eti.pg.gda.pl/. Accessed (2015)
13. Geem, Z.W., Kim, J.H., Loganathan, G.V.: A new heuristic optimization algorithm: harmony search. Simulation **76**, 60–68 (2001)

14. Gunarathne, T. et al.: Cloud computing paradigms for pleasingly parallel biomedical applications. In: Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, pp. 460–469, Chicago, Illinois (2010)
15. Guojun, L., Ming, Z., Fei, Y.: Large-scale social network analysis based on MapReduce. In: Proceedings International Conference on Computational Aspects of Social Networks, pp. 487–490 (2010)
16. Huang, Z., Wang, C., Liu, L., Peng, Y.: Improve availability of fault-tolerant computing: optimal multi-task allocation in MapReduce. In: Proceedings on 7th International Conference on Computer Science & Education, pp. 249–254 (2012)
17. Jennings, N.R., Wooldridge, M.: Applications of intelligent agents. In: Jennings, N.R., Wooldridge, M. (eds.) Intelligent Agents, pp. 3–28. Springer, New York (1998)
18. Kafil, M., Ahmad, I.: Optimal task assignment in heterogeneous distributed computing systems. IEEE Concurrency **6**, 42–51 (1998)
19. Kartik, S., Murthy, C.S.R.: Task allocation algorithms for maximizing reliability of distributed computing systems. IEEE Trans. Comput. **46**, 719–724 (1997)
20. Leyton-Brown, K., Shoham, Y.: Multiagent Systems: Algorithmic, Game-theoretic, and Logical Foundations. Cambridge University Press, UK (2008)
21. Li, H.X., Chosler, R.: Application of multilayered multi-agent data mining architecture to bank domain. In: Proceedings the International Conferences on Wireless Communications, Networking and Mobile Computing, pp. 6721–6724 (2007)
22. Manjarres, D., et al.: A Survey on Applications of the Harmony Search Algorithm. Eng. Appl. Artif. Intell. **26**, 1818–1831 (2013)
23. Mardani, S., Akbari, M.K., Sharifian, S.: Fraud detection in process aware information systems using MapReduce. In: Proceedings on Information and Knowledge Technology, pp. 88–91 (2014)
24. Marz, N., Warren, J.: Big Data – Principles and Best Practices of Scalable Realtime Data Systems. Manning Pub. Co., USA (2014)
25. O'Leary, D.E.: Artificial intelligence and big data. IEEE Intell. Syst. **28**, 96–99 (2013)
26. Ostrowski, D.A.: MapReduce design patterns for social networking analysis. In: Proceedings International Conference on Semantic Computing, pp. 316–319 (2014)
27. Paluszak, J.: Optimizing the Use of Resources in Distributed Systems with Grid Architecture, (Ph.D. Dissertation). Gdańsk University of Technology, Gdańsk, Poland (2015)
28. Qiu, X. et al.: Using MapReduce technologies in bioinformatics and medical informatics. In: Proceedings International Conference for High Performance Computing, Networking, Storage and Analysis, pp. 22–30, Portland (2009)
29. Sarvari, H., Zamanifar, K.: A Self-adaptive harmony search algorithm for engineering and reliability problems. In: Second International Conferences on Computer Intelligence, Modelling and Simulation, pp. 59–64 (2010)
30. Schneidewind, N.: Allocation and Analysis of Reliability: Multiple Levels: System, Subsystem, and Module, Innovations in System and Software Engineering, 2, pp. 121–136. Springer, London (2006)
31. Shatz, S.M., Wang, J.P.: Models & algorithms for reliability-oriented task-allocation in redundant distributed-computer systems. IEEE Trans. Reliab. **38**, 16–27 (1989)
32. Shvachko, K. et al.: The hadoop distributed file system. In: The 26 Symposium on Mass Storage System and Technology, pp. 1–10 (2010)
33. Snijders, C., Matzat, U., Reips, U.D.: 'Big Data': big gaps of knowledge in the field of internet. Int. J. Internet Sci. **7**, 1–5 (2010)
34. Shwe, T., Win, A.: A fault tolerant approach in cluster computing system. In: The 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 1, pp. 149–152 (2010)
35. Twardowski, B., Ryzko, D.: Multi-agent architecture for real-time big data processing. In: Proceedings International Conferences on Web Intelligence and Intelligent Agent Technologies 3, pp. 333–337 (2014)

36. Varvarigou, T., Trotter. J.: Module replication for fault-tolerant real-time distributed systems. IEEE Trans. Reliab. **47**, 8–18 (1998)
37. Vavilapalli, V.K.: Apache hadoop yarn: yet another resource negotiator. In: Proceedings of the 4th Annual Symposium on Cloud Computing, pp. 5:1–5:16, New York, USA (2013)
38. Verbrugge, T., Dunin-Kęplicz, B.: Teamwork in Multi-agent Systems. A Formal Approach. Wiley (2010)
39. Wang, L., Li. L.P.: A coevolutionary differential evolution with harmony search for reliability-redundancy optimization. Expert Syst. Appl. **39**, 5271–5278 (2012)
40. Węglarz, J., Błażewicz, J., Kovalyov, M.: Preemptable malleable task scheduling problem. IEEE Trans. Comput. **55**, 486–490 (2006)
41. Wooldridge, M: Introduction to Multiagent Systems. Wiley, (2002)
42. Zhou, D. et al.: Multi-agent distributed data mining model based on algorithm analysis and task prediction. In: Proceedings 2nd International Conferences on Information Engineering and Computer Science pp. 1–4 (2010)
43. Zou, D., et al.: A Novel Global Harmony Search Algorithm for Reliability Problems. Comput. Ind. Eng. **58**, 307–316 (2010)

Author Index

Z. Kowalczuk (ed.), *Advanced and Intelligent Computations in Diagnosis and Control*, Advances in Intelligent Systems and Computing 386,
DOI 10.1007/978-3-319-23180-8