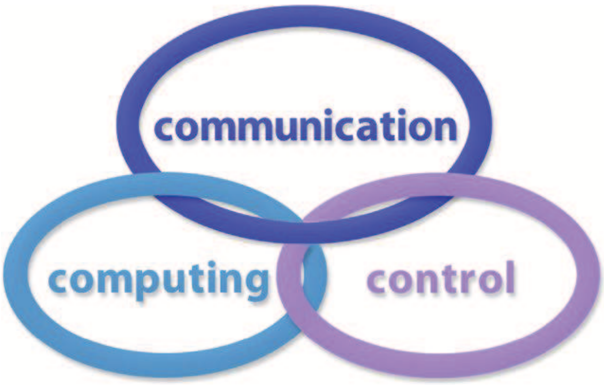
INTERNATIONAL JOURNAL

of

COMPUTERS COMMUNICATIONS & CONTROL

ISSN 1841-9836



A Bimonthly Journal
With Emphasis on the Integration of Three Technologies

Year: 2016 Volume: 11 Issue: 1 (February)

This journal is a member of, and subscribes to the principles of, the Committee on Publication Ethics (COPE).

C O P E
Member since 2012
JM08090

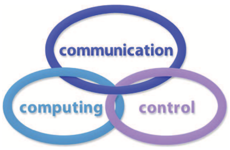
CCC Publications - Agora University

CCC Publications

http://univagora.ro/jour/index.php/ijccc/

BRIEF DESCRIPTION OF JOURNAL

Publication Name: International Journal of Computers Communications & Control.
Acronym: IJCCC; **Starting year of IJCCC:** 2006.
Abbreviated Journal Title in JCR: INT J COMPUT COMMUN.
International Standard Serial Number: ISSN 1841-9836.
Publisher: CCC Publications - Agora University of Oradea.
Publication frequency: Bimonthly: Issue 1 (February); Issue 2 (April); Issue 3 (June); Issue 4 (August); Issue 5 (October); Issue 6 (December).
Founders of IJCCC: Ioan DZITAC, Florin Gheorghe FILIP and Mişu-Jan MANOLESCU.
Logo:



Indexing/Coverage:

- Since 2006, Vol. 1 (S), IJCCC is covered by Thomson Reuters and is indexed in ISI Web of Science/Knowledge: Science Citation Index Expanded.
- Journal Citation Reports (JCR2014 - Science Edition): IF/3 years = 0.746, IF/5 years = 0.739.
  Subject Category:
  - Automation & Control Systems: Q4 (47 of 58);
  - Computer Science, Information Systems: Q3 (96 of 139).
- Since 2008 IJCCC is covered by Scopus (2014: SNIP = 1.029, IPP = 0.619, SJR = 0.450).
  Subject Category:
  - Computational Theory and Mathematics: Q3;
  - Computer Networks and Communications: Q2;
  - Computer Science Applications: Q2.
- Since 2007, 2(1), IJCCC is covered in EBSCO.

Focus & Scope: International Journal of Computers Communications & Control is directed to the international communities of scientific researchers in computer and control from the universities, research units and industry.

To differentiate from other similar journals, the editorial policy of IJCCC encourages the submission of original scientific papers that focus on the integration of the 3 "C" (Computing, Communication, Control).

In particular the following topics are expected to be addressed by authors:

- Integrated solutions in computer-based control and communications;
- Computational intelligence methods (with particular emphasis on fuzzy logic-based methods, ANN, evolutionary computing, collective/swarm intelligence);
- Advanced decision support systems (with particular emphasis on the usage of combined solvers and/or web technologies).

IJCCC EDITORIAL TEAM

Editorial Address:
Agora University/ R&D Agora Ltd. / S.C. Cercetare Dezvoltare Agora S.R.L.
Piata Tineretului 8, Oradea, jud. Bihor, Romania, Zip Code 410526
Tel./ Fax: +40 359101032
E-mail: ijccc@univagora.ro, rd.agora@univagora.ro, ccc.journal@gmail.com
Journal website: http://univagora.ro/jour/index.php/ijccc/

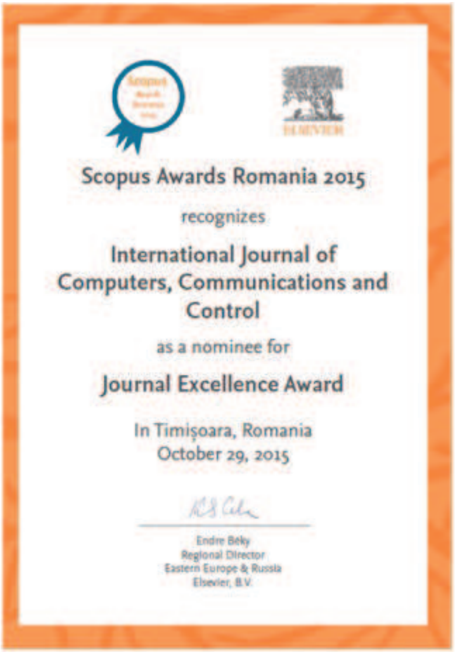
IJCCC EDITORIAL BOARD MEMBERS

DATA FOR SUBSCRIBERS

Supplier: Cercetare Dezvoltare Agora Srl (Research & Development Agora Ltd.)
Fiscal code: 24747462
Headquarter: Oradea, Piata Tineretului Nr.8, Bihor, Romania, Zip code 410526
Bank: BANCA COMERCIALA FEROVIARA S.A. ORADEA
Bank address: P-ta Unirii Nr. 8, Oradea, Bihor, România
IBAN Account for EURO: RO50BFER248000014038EU01
SWIFT CODE (eq.BIC): BFER

Nomination by Elsevier for "Journal Excellence Award"- Scopus Awards 2015

`https://www.elsevier.com/solutions/scopus/promo/scopus_awards_romania/award-2`



Scopus Awards Romania 2015

recognizes

International Journal of
Computers, Communications and
Control

as a nominee for

Journal Excellence Award

In Timişoara, Romania
October 29, 2015

Endre Béky
Regional Director
Eastern Europe & Russia
Elsevier, B.V.

"Research Excellence Award will recognize outstanding scientific journals as seen by modern bibliometric methods. To identify such journals, SNIP - Source Normalized Impact per Paper indicator is used, based on the data from Scopus database.

SNIP (Source Normalized Impact per Paper) measures a source's contextual citation impact by weighting citations based on the total number of citations in a subject field. It helps you make a direct comparison of sources in different subject fields.

SNIP takes into account characteristics of the source's subject field, which is the set of documents citing that source. SNIP especially considers: the frequency at which authors cite other papers in their reference lists; the speed at which citation impact matures; the extent to which the database used in the assessment covers the field's literature.

SNIP is the ratio of a source's average citation count per paper and the citation potential of its subject field.

The citation potential of a source's subject field is the average number of references per document citing that source. It represents the likelihood of being cited for documents in a particular field. A source in a field with a high citation potential tends to have a high impact per paper.

Citation potential is important because it accounts for the fact that typical citation counts vary widely between research disciplines. For example, they tend to be higher in life sciences than in mathematics or social sciences. If papers in one subject field contain an average of 40 cited references while those in another contain an average of 10, then the former field has a citation potential that is 4 times higher than that of the latter.

Citation potential also varies between subject fields within a discipline. For instance, basic journals tend to show higher citation potentials than applied or clinical journals, and journals covering emerging topics tend to have higher citation potentials than periodicals in well established areas."

Interview with Editor-in-Chief Ioan Dzitac

https://www.elsevier.com/solutions/scopus/promo/scopus_awards_romania/award-2/ijccc-interview

Elsevier: How do you feel about being nominated for Scopus Awards 2015?

Editor-in-Chief Ioan Dzitac: The whole team is honored by the nomination which we believe is an acknowledgement of all the hard work that stands behind the publishing of International Journal of Computers Communication & Control (IJCCC). We would like to thank our authors, reviewers and the entire team for all their dedication, originality and hard work.

Elsevier: What gap do you think your journal fills in your respective field of research?

Editor-in-Chief Ioan Dzitac: IJCCC has been focused from the very beginning on promoting research that integrates the "3Cs" - Computing, Communication and Control as to N. Wiener's theory in order to try and differentiate ourselves from the other journals indexed in the same category by Scopus.

Elsevier: If you could pick 5 articles of great importance for your field of research that have been published in your journal which would those be and why?

Editor-in-Chief Ioan Dzitac: I have decided to choose 5 of the most cited articles according to international databases such as Scopus and SCI Expanded:
(1) Spiking neural P systems with anti-spikes, by L. Pan, G. Paun, 2009.
(2) Tissue P systems with cell division, by G. Paun,M.J. Pérez-Jiménez, A. Riscos-Núnez, 2008.
(3) Computing Nash equilibria by means of evolutionary computation, by R.I. Lung, D. Dumitrescu, 2008.
(4) Lorenz system stabilization using fuzzy controllers, by R.E. Precup, M.L. Tomescu, S. Preitl, 2007.
(5) Neuro-fuzzy based approach for inverse kinematics solution of industrial robot manipulators, by S. Alavandar, M.J. Nigam, 2008.

Elsevier: For you, as Editor-in-chief, what is the most important development objective for 2016?

Editor-in-Chief Ioan Dzitac: Reinforcing the intellectual current we have created through publishing high quality articles that bring new ideas and multidisciplinary approaches.

Elsevier: Why did you choose to publish your journal Open Access?

Editor-in-Chief Ioan Dzitac: We have never followed financial gain as our main goal has been to publish high quality articles with a real effect on the evolution of society.

Elsevier: What do you think makes your journal stand out?

Editor-in-Chief Ioan Dzitac: First of all it stands out through the approach of the subject by integrating the 3Cs. On top of this I would add the geographical distribution of the authors that come from over 45 countries and are affiliated to more than 150 universities. The prestige of the editorial board that included researchers from 14 countries.

The editorial team is another strong point as it includes researchers affiliated to high ranked universities from top 100 QS (1. Massachusetts Institute of Technology (MIT), 17. Cornell University, 24. McGill University, 25. Tsinghua University, 26. University of California Berkeley, Berkeley (UCB), 56. Tokyo Institute of Technology, 70. Shanghai Jiao Tong University, 80. University of Sheffield, 89. Purdue University, 96. University of Alberta).

Finally I would like to point out our association with the International Conference on Computers Communications and Control.

6th INTERNATIONAL CONFERENCE on COMPUTERS, COMMUNICATIONS and CONTROL (ICCCC 2016)

Hotel President, Baile Felix, Oradea, Romania, May 10-14, 2016
Organized by Agora University of Oradea,
under the aegis of Romanian Academy: Information Science and Technology Section.
http://univagora.ro/en/icccc2016/

Scope and Topics

The International Conference on Computers Communications and Control (ICCCC) has been founded in 2006 by I. Dzitac, F.G. Filip and M.-J. Manolescu and organized every even year by Agora University of Oradea, under the aegis of the Information Science and Technology Section of Romanian Academy and IEEE - Romania Section.

The goal of this conference is to bring together international researchers, scientists in academia and industry to present and discuss in a friendly environment their latest research findings on a broad array of topics in computer networking and control.

The Program Committee is soliciting paper describing original, previously unpublished, completed research, not currently under review by another conference or journal, addressing state-of-the-art research and development in all areas related to computer networking and control.

In particular the following topics are expected to be addressed by authors:

1) Integrated solutions in computer-based control and communications;
2) Network Optimization and Security;
3) Computational intelligence methods (with particular emphasis on fuzzy logic-based methods, ANN, evolutionary computing, collective/swarm intelligence);
4) Data Mining and Intelligent Knowledge Management;
5) Advanced decision support systems (with particular emphasis on the usage of combined solvers and/or web technologies);
6) Membrane Computing - Theory and Applications;
7) Stereovision Based Perception for Autonomous Mobile Systems and Advanced Driving Assistance.

Special Sessions

Special Session 1: Network Optimization and Security, Organizer and Chair: Yezid DONOSO (Colombia);
Special Session 2: Data Mining and Intelligent Knowledge Management, Organizers and Chairs: Gang KOU and Yi PENG (China);
Special Session 3: Computational Intelligence Methods, Organizers and Chairs: Razvan ANDONIE and Donald DAVENDRA (USA);
Special Session 4: Advanced Decision Support Systems, Organizers and Chairs: Marius CIOCA (Romania) and Felisa CORDOVA (Chile);
Special Session 5: Fuzzy Control, Modeling and Optimization, Organizer and Chair: Radu-Emil PRECUP (Romania);
Special Session 6: Membrane Computing - Theory and Aplications, Organizers and Chairs: Marian GHEORGHE (UK) and Florentin IPATE (Romania);
Special Session 7: Stereovision Based Perception for Autonomous Mobile Systems and Advanced Driving Assistance, Organizer and Chair: Sergiu NEDEVSCHI (Romania).
Keynote Speakers: Enrique HERRA VIEDMA(Spain), Zenonas TURSKIS (Lithuania), Gang KOU (China).
Conference Chairs: Ioan DZITAC, Florin Gheorghe FILIP and Misu-Jan MANOLESCU.

Contents

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):11-25, February 2016.

A Fuzzy Networked Control System Following Frequency Transmission Strategy

H. Benítez-Pérez, J. Ortega-Arjona, O. Esquivel-Flores,
J.A. Rojas-Vargas, A. Álvarez-Cid

Héctor Benítez-Pérez*
IIMAS UNAM
Cto. Escolar 3000, C. U., México D. F.
*Correspondig author: hector.benitez@iimas.unam.mx

Jorge Ortega-Arjona
Facultad de Ciencias UNAM
Av. Universidad 3000, C. U., México D. F.
jloa@ciencias.unam.mx

Oscar Esquivel-Flores, Jared A. Rojas-Vargas
IIMAS UNAM
Cto. Escolar 3000, C. U., México D. F.
oaefmcc@hotmail.com, jared_36_23@hotmail.com

Andrés Álvarez-Cid
Facultad de Ingeniería UNAM
Av. Universidad 3000, C.U., México D. F.
waveform@live.com.mx

Abstract: At present, network control systems employ a common approximation to solve the connectivity issue due to time delays coupled with external factors . However, this approach tends to be complex in terms of time delays, and the inherent local phase is missing. Therefore, it is necessary to study the behavior of the delays as well as the integration of the differential equations of these bounded delays. The related time delays need to be known a priori, but from a dynamic real-time perspective in order to understand the dynamic phase behavior. The objective of this paper is to demonstrate the inclusion of the data frequency transmission and time delays that are bounded as parameters of the dynamic response from a real-time scheduling approximation, considering the local phase situation. The related control law is designed considering a fuzzy logic approximation for nonlinear time delays coupling. The main advantage is the integration of this behavior through extended state space representation. This keeps certain linear and bounded behavior leading to a stable situation during an events presentation, based on an accurate data transmission rate. An expected result is that the basics of the local phase missing as a result of the local bounded time delays from the lack of tide synchronization conforms to the modeling approximation.

Keywords: Fuzzy networks control (FNC), frequency control, local phase challenge.

1 Introduction

Real-time restrictions are the primary cause of time delays when general conditions tend to be periodic and repeatable . The control design and stability analysis of network-based control systems (NCSs) have been studied in recent years, and approaches such as the codesign strategy have been developed [2]. The main advantages of these types of systems are their low cost, small wiring volume, distributed processing, simple installation, ease of maintenance, and reliability.

A key issue in an NCS is the effect of network-induced delay on the system performance. The delay can be constant, time-varying, or even random depending on the scheduler, network type, architecture, or operating systems. [16] analyzed several important facets of NCSs and introduced models for their delays, including first a fixed delay, then an independently random delay, and finally a Markov process. This work introduced optimal stochastic control theorems for NCSs based upon independently random and Markovian delay models. [20], introduced static and dynamic scheduling policies for the transmission of sensor data over a continuous time linear time-invariant (LTI) system. They introduced the notion of the maximum allowable transfer interval (MATI), which is the longest time after which a sensor should transmit a data. The MATI determines that the Lyapunov function of the system under consideration is strictly decreasing at all times. [22], extended the work of Walsh et al. by developing a theorem that ensures the decrease of a Lyapunov function for a discrete-time LTI system at each sampling instant, using two different bounds. This used a number of different linear matrix inequality (LMI) tools for analyzing and designing optimally switched NCSs. The MATI is an important concept for bounded time delays and the maximum local dephasing problem.

Alternatively, [23] considered both the network induced delay and the time delay in a plant, and proposed a controller design method using the delay-dependent approach. An appropriate Lyapunov functional candidate was used to obtain a memoryless feedback controller that was derived by solving a set of LMIs. [21] modeled the network-induced delays of the NCSs as interval variables governed by a Markov chain. Using the upper and lower bounds of the delays, a discrete-time Markovian jump system with norm-bounded uncertainties was presented to model the NCSs. Based on this model, the $H\infty$ state feedback controller was constructed via a set of LMIs. [7] introduced a model transformation for the delay-dependent stability of systems with time-varying delays in terms of LMIs. Their model also refined results from delay-dependent $H\infty$ control and extended them to the case of time-varying delays. Based upon this review, the present paper defines a model that integrates time delays into a fuzzy control for NCSs [18, 22] taking into consideration the local phase margin induced by the computer network as a result of online reconfiguration.

Since NCS is modified according to time delays, reconfiguration is a transition that modifies the structure of a system so it changes its representation of states. The objective of this paper is to show how the control of frequency transmission has a bounded impact over a network control system.

In general there are two types of tasks in a NCS. The first is a periodic task that is time-triggered. In this type, tasks have a transmission time c_i, a constant period of execution p_i, and a deadline d_i. Thus, the sum of the transmission times of n nodes' tasks, divided into their periods p_i for a fixed priority scheduler [15].

The network scheduler is a high priority in the design of a distributed system and it is also critical in an NCS, since if there is no scheduling between nodes, data transmissions may occur simultaneously which lead to collisions or bandwidth violations. This behavior results in a transmission with time delays, leading to failure in complying with deadlines, data loss, and an obvious decrease in system performance due to local phase misleading. A good scheduling control algorithm minimizes the decrease in system performance [4]; nevertheless, there are no global schedulers that guarantee optimal system performance [15]. Some strategies include methods in which nodes generate proper control actions in order to optimally utilize bandwidth [11, 12]. In the digital control case, the performance only depends on the sampling frequency without uncertainties. For networked control, the minimum transmission frequency f_m is necessary to guarantee suitable system performance. As the transmission frequency increases the system performance improves; however, the load on the network also increases until a maximum transmission frequency f_M is reached, then the system performance decreases because the network is overloaded. It is very important to modify the transmission frequency to obtain better system performance within a bounded region that is particularly defined for the current system needs.

In control systems, several modeling strategies for managing time delay within control laws have been studied by different research groups. [16] proposed the use of a time delay scheme integrated to a reconfigurable control strategy, based on a stochastic methodology. [9] described how time delays are used as uncertainties, which modify pole placement of a robust control law. [8] presented an interesting case of fault tolerant control approach related to time delay coupling. [3] studied reconfigurable control from the point of view of structural modification, establishing a logical relation between dynamic variables and the respective faults. Finally, [19] and [2] considered that reconfigurable control strategies perform a combined modification of system structure and dynamic response; thus, this approach has the advantage of bounded modifications over system response.

The present paper proposes a mixed strategy using bounded variable time delays and frequency transmission as the data communication rate. The novelty of this approximation is that it guarantees schedulability as well as stability in the presence of bounded time delays, considering an accurate data transmission rate through the network. This is feasible since the time delays are bounded according to scheduler response.

2 Frequency control

As presented in [17], a fuzzy approach to network control systems allows a real representation of bounded time delays and control design. Specifically, this can be used when time delays are the result of the controlling frequency transmission, as presented previously. Several potential scenarios are presented that follow this time delay behavior. In fact, the number of scenarios is finite since the combinatorial formation is bounded. Therefore, any strategy for designing a control law needs to take into account the gain scheduling approximation.

Therefore, an alternative strategy is the use of control for periodic messages, namely the control of frequency transmissions as presented by [6]. The change in frequency transmission of each node through time is represented using the relations of frequencies for a particular node and among all nodes. This means that the frequency transmission rate of each node is influenced by changes in the transmission rate of the other nodes.

Upper and lower bounding are necessary since the data transmission frequency of the network can be high. Due to uncertainties during sensing, the system may become unstable; similarly, the low data frequency transmission may result in undersampling which goes to control performance weakness. Having defined this dynamic bounding strategy, it becomes interesting to examine the effects of known time delays over a well-defined network control approximation.

3 Modified Frequency Transmission for NCS's

The following section presents an illustrative experiment considering a twin rotor case study as well as a magnetic levitation system (Fig. 1, 2) These cases include the local dynamics necessary for a complete networked control system where a global computer network is implemented.

3.1 Twin Rotor and Magnetic Levitation System Equations

The free-body diagram of the twin Rotor is illustrated in Figure 1 and from this the next nonlinear equations of motion are obtained using the Euler-Lagrange formula [24]:

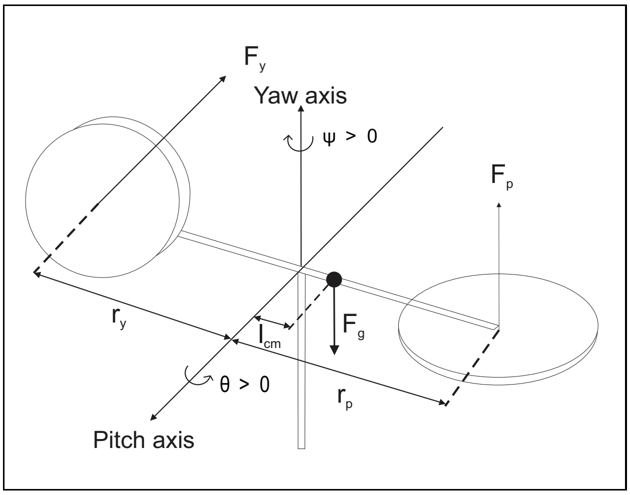


Figure 1: Twin Rotor.

$$\begin{aligned}\ddot{\theta} &= \frac{1}{(J_{eqp}+m_{heli}l_{cm}^2)}(K_{pp}V_{mp}+K_{py}V_{my}-\\ &m_{heli}l_{cm}^2(sin(\theta)cos(\theta))\dot{\psi}^2-cos(\theta)m_{heli}gl_{cm}-B_{qp}\dot{\theta})\\ \ddot{\psi} &= \frac{1}{(J_{eqy}+m_{heli}l_{cm}^2cos(\theta)^2)}(K_{yp}V_{mp}+K_{yy}V_{my}+\\ &2m_{heli}l_{cm}^2cos(\theta)sin(\psi)\dot{\theta}\dot{\psi}-B_{qy}\dot{\psi})\end{aligned} \tag{1}$$

where:
J_{eqp} Total moment of inertia about pitch axis
J_{eqy} Total moment of inertia abut yaw axis
K_{pp} Thrust force constant acting on pitch axis from pitch motor/propeller
K_{yy} Thrust force constant acting on yaw axis from yaw motor/propeller
K_{py} Thrust force constant acting on pitch axis from yaw motor/propeller
K_{yp} Thrust force constant acting on yaw axis from pitch motor/propeller
B_{qp} Equivalent viscous damping about pitch axis
B_{qy} Equivalent viscous damping about yaw axis
l_{cm} Center of mass length along helicopter body from pitch axis
m_{heli} Total moving mass of helicopter
g Gravitational constant

Table 1 shows the values of this parameters.

In the case of magnetic levitation system model [25] specifications are derived from the mechanical and electric system representation as illustrated in Figure 2.

The nonlinear equations for the Magnetic Levitation System are:

$$\begin{aligned}\dot{x_1} &= x_2\\ \dot{x_2} &= \frac{-K_m x_3^2}{2M_b(x_1)^2}+g\\ \dot{x_3} &= \frac{1}{L_c}(-Rx_3+u)\end{aligned}$$

where $R = R_c + R_s$ and $u = V_c$ input voltage and

Table 1: System Specifications

Symbol	**Value**	**Unit**
J_{eqp}	0.0348	$kg \cdot m^2$
J_{eqy}	0.0432	$kg \cdot m^2$
K_{pp}	0.204	$N \cdot m/V$
K_{yy}	0.072	$N \cdot m/V$
K_{py}	0.0068	$N \cdot m/V$
K_{yp}	0.0219	$N \cdot m/V$
B_{qp}	0.800	N/V
B_{qy}	0.318	N/V
m_{heli}	1.3872	kg
l_{heli}	0.186	m

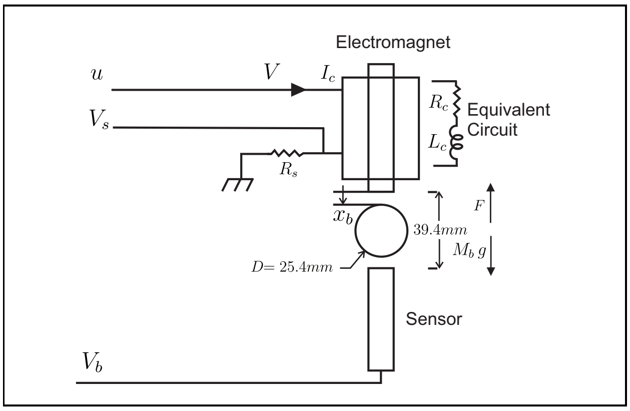


Figure 2: Magnetic Levitation System

Table 2: System parameters

Symbol	**Value**	**Unit**
L_c	0.4125	H
R_c	10	Ω
R_s	1	Ω
k_m	$6.5308e^{-5}$	Nm^2/Amp^2
M_b	0.068	kg
g	9.81	m/s^2

R_c electromagnet resistance
R_s resistor in series wiht the coil
K_m constant of electromagnet force
M_b mass of the ball
g gravitacional constant
L_c coil inductance

The values of the parameters are provided in Table 2

These systems are distributed and communicated through an Ethernet network to implement a complete networked control system, as depicted in Figure 3.

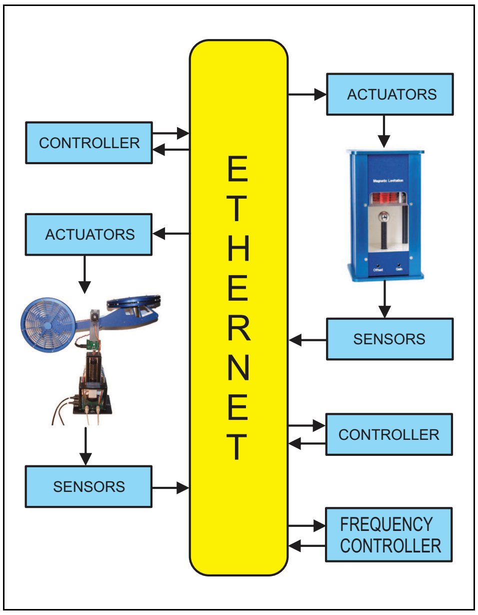


Figure 3: NCSs

In this work, we define three rules of fuzzy model to approximate the twin rotor nonlinear system, as follows:

Rule 1:

IF x_1 is about 0°,

THEN $\dot{x} = A_1 x + B_1$

Rule 2:

IF x_1 is about 40°,

THEN $\dot{x} = A_2 x + B_2$

Rule 3:

IF x_1 is about $-40°$,

THEN $\dot{x} = A_3 x + B_3$

where x_1 is the pitch angle and

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -9.2593 & 0 \\ 0 & 0 & 0 & -3.4868 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 2.3611 & 0.0787 \\ 0.2401 & 0.7895 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 18.8362 & 0 & -9.2768 & 0 \\ 2.5148 & 0 & 0 & -4.4618 \end{bmatrix}$$

$$B_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 2.3667 & 0.0789 \\ 0.3073 & 1.0102 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -18.8362 & 0 & -9.2768 & 0 \\ -2.5148 & 0 & 0 & -4.4618 \end{bmatrix}$$

$$B_3 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 2.3667 & 0.0789 \\ 0.3073 & 1.0102 \end{bmatrix}$$

Figure 4 shows the membership function for the rules of the helicopter fuzzy model.

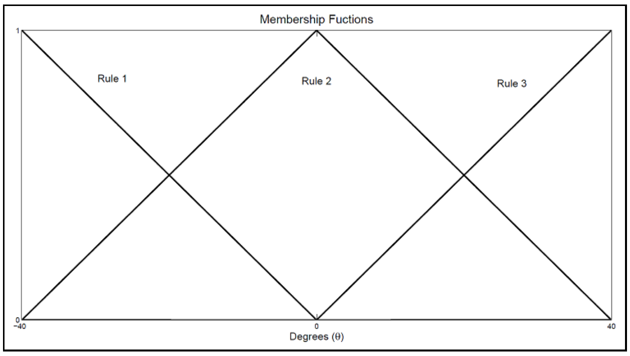


Figure 4: Membership Functions of Helicopter Model

Here, two fuzzy rules are defined to approximate the magnetic levitation system by means of two linear models, as follows:

Rule 1:

IF x_1 is about $0.006\,m$,

THEN $\dot{x} = A_1 x + B_1$

Rule 2:

IF x_1 is about $0.014\,m$,

THEN $\dot{x} = A_2 x + B_2$

where x_1 is the ball position in meters and

$$A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 4097.8 & 0 & -25.6 \\ 0 & 0 & -26.7 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1509.2 & 0 & -14.2 \\ 0 & 0 & -26.7 \end{bmatrix}$$

$$B_1 = B_2 = \begin{bmatrix} 0 \\ 0 \\ 2.4242 \end{bmatrix}$$

The membership functions for this case are shown in Figure 5

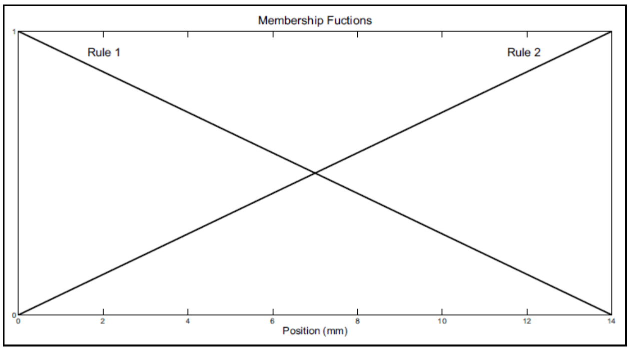


Figure 5: Membership Functions of Levitator Model

The fuzzy controller for each system is design by PDC as follows:

Control Rule i:
IF $z_i(t)$ is M_{i1} and $z_n(t)$ is M_{in}
THEN $u = -K_i x$

The feedback gain matrix K_i was obtained by means of the linear-quadratic regulator (LQR) of each linear system. The stability analysis of the fuzzy control system should be performed employing a numerical mathematical program, for example, using Matlab's linear matrix inequalities toolbox to check feasibility.

$$K_1 =$$
$$\begin{bmatrix} 18.9369 & 1.9770 & 7.4917 & 1.5258 & 7.0292 & 0.7685 \\ -2.2192 & 19.4463 & -0.4496 & 11.8949 & -0.7685 & 7.0292 \end{bmatrix}$$
$$K_2 =$$
$$\begin{bmatrix} 28.5899 & 2.0444 & 7.8331 & 1.6059 & 7.0264 & 0.7933 \\ -2.3969 & 19.3161 & -0.5528 & 11.5476 & -0.7933 & 7.0264 \end{bmatrix}$$
$$K_3 =$$
$$\begin{bmatrix} 51.0473 & 6.1419 & 28.4307 & 4.7239 & 22.2386 & 2.3333 \\ -6.3519 & 59.4161 & -2.4313 & 41.5536 & -2.3333 & 22.2386 \end{bmatrix}$$

Following the same procedure for the fuzzy helicopter controller to find the matrix gains, we obtain the following matrix:

$$K_1 = \begin{bmatrix} -11971 & -187 & 53 & 7 \end{bmatrix}$$
$$K_2 = \begin{bmatrix} -5739 & -147.7 & 32.1 & 10 \end{bmatrix}$$

Now, in terms of frequency transitions, the dynamics representation is given as follows:

$$\dot{x} = \begin{bmatrix} 0.1250 & 0.1300 & 0.1350 & 0.1400 & 0 \\ 0.1060 & 0.1111 & 0.1160 & 0.1210 & 0 \\ 0.0900 & 0.0950 & 0.1000 & 0.1050 & 0 \\ 0.0135 & 0.0185 & 0.0235 & 0.0286 & 0 \\ 0.001 & 0.001 & 0.001 & 0.001 & 1 \end{bmatrix} x$$
$$+ \begin{bmatrix} 40 & 0 & 0 & 0 & 0 \\ 0 & 45 & 0 & 0 & 0 \\ 0 & 0 & 50 & 0 & 0 \\ 0 & 0 & 0 & 175 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} u$$
$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x$$

Where
$x_1 =$ is the frequency transmission in node 1
$x_2 =$ is the frequency transmission in node 2
$\vdots$
$x_n =$ is the frequency transmission in node n
$x_c =$ is the network utilization

and

$u_1 =$ is the frequency input 1 of system
$u_2 =$ is the frequency input 2 of system
$\vdots$
$u_n =$ is the frequency input n of system
$u_c =$ is the frequency input c of system

We design a control for the trajectory tracking, which is the desired frequency transmission by means of the LQR algorithm, so that the gain matrix obtained is:

$$K = \begin{bmatrix} 3.5521 & 0.0027 & 0.0023 & 0.0007 & 0.0003 \\ 0.0030 & 3.5101 & 0.0021 & 0.0006 & 0.0003 \\ 0.0029 & 0.0024 & 3.4763 & 0.0005 & 0.0003 \\ 0.0029 & 0.0022 & 0.0017 & 3.2792 & 0.0002 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 5.7417 \end{bmatrix}$$

and the integral gain matrix is

$$K_I = \begin{bmatrix} -48.3570 & 0.0085 & 0.0150 & 0.0518 \\ -0.0085 & -48.3184 & 0.0064 & 0.0389 \\ -0.0150 & -0.0064 & -48.2873 & 0.0297 \\ -0.0398 & -0.0299 & -0.0228 & -62.5185 \\ 0.0048 & 0.0048 & 0.0048 & 0.0050 \end{bmatrix}$$

The nominal values and schedulability regions were found using Truetime [5] and are shown in Table 3. These take into account that the frequency system models 5 network nodes; but in the experiment, only 4 nodes are employed because the last node corresponds to the use of the network.

For the Stability proof it is necessary to take into account there are two transmission bound, superior and lower bound and consider the schedulability restriction.

The schedulability restriction is given taking the maximum frequency value if keeps the restriction.

In the other hand for the lower transmission value it is necessary to prove that this values are the maximum allowable delay bound which guaranty stability of the NCS. In [10] it is presented a LMI formulation to find the maximum allowable delay bound.

If there exist $P > 0$, $Q_i > 0$, X_i, Y_i and Z_i, $i = 1, ..., N$, such that

$$\begin{bmatrix} \mathfrak{P}_{11} & \mathfrak{F}^T \mathfrak{L} \\ \mathfrak{L}\mathfrak{A} & -\Gamma \end{bmatrix} < 0, \quad \begin{bmatrix} X_i & Y_i \\ Y_i^T & Z_i \end{bmatrix} \geqslant 0$$

where

$$\mathfrak{P}_{11} = \begin{bmatrix} \mathfrak{F}_{11} & P\mathfrak{F}_1 - \mathfrak{Y} \\ \mathfrak{F}_1^T P - \mathfrak{Y} & -\mathfrak{D} \end{bmatrix},$$

$$\mathfrak{F} = \begin{bmatrix} F & F_1 & ... & F_N \end{bmatrix},$$

$$\mathfrak{F}_1 = \begin{bmatrix} F_1 & ... & F_N \end{bmatrix},$$

$$\mathfrak{Y} = \begin{bmatrix} Y_1 & ... & Y_N \end{bmatrix},$$

$$\mathfrak{L} = \tau \begin{bmatrix} Z_1 & ... & Z_N \end{bmatrix},$$

Table 3: Nominal frequency, maximum, minimum and transmission time

Node	$c_i\,(s)$	$f^i_{nom}\,(Hz)$	$0.9\ f^i_{min}\,(Hz)$	$f^i_{max}\,(Hz)$
1	0.001	280	40	83
2	0.001	300	45	98
3	0.001	310	50	70
4	0.001	450	175	212
5	0.001	480	165	196
6	0.001	500	160	188

$$\mathfrak{D} = diag\left\{Q_1, ..., Q_N\right\},$$

$$\Gamma = \bar{\tau} diag\left\{Z_1, ..., Z_N\right\},$$

$$\mathfrak{F}_{11} = F^T P + PF + \sum_{i=1}^{N} \left\{Y_i + Y_i^T + \bar{\tau} X_i + Q_i\right\}$$

the system is asymptotically stable for any time-delay τ_i satisfying $0 \leqslant \tau_i \leqslant \bar{\tau}_i$, $i = 1, ..., N$ Solving LMIs with MATLAB TOOLBOX for each subsystem of the Fuzzy representation the next τ_i are found

$\tau_1 = 0.0058$, $\tau_2 = 0.0071$, and $\tau_3 = 0.0067$

Which in frequency terms are 172.42, 140.85 and 149.25 Hz respectively being consistent with the values previously found.

Now in the twin rotor case the values are:
$\tau_1 = 0.050$, $\tau_2 = 0.067$, and $\tau_3 = 0.050$

In frequency terms are 20, 15 and 20 Hz respectively which approach to the values previously found in Table 3.

4 Frequency Control Design

To implement this experiment, 7 nodes employed to obtain a complete distributed NCS configuration as follows: 3 nodes to the twin rotor system (sensor, actuator, and controller), 3 to the magnetic levitation system (sensor, actuator, and controller), and 1 more used for perform frequency control. The latter node assigns the changes in the transmission frequency to sensors and controller nodes of the systems.

The platform was built with 10 Mbps Ethernet as communication medium and Quarc adquisition boards to accomplish data exchange between nodes. Quarc Matlab's toolbox provides an useful and straightforward way to stablish communication among several nodes which contain embedded Simulink models (in this case Twin Rotor and Levitation distributed models) by means its communication blocks.

The processes run at nominal frequency transmission values according to Table 3. The controller frequency node, which contains the embedded Simulink frequency model, modifies nominal to schedulability values by means LQR controller action, as seen in Figure 6. In this figure, the red and blue signals represent the helicopter's sensor and controller transmission frequency, respectively, and the green and brown signals depict the levitator's sensor and controller transmission frequency.

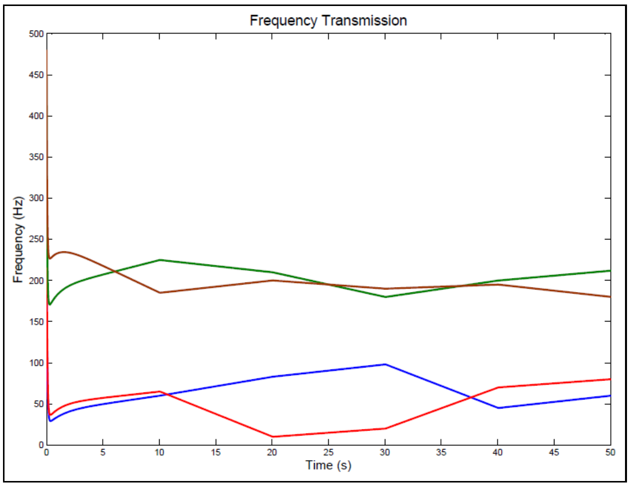


Figure 6: Frequency Transmission Transition

The results of these experiments are presented in Fig. 7 , in which the twin rotor presents a suitable response even with frequency transitions. The red color signal is the desired pitch and yaw angle, and the blue color signal is the obtained response.

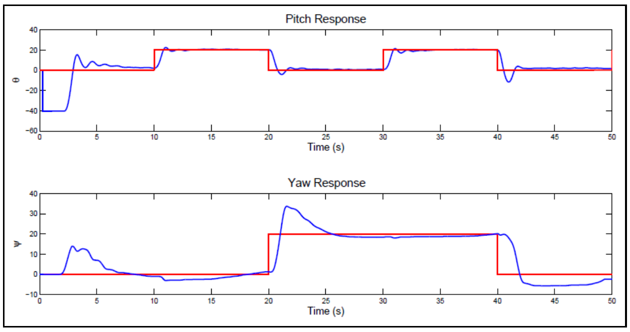


Figure 7: Pitch and Yaw angle when frequencies are whitin the bounds

In the second experiment, the transmission frequencies are outside the lower bound of the schedulability region for short time instants. It can be seen at $t = 10$ and $t = 30$ in Figure 8 where the frequencies leave the schedulability region for $5\,s$ and then return within the region.

The magnetic levitation system response, in terms of the current value and ball position during the frequency transmission change, is presented in Fig. 9. In this figure, the red color signal is the desired current and ball position, and the blue color signal is the system response.

The next result is obtained (Figure 10) when the frequencies change at $t = 25$ and $t = 40$ leaving slightly the schedulability region for $3\,s$ and $5\,s$ respectively and then the frequencies return within the region.

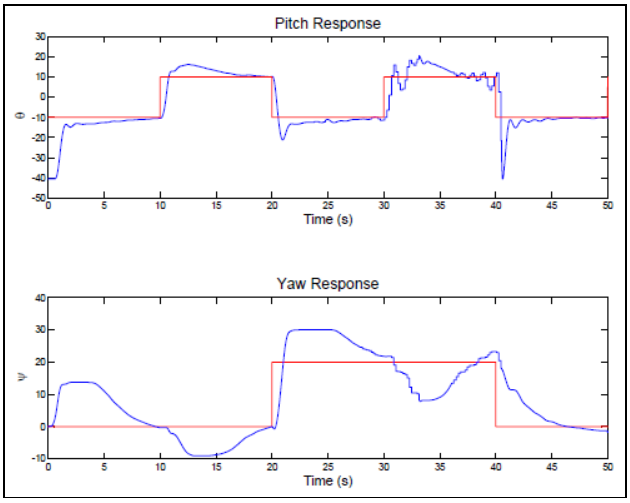


Figure 8: Pitch and Yaw angle when frequencies by instants are not within the bounds

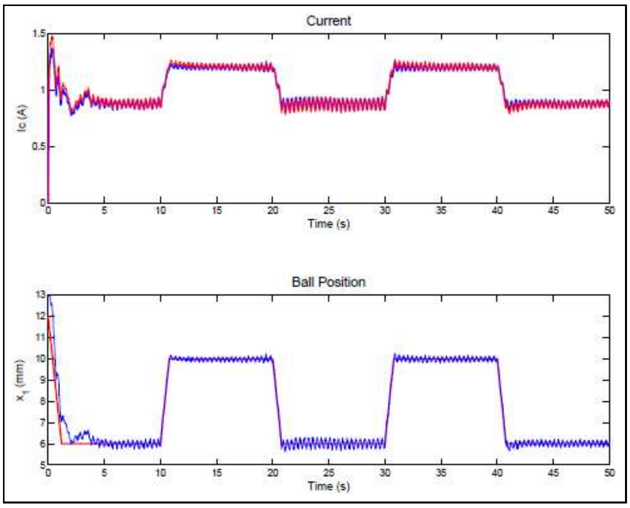


Figure 9: Current and ball position when frequencies are within the bounds

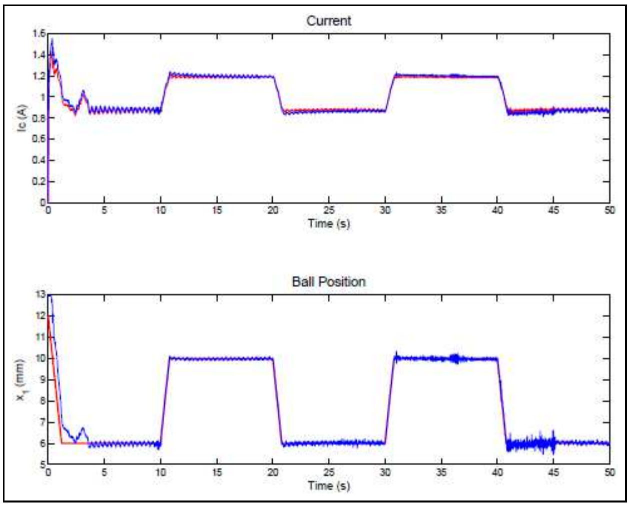


Figure 10: Current and ball position when frequencies by instants are not within the bounds

5 Conclusion

Currently, time delays can be modeled using a bounded frequency transmission control approach; however, the resulting delays are time varying and stationary. Therefore, a related local control law must be designed according to this characteristic, where time integration is the key global issue to be considered. Global stability is reached by the use of the Takagi Sugeno fuzzy control design, where nonlinear combination is followed by the current situation of the states. These are partially delayed due to the communication behavior in terms of the phasing problem. In this case, the bounded local stochastic variables are presented in terms of local phases.

The main contribution of this paper is the capability of determining the local time delays that can be aggregated per event, since the frequency transmission control contributes to the bound time response, resulting in a global problem of local synchronized elements. Therefore, fuzzy control may be attractive to guarantee global stability, since any condition is bound to be less than the sampling period of the worst-case scenario with no loss of generality. However, local phase missing is still an open problem, since these are stochastically calculated and linked to a tight synchronization situation.

The use of the dynamic scheduling approximation, as reviewed here, allows the system to be predictable and bounded; therefore, time delays can be modeled in these terms. Moreover, the resulting dynamic representation tackles the inherent switching situation per scenario. This approximation has the main drawback that the context switch may be invoked every time that a periodic task takes place, and it is possible to be executed. In this case, inherent time delays to this action are taken into account to be processed as bounded known values and to modify the related local stochastic phase. Thus, there is desynchronizing global behavior of the network.

Acknowledgments

The authors acknowledge the support of PAPIIT IN100813 and CONACYT 176556 as well as the collaboration from Mr. Adrian Duran Chavesti.

Bibliography

[1] Benítez-Pérez, H.; Benítez-Pérez A.; Ortega-Arjona J. (2012); Networked Control Systems Design considering Scheduling Restrictions and Local Faults using Local State Estimation, *Accepted in International Journal of Innovative Computing, Information and Control (IJICIC).*

[2] Benítez-Pérez, H.; García-Nocetti F. (2005); *Reconfigurable Distributed Control*, Springer Verlag.

[3] Blanke, M. et al. (2003); *Diagnosis and Fault Tolerant Control*, Springer.

[4] Branicky, M. et al. (2003); Networked control system co-simulation for co-design, *Proc. American Control Conf*, 4:3341-3346.

[5] Cervin, A. et al. (2003); How Does Control Timing Affect Performance? Analysis and Simulation of Timing Using Jitterbug and TrueTime, *IEEE Control Systems Magazine*, 23(3):16-30.

[6] Benítez-Pérez, H. et al. (2012); Bounded Communication Between Nodes in a NCS as a Strategy of Scheduling Approximation, *International Journal on Parallel Emergent and Distributed Systems*, 27(6):481-502.

[7] Fridman, E.; Shaked, U. (2003); Delay-dependent stability and $H\infty$ control: constant and time-varying delays, *International Journal Control*, 76(1):48-60.

[8] Izadi, Z. R.; Blanke, M. (1999); *A Ship Propulsion System as a Benchmark for Fault-Tolerant Control*, Control Engineering Practice, 7:227-239.

[9] Jiang, J.; Zhao, Q. (1999); Reconfigurable Control Based on Imprecise Fault Identification, *Procedings of the American Control Conference, IEEE, San Diego*, 114-118.

[10] Kim, D. et al (2002); *Maximum allowable delay bounds of networked control systems.*Control Engineering Practice, 11:1301-1313.

[11] Lian, F. et al (2001); *Time delay modelling and sample time selection for networked control systems*, Proceedings of ASME-DSC, XX.

[12] Lian, F. et al (2002); Network design considerations for distributed networked for distributed control systems, *IEEE Transactions on Control Systems Technology*, 10(2):297-307.

[13] Liu, C.; Layland, J. (1973); Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment, *Journal of the Association for Computing Machinery*, 20(1):46-61.

[14] Méndez-Monroy P.E.; Benítez-Pérez, H. (2009); Supervisory Fuzzy Control for Networked Control Systems, *International Journal Innovative Computing, Information and Control Express Letters, ICIC-EL*, 233-240.

[15] Menendez, A., Benítez-Pérez, H. (2010); Scheduling strategy for Real-Time Distributed Systems. *Journal of Applied Research and Technology*, 8(2):177-185.

[16] Nilsson, J. (1998); *Real-Time Control with Delays*, PhD. Thesis, Department of Automatic Control, Lund Institute of Technology, Sweden.

[17] Benítez-Pérez, H. et al. (2010); Reconfigurable Fuzzy Takagi Sugeno Model Predictive Control Networked Control (Magnetic Levitation Case Study). Proceedings of the Institution of Mechanical Engineers, *Part I: Journal of Systems and Control Engineering*, 224(8):1022-1032.

[18] Tanaka, K.; Wang, H.O. (2001); *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*, New York, Wiley.

[19] Thompson, H. (2004); *Wireless and Internet Communications Technologies for monitoring and Control*, Control Engineering Practice, 12:781-791.

[20] Walsh, G.C. et al. (1999); Stability Analysis of Networked Control Systems, *Proc. American Control Conference*, San Diego, USA, 2876-2880.

[21] Wang, Y.; Sun, Z. (2007); Control of Networked Control Systems Via LMI Approach, *International Journal of Innovative Computing, Information and Control*, 3(2).

[22] Zhang, H. et al. (2007); Guaranteed cost networked control for T-S fuzzy system switch time delays, *IEEE Transactions on Systems Man and Cybernetics part C: Applications and Reviews*, 37(2):160-172.

[23] Zhu, X. et al.(2008); State Feedback Control Design of Networked Control Systems with Time Delay in the Plant, *International Journal of Innovative Computing, Information and Control*, 4(2):283-290.

[24] Quanser Inc. (2006); *2-DOF Helicopter Reference Manual* Quanser Innovate Educate.

[25] Quanser Inc. (2006); *Magnetic Levitation Experiment*, Quanser Consulting.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):26-38, February 2016.

Fuzzy Linear Physical Programming for Multiple Criteria Decision-Making Under Uncertainty

A. ElSayed, E. Kongar, S.M. Gupta

Ahmed ElSayed
Department of Computer Science and Engineering
University of Bridgeport
221 University Avenue,
Bridgeport, CT 06604, USA
aelsayed@my.bridgeport.edu

Elif Kongar*
Departments of Mechanical Engineering and Technology Management
University of Bridgeport
221 University Avenue,
Bridgeport, CT 06604, USA
*Corresponding author: kongar@bridgeport.edu

Surendra M. Gupta
Department of Mechanical and Industrial Engineering
Northeastern University
360 Huntington Avenue,
Boston, MA 02115, USA
gupta@neu.edu

Abstract: This paper presents a newly developed fuzzy linear physical programming (FLPP) model that allows the decision maker to introduce his/her preferences for multiple criteria decision making in a fuzzy environment. The major contribution of this research is to generalize the current models by accommodating an environment that is conducive to fuzzy problem solving. An example is used to evaluate, compare and discuss the results of the proposed model.

Keywords: Decision Support Systems, Fuzzy Goal Programming, Fuzzy Sets and Systems, Heuristics, Linear Physical Programming, Logistics, Multi-criteria Decision Making, Supply Chain.

1 Introduction

Many real life decisions are often complex and typically involve several parties with different goals and objectives. Today, as also stated by Filip et al. [1], technological and organizational improvements have led to more complex and complicated decision making environments. As a result, current decision models are stretched to the limit to handle these changes as the data availability, number of participants in the decision making process, size of markets and several other factors grow. That is, decision making has become a cumbersome task requiring models that are able to consider various multiple goals, constraints that are somewhat uncertain and flexible in nature.

For instance, the success of supply chains heavily depends on the joint effort focusing on the overall value creation as opposed to optimizing segregated objectives. Furthermore, some of the supply chain outcomes, viz., responsiveness, sustainability, innovation, are less likely to be expressed with crisp objective values [2]. In many cases desired business outcomes are fuzzy in nature and contain a considerable degree of flexibility. The reason for this is twofold. First, the outcome measures are difficult to calculate and quantify. In addition, the conflicting objectives among different parties in the supply chain are likely to mandate varying levels of preference levels even for an identical performance measure. Therefore, it is reasonable to model supply chain performance measurement problems via fuzzy and multiple objective approaches which are, in fact, more suitable for such problem environments.

With this motivation, this paper presents a decision support model which would accommodate the uncertainty involved in the problem environment caused by the flexibility in the goals, information, perspective and/or different visions. The model, based on Linear Physical Programming (LPP), is suitable for individualized decisions as well as decisions where multiple participants are involved and has the ability to integrate and reflect varying preferences of the decision maker.

Linear Physical Programming (LPP) is a preference based optimization method, which operates in the environment of multiple criteria and uses a utility function to represent the decision maker's (DM) preference levels. The key distinguishing feature of LPP is that the decision maker is excluded from the process of choosing appropriate weights [3] which is one of the major challenges when defining a utility function.

Furthermore, similar to other decision support systems [4], the fuzzy LPP formulation can assist the decision maker in overcoming individual limitations and constraint by (i) handling the computational complexity which would otherwise be practically prohibitive and by (ii) introducing vagueness that is naturally embedded in real life decisions or introduced due to multi-participant nature of the decision environment.

Due to these advantages, physical programming has been the topic of various theoretical and practical studies. The technique has been extensively used in various industrial and mechanical engineering applications as well as in other disciplines.

This paper presents a newly developed fuzzy Linear Physical Programming (FLPP) model to accommodate the vagueness of the problem environment while eliminating the need for precision in decision makers' preference levels. The model presented significantly improves the basic methodology and generalizes it to assist in overcoming the decision maker's vagueness.

2 Literature Review

Related body of knowledge offers a large variety of models where the decision maker's preference levels are integrated into an objective function. Most techniques require that the decision maker construct an aggregate objective function using the weights determined as a result of an exasperating trial and error process. In LPP, however, the decision maker specifies the ranges of different degrees of desirability instead of defining the weights. LPP then uses these ranges of desirability to formulate the aggregate objective function, thus eliminating the tedious weight assignment process by providing decision makers with a flexible and more natural problem formulation.

Maria et al. [5] proposed a production-planning model based on Linear Physical Programming (LPP) that would utilize previous design knowledge when available.

Messac et al. [6] applied LPP to reconfigure a distribution network including various quantitative and qualitative factors such as costs of relocation/consolidation, inbound and outbound transportation, relocation, customer delivery time, labor quality, labor-management relations and tax incentives. The proposed model is then utilized to determine the demand allocations of plants to warehouses and warehouses to customers.

Ondemir and Gupta [7] a applied Linear Physical Programming to a multi-criteria advanced repair-to-order and disassembly-to-order (ARTODTO) system for sensor embedded products with RFID tags. The proposed approach aimed at determining how to process EOL products to satisfy life expectancy based demand. Various types of demand are met by a combination of disassembly, repair, and recycling operations. The model allowed third party procurement to prohibit backorders.

A study that introduces heuristics into Physical Programming (PP) was proposed by Sanchis et al. [8]. In this work, the authors replaced PP optimizer by a genetic algorithm to avoid potential local minima issue in addition to simplifying the algorithm that constructs the PP preference functions.

Relevant research combining disassembly sequencing and scheduling via Linear Physical Programming have previously been conducted by Kongar and Gupta [9] and Lambert and Gupta [10].

Another multi-objective method that has been extensively used in problem environments with high uncertainty is fuzzy programming. The method has been widely used due to its ability to respond to the imprecision associated with the input data in multi-objective problems and hence its ability to handle vagueness. In fuzzy programming each objective function value is interpreted as a fuzzy number bounded by the best and worst values of the particular objective. The corresponding fuzzy membership function value of the objective is the deviation from the goal divided by that objective's range, thus a value between 0 and 1. Fuzzy programming aims at finding the solution that minimizes the largest fuzzy membership function. In addition to the goals, other input parameters can be represented by fuzzy numbers and fuzzy arithmetic can be used to combine fuzzy numbers based on fuzzy set theory first introduced by Zadeh [11].

One of the earlier studies in fuzzy was published by Narasimhan [12] who solved a goal programming problem in a fuzzy environment. In order to illustrate the capability of the proposed model, the author provided solutions for the typical goal programming model that aims at a single optimum solution. A similar study has also been proposed by Hannan [13] demonstrating the incorporation of fuzzy goals into typical goal programming formulation via numerical examples. The study also provides a discussion on the advantages and shortcomings of fuzzy and standard goal programming modeling.

Tian et al. [14,15] developed a fuzzy physical programming model for multidisciplinary design optimization. In their study, the authors utilized fuzzy PP for the optimization modeling of through passenger train plan and solved it via genetic algorithm.

Zhang et al. [16] utilized a fuzzy physical programming approach in a linear environment. However, a major drawback in this approach was that it limited the fuzziness only to the boundaries of the problem whereas fuzziness can occur in the variables of both the goals and the classes resulting in more complicated fuzzy relationship between the two.

Another relevant work in the area of fuzzy physical programming (FPP) was published by Huang et al. [17]. In that study, the authors proposed a fuzzy aggregate objective function with fuzzy constraints for each soft and hard class. These studies utilized a single membership function for each goal, and calculated the class function as a function of the membership values as opposed to as a function of the goal values as proposed by the original physical programming methodology. Class functions are considered to be aggregation of goal values and its membership function. The problem is then converted back to its crisp form and solved by genetic algorithm.

Ilgin and Gupta [18] provide a state of the art review of Physical Programming (PP) by classifying the PP related studies into four areas, viz., (1) methodological papers, (2) industrial engineering applications, (3) mechanical engineering applications, and (4) other applications.

The methodology described in this paper combines Linear Physical Programming (LPP) and Fuzzy Sets and proposes a multi-objective model for disassembly sequencing under uncertainty.

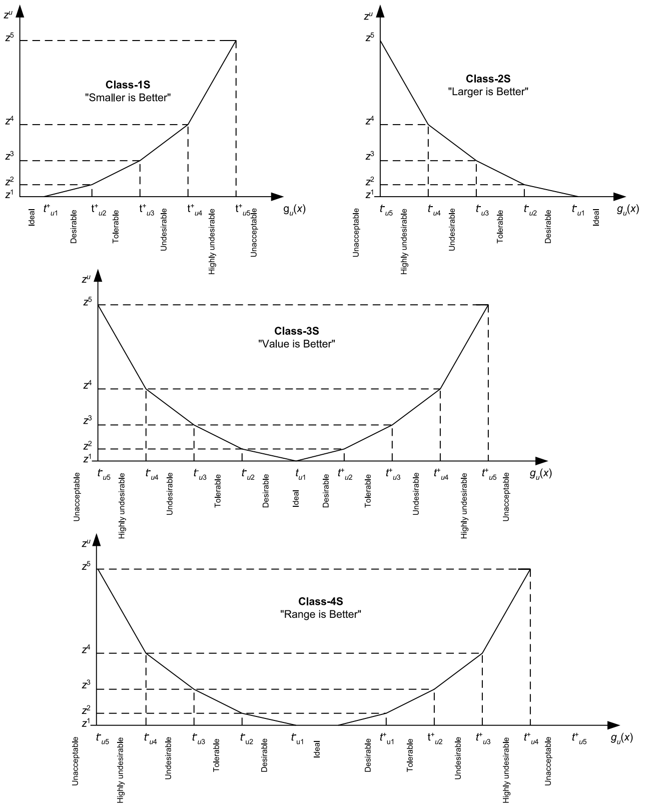


Figure 1: LPP soft class functions for a generic u^{th} objective

3 Theoretical Background and Methodology

3.1 Linear Physical Programming

The physical programming algorithm requires that the decision maker expresses his/her preferences with respect to each criterion using one of the eight different *classes*. The first four classes are *"Soft class functions"*, and represent minimization (Class-1S), maximization (Class-2S), value (Class 3S), and range (Class 4S) optimization. The remaining four are *"Hard class functions"* and are used to introduce inequality and range restrictions into the problem environment. In this regard, Class 1H and Class 2H define upper and lower bounds, respectively, while Class 3H imposes equality and Class 4H imposes range related restrictions to the problem environment. The qualitative and quantitative depiction of each *class* is provided in Figures 1 and 2.

The *soft* class functions allow the DM to express varying levels of preferences for each criterion. This is done by introducing corresponding constraints for each preference level in each of the classes. To provide better understanding, consider Class 1-S and Class 2-S, depicted in Figure 1, which are used for "Smaller is Better" and "Larger is Better" cases respectively. Table 1 demonstrates the ranges and corresponding constraints for the problem. Note that all the *soft* class functions will be embedded in the aggregate objective function to be minimized.

In Figure 1, the u^{th} generic criterion is indicated as $g_u(x)$ where x is the decision variable vector. The goal value, g_u is represented on the horizontal axis while, z_u, the *class function* that is subject to minimization is represented on the vertical axis. Since LPP algorithm considers the lower values of the class functions as "better" values but prohibits negative values, the class function that corresponds to the ideal value is set to zero.

Table 1: Preference levels and constraints for Class-1S and Class 2-S

Class 1-S			Class 2-S		
Range Index	Preference Level	Constraint	Range Index	Preference Level	Constraint
1	Ideal	$g_u \leq t_{u_1}^+$	1	Ideal	$g_u \geq t_{u_1}^-$
2	Desirable	$t_{u_1}^+ \leq g_u \leq t_{u_2}^+$	2	Desirable	$t_{u_1}^- \leq g_u \leq t_{u_2}^-$
3	Tolerable	$t_{u_2}^+ \leq g_u \leq t_{u_3}^+$	3	Tolerable	$t_{u_2}^- \leq g_u \leq t_{u_3}^-$
4	Undesirable	$t_{u_3}^+ \leq g_u \leq t_{u_4}^+$	4	Undesirable	$t_{u_3}^- \leq g_u \leq t_{u_4}^-$
5	Highly Undesirable	$t_{u_4}^+ \leq g_u \leq t_{u_5}^+$	5	Highly Undesirable	$t_{u_4}^- \leq g_u \leq t_{u_5}^-$
6	Unacceptable	$t_{u_5}^+ \leq g_u$	6	Unacceptable	$t_{u_5}^- \geq g_u$

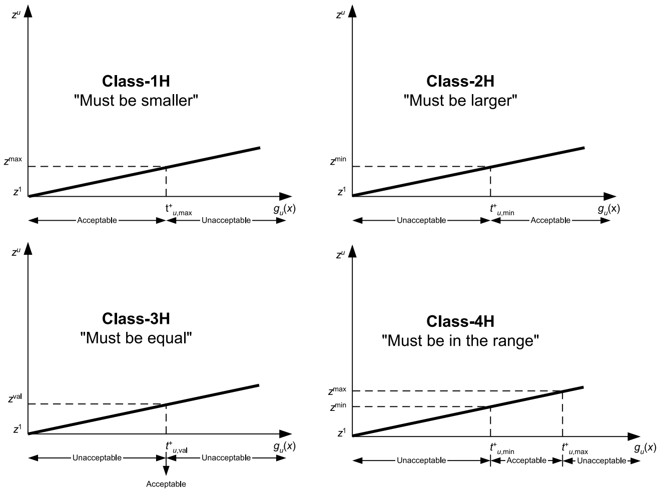


Figure 2: LPP hard class functions for a generic u^{th} objective

As for the hard classes, only two ranges are defined namely, *acceptable* and *unacceptable.* Hard class functions appear as constraints in the LPP model. They can be considered as the hard constraints in Goal Programming and are handled in the same manner. Therefore, they are used for bounding and hence creating the feasible search space (Figure 2).

It is important to note that both soft and hard class functions are piecewise linear and convex. This can be achieved by ensuring that the calculated weights are positive. LPP's built-in-weight calculation algorithm, Linear Physical Programming Weight (LPPW), ensures this condition and guarantees that is each criterion is associated with a piecewise linear class function. Briefly, LPPW, first proposed by Messac [19], uses the decision maker's target values for each criterion to calculate a weight vector. Following this, LPP utilizes these weights to form the objective function to be minimized.

3.2 Fuzzy Linear Physical Programming Model

The methodology proposed in this paper utilizes separate membership functions for each linguistic variable for each goal, and converts the relationship between the class variable and the goals to a fuzzy relationship. Furthermore, the weight for each goal is also calculated based on the fuzzy relationship prior to problem solving via fuzzy goal programming. Therefore, the algorithm detailed in this paper is fuzzy in nature; modeled, initialized and solved in a fuzzy environment throughout.

The concept of creating a fuzzy relationship between the goals and the classes was introduced

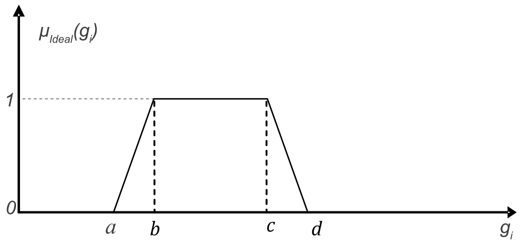


Figure 3: Trapezoidal membership function

by Kosko [20].

The steps of the proposed algorithm are provided below.

1. Select a class type for every goal S1, S2, S3, S4 or H1, H2, H3, H4
2. Decision maker should define the membership function of each linguistic variable for each goal. Assuming that the membership function is trapezoidal, then the function can be expressed mathematically as follows:

$$\mu_{Ideal}\left(g_i\right) = \begin{cases} 0, & \left(g_i < a\right) or \left(g_i > d\right) \\ \frac{g_i - a}{b-a}, & a \leq g_i \leq b \\ 1, & b \leq g_i \leq c \\ \frac{d-g_i}{d-c}, & c \leq g_i \leq d \end{cases} \tag{1}$$

where g_i is the goal number i to be used in the decision process, and, a, b, c, and d are the parameters of the membership function of the Ideal linguistic variable as shown in Figure 3.

3. The class function z^i will also turn into a set of linguistic variables, each member of the set corresponding to the respective linguistic variable for the goal, *i.e.*, Ideal, Desirable, etc., with a trapezoidal assumption the membership function will be as demonstrated follows:

$$\mu_{z^i}\left(z\right) = \begin{cases} 0, & \left(z < a\right) or \left(z > d\right) \\ \frac{z - a}{b-a}, & a \leq z \leq b \\ 1, & b \leq z \leq c \\ \frac{d-z}{d-c}, & c \leq z \leq d \end{cases} \tag{2}$$

For instance $\mu_{z^{Ideal}}\left(z\right)$ value for $\tilde{z}^{Ideal}$ will be as follows (Figure 4):

$$\mu_{z^{Ideal}}\left(z\right) = \begin{cases} 0, & \left(z > c\right) \\ 1, & a \leq z \leq b \\ \frac{c-z}{c-b}, & b \leq z \leq c \end{cases} \tag{3}$$

The values of a, b, c, and d of each membership function will be determined according to the LPPW algorithm to guarantee that the fuzzy relationship between the class function and each goal is a convex fuzzy relationship.

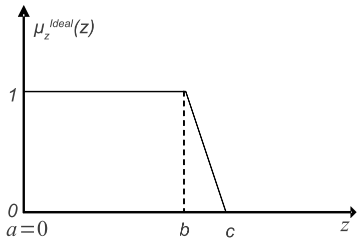


Figure 4: Sample trapezoidal membership function for the Ideal linguistic variable for Class 1-S

4. After determining the values of membership function limits of z_i sets, the fuzzy arithmetic is utilized to obtain the value of the fuzzy weight of each linguistic variable for each goal. Through a fuzzy division, the Ideal set will be calculated as follows:

$$\tilde{w}_{Ideal} = \frac{\underset{\sim}{z}^{Ideal}}{\tilde{g}_{Ideal}} \tag{4}$$

which will generate a membership function for the fuzzy weight.

Please note that equation 4 can easily be solved using a computer program based on the fuzzy division operation described in [21, 22].

5. The weight of each linguistic variable can be obtained by defuzzification of the fuzzy weights. In this paper the center of gravity (centroid) *defuzzification* technique is utilized to defuzzify the weights in order to compute the corresponding crisp values. Following *defuzzification*, each crisp weight is multiplied by the membership function of the corresponding goal's linguistic variable. This way each linguistic goal is shaped according to its priority. The resulting modified membership function becomes:

$$\mu_{Ideal}\left(g_i\right) = \begin{cases} 0, & (g_i < a)\, or\, (g_i > d) \\ \frac{w_{Ideal}(g_i - a)}{b-a}, & a \leq g_i \leq b \\ w_{Ideal}, & b \leq g_i \leq c \\ \frac{w_{Ideal}(d-g_i)}{d-c}, & c \leq g_i \leq d \end{cases} \tag{5}$$

6. Calculate the union of all the membership functions of the linguistic variables of each goal to obtain the final membership function for the corresponding goal values:

$$\mu_{Goal_i}\left(g_i\right) = \mu_{Ideal}\left(g_i\right) \vee \mu_{Desiarble}\left(g_i\right) \vee \mu_{Tolerable}\left(g_i\right) \vee \mu_{Undesirable}\left(g_i\right) \vee \mu_{HUndesirable}\left(g_i\right) \vee \mu_{Unacceptable}\left(g_i\right) \tag{6}$$

7. Define the membership function of the constrains $\mu_{C_i}\left(g_1, g_2, g_3, \ldots\right)$ where C_i is the constraint number i defined by the decision maker (can be considered as one of the Hard classes).

8. Apply Fuzzy Goal Programming by calculating the minimum point that provides the highest value for the membership function of the intersection of every defined membership function :

$$\mu_D\left(g_1, g_2, g_3, \ldots\right) = \min_i \left(\mu_{C_i}\left(g_1, g_2, g_3, \ldots\right)\right) \wedge \min_j \left(\mu_j\left(g_j\right)\right) \tag{7}$$

where $\mu_D(g_1, g_2, g_3, \ldots)$ will be the membership function of the required objectives. Then the optimum goal values will be:

$$G = \min_G (\max(\mu_D(g_1, g_2, g_3, \ldots))) \tag{8}$$

where G is the optimum values of all the goals which satisfy the required constrains.

4 Model Application and Discussion

To demonstrate the applicability of the proposed algorithm, consider the numerical example provided in Messac et al. [3]:

A company manufactures two types of products, product A and product B. Due to factory's physical and human-resource limitations, the company can comfortably manufacture 25 units per month of product A, and 10 of product B. These levels are considered ideal and independent of other considerations. The profits generated per unit from product A and B are respectively \$12k and \$10k. At these levels of manufacturing, the resulting monthly profit becomes \$400k. Investors have decided that if the company is to remain in business, a minimum monthly profit of \$580k must be achieved at the optimal level. After appropriate market research and examination of the situation, the DM decides that, the preferences regarding production rate are as follows:

Preference Level	Product A	Product B
Ideal	<25	<10
Desirable	25-31	10-18
Tolerable	31-36	18-26
Undesirable	36-44	26-33
Highly-Undesirable	44-50	33-40
Unacceptable	>50	>40

The profit (constraint) equation for the factory takes the form: $12g_1 + 10g_2 \geq 580$, where g_1 and g_2 represent two criteria that denote the respective monthly production levels of products A and B.

The following demonstrates the application of the proposed algorithm to the problem described above. Both goals (g_1 and g_2) belong to Class-1S since they are "Smaller Is Better" in nature. Then the membership function of the goals' linguistic variable is assumed as shown in Figure 5 (*i.e.*, '*' for Ideal, '+' for Desirable, '.' for Tolerable, 'o' for Undesirable and '–' for Highly-Undesirable).

Following this assumption, the LPPW algorithm is then employed to calculate the parameters of the class function membership limits to guarantee the convexity of the fuzzy relation between the goals and the class function. Figure 6 shows the resulting membership function for the class function.

Figure 7 shows the fuzzy relation between the class function z and the goal for Product A.

The membership function of the constrains is also assumed as shown in Figure 8,where the constraints are fuzzified so that, for instance, "Less than 25 is Ideal" statement is converted into "Less than 25 is *somewhat* Ideal".

By calculating the weights and applying equation 6, the final membership function for the intersection of all goals become as shown in Figure 9.

The final membership function after applying equation 7 is depicted in Figure 10.

The algorithm results in (g_1 , g_2) = (25.5, 20.2) for Product A and Product B respectively in the case of trapezoid membership functions (shown figures). Product A result (25.5) is in the

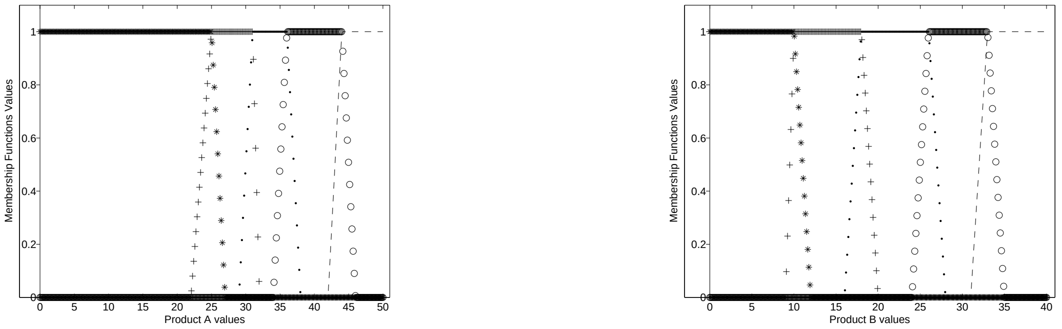


Figure 5: Membership functions of the Product A and Product B linguistic variables

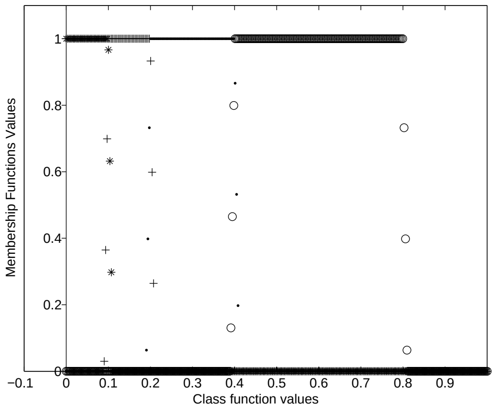


Figure 6: Membership Functions of the Class Function Linguistic Variables

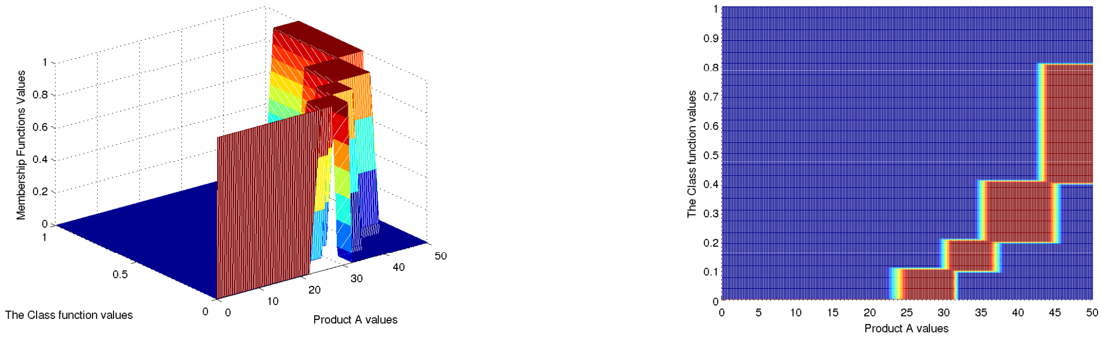


Figure 7: Two Views of the Fuzzy Relation between Product A Values and the Corresponding Class Function

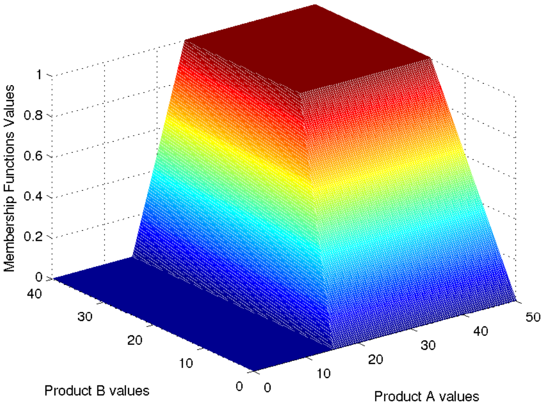


Figure 8: Membership Functions of the Constraints

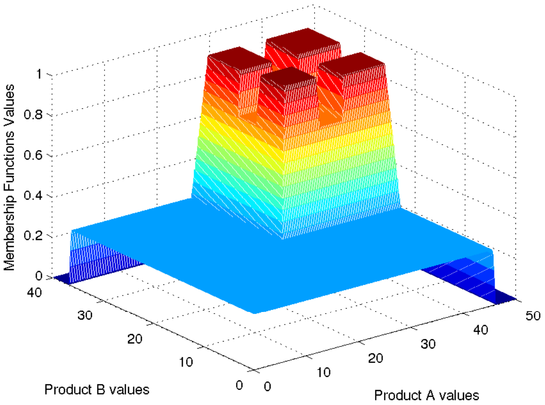


Figure 9: The Intersection Region Between the Membership Functions of Product A and Product B

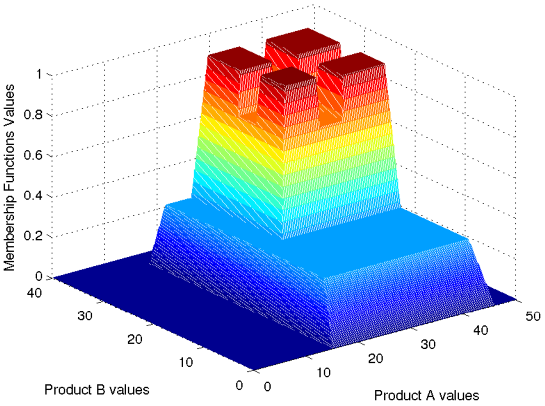


Figure 10: The Intersection Region between the Membership Functions of Product A and Product B and Constrains

Table 2: Number of samples and corresponding execution times

Number of Samples	Execution Time (seconds)
150	1.02
300	2.79
400	4.70
500	7.35
600	10.78

desirable range with close proximity to the *ideal*. As for Product B, the result (20.2) is in the *tolerable* range closer to the *desirable* boundary. When the membership function is defined as triangular, the best goals become (g_1 , g_2) = (28, 22.4) for Product A and Product B, respectively. Here, Product A result (28) is in the *desirable* range, whereas Product B result (22.4) is in the *tolerable* range.

Messac et al. [19] first modeled and solved the problem via Goal Programming (GP) and calculated the (g_1 , g_2) coordinates as (25, 28) and (40, 10). The authors described the solution alternatives as quite restrictive and highly dependent on the choice of weights and on the goals/targets. Furthermore, the authors note that DM's knowledge cannot be implemented in the problem model. Following this, the study proposed the utilization of LPP for the same problem. The resulting value is located at the *desirable/tolerable* boundary for g_1, and within the *tolerable* range for g_2, with (g_1 , g_2) = (31, 20.8).

Compared to both GP and LPP results, the proposed methodology provides better solutions for all goals. In addition, the proposed algorithm eliminates the need for deciding on the weights of goals.

The approach can be easily adapted to general purpose problems as well as specialized problem environments with narrower focus. That is, if a problem can be formulated with an objective function and one or more technological constraints and sign restrictions the presented model will be able to solve the problem in a complete fuzzy environment. The model can also be easily converted into a multiple objective optimization and regardless of the number of objectives in the problem environment, it would still be able to preserve its fuzzy nature. It should also be noted that the only alpha cut used in the computation is in the final step of the optimum value for calculation. Here, the maximum membership function is:

$$\alpha = \max\left(\mu_D\left(g_1, g_2, g_3, \ldots\right)\right),$$

whereas the corresponding optimal value becomes:

$$G = \min\left(A_\alpha\right).$$

.

All the computations in the example are numerical, since they were computed via Von-Neumann architecture computer system and not via a fuzzy processor. For the example shown, the number of samples used for all variables constitutes a domain of 300 samples (*i.e.*, N =300). Increasing the number of samples would adversely affect the overall execution time since calculations and complexity of the fuzzy relations are similar to regular matrix multiplication. Table 2 depicts the relationship between the number of samples and the execution time for a selection of numerical values.

5 Conclusions

Unlike conventional goal programming techniques, fuzzy goal programming, including the proposed fuzzy linear physical programming model, implies a fuzzy system. Therefore, the model's computational complexity on regular digital machines not only depends on the number of input and output variables, but also on the number of discretization levels, viz., samples, from both variables [15]. Using this principle, the computational complexity of the problem under test can be restricted to a number that improves result accuracy while improving the computational time (approximately 300 discretization level for each input and output variable universe of discourse). This simplified fuzzy logic control is proven be suitable for a loop controller with regard to the computation time and the required memory.

Bibliography

[1] Filip, F.G., A.-M. Suduc, M. Bazoi (2014); DSS in numbers. *Technological and Economic Development of Economy*, 20(1): 154-164.

[2] Melnyk, S.A., E.W. Davis, R.E. Spekman, J. Sand (2010); Outcome-Driven Supply Chains, *Sloan Management Review*, 51(2): 33-38.

[3] Messac, A., S.M. Gupta, B. Akbulut (1996); Linear Physical Programming: A New Approach to Multiple Objective Optimization, *Transactions on Operational Research*, 8(2): 39-59.

[4] Filip, F.G. (2008); Decision support and control for large-scale complex systems, *Annual Reviews in Control*, 32(1): 61-70.

[5] Maria, A., C.A. Mattson, A. Ismail-Yahaya, A. Messac (2003); Linear Physical Programming for Production Planning Optimization, *Engineering Optimization*, 35(1): 19-37.

[6] Melachrinoudis, E., A. Messac, H. Min (2005); Consolidating a warehouse network:: A physical programming approach, *International Journal of Production Economics*, 97(1): 1-17.

[7] Ondemir, O. and S.M. Gupta (2014); A multi-criteria decision making model for advanced repair-to-order and disassembly-to-order system, *European Journal of Operational Research*, 233(2): 408-419.

[8] Sanchis, J., M. Martinez, X. Blasco, G. Reynoso-Meza (2010); Modelling preferences in multi-objective engineering design, *Engineering Applications of Artificial Intelligence*, 23(8): 1255-1264.

[9] Kongar, E., S.M. Gupta (2009), Solving the disassembly-to-order problem using linear physical programming, *International Journal of Mathematics in Operational Research*, 1(4): 504-531.

[10] Lambert, A.J.D.; S.M. Gupta (2005), *Disassembly Modeling for Assembly, Maintenance, Reuse, and Recycling*, Boca Raton, FL: CRC Press.

[11] Zadeh, L.A. (1965); Fuzzy Sets, *Inf. Control*, 8: 338-353.

[12] Narasimhan, R. (2008), Goal programming in a fuzzy environment, *Decision Sciences*, 11: 325-338.

[13] Hannan, E.L. (1981), Linear programming with multiple fuzzy goals, *Fuzzy Sets and Systems*, 6(3): 235-248.

[14] Tian, Z., H. Huang, L. Guan (2002); Fuzzy Physical Programming and Its Application in Optimizationof Through Passenger Train Plan, *International Conference on Traffic and Transportation Studies (ICTTS)* Guilin, China: ASCE, 498-503.

[15] Tian, Z., H. Huang, X. Yao, H. Li (2002), Fuzzy physical programming and its application to structural design, *China Mechanical Engineering*, (24): 2131-2133.

[16] Zhang, X., H.-Z. Huang, L. Yu (2006), Fuzzy preference based Interactive Fuzzy Physical Programming and its application in multi-objective optimization, *Journal of Mechanical Science and Technology*, 20(6): 731-737.

[17] Huang, H.-Z., X. Zhang, Z.-G. Tian, C.-S. Liu, Y.-K. Gu (2005), Optimal Design of Conic-Cylindrical Gear Reduction Unit Using Fuzzy Physical Programming, *Intelligent Information Processing II*, Springer: Boston, 191-200.

[18] Ilgin, M.A. and S.M. Gupta (2012), Physical Programming: A Review of the State of the Art, *Studies in Informatics and Control*, 21(4): 349-366.

[19] Messac, A. (1996), Physical Programming: Effective Optimization for Computational Design, *AIAA Journal*, 34(1): 149-158.

[20] Kosko, B. (1994), Fuzzy systems as universal approximators, *IEEE Transactions on Computers*, 1994. 43(11): 1329-1333.

[21] Peeva, K.; Y. Kyosev (2005), Fuzzy Relational Calculus - Theory, Applications and Software: WORLD SCIENTIFIC, 2005 (Advances in Fuzzy Systems - Applications and Theory).

[22] Sumathi, S.; S. Paneerselvam (2010), *Computational Intelligence Paradigms: Theory & Applications Using MATLAB*, Boca Raton, FL, USA. : CRC Press.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):39-50, February 2016.

A Hardware Independent Real-time Ethernet for Motion Control Systems

S. Ji, C. Zhang, T. Hu, K. Wang

Shuai Ji
Mechantronic Engineering School
Shandong Jianzhu University
Jinan, Shandong, China
jizhongzhe@126.com

Chengrui Zhang*
School of Mechanical Engineering
Shandong University,
Jinan, Shandong, China
*Corresponding author: 13969076910@126.com

Tianliang Hu
School of Mechanical Engineering
Shandong University
Jinan, Shandong, China
tlhu@sdu.edu.cn

Ke Wang
Shanghai STEP Electric Corporation
Shanghai, China
ahker534@sina.cn

Abstract: Ethernet for Manufacture Automation Control (EtherMAC) is a new kind of real-time Ethernet used in motion control systems. It adopts a line topology with a standard industrial computer based master node and field-programmable-gate-array based slave nodes. EtherMAC employs one slave node to manage cycle communication and clock synchronization, so the real-time demand for its master node can be greatly reduced and dedicated hardware is no longer mandatory. Its distributed clock compensation mechanism can get synchronization accuracy in nanosecond order. The advantages of industrial computer and field programmable-gate-array are combined with EtherMAC, so that high control performance can be achieved.

Keywords: Real-time Ethernet, cycle time, synchronization, motion control, FPGA

1 Introduction

Fieldbus has played an important role in factory automation field during the past 20 years [1]. However, it can hardly meet the development of the networked control systems any longer because of the numerous standards, low baud rate, high cost and etc [12]. At the same time, as a mature technology in office, Ethernet is introduced into field control layer for its low price, high communication rate, robustness and easy deployment. Many automation vendors begin to propose their own industrial communication solutions by modifying the original Ethernet protocol to improve its real-time performance (see [4, 13]). These types of Ethernet based on IEEE 802.3 standards and with real-time property are called real-time Ethernet (RT Ethernet) [5]. Standardization of industrial ethernet Standardization of industrial ethernet

IEC61784-2 has defined some indicators to specify the real-time performance of RT Ethernet, such as delivery time and synchronization accuracy. But the minimum cycle is a more direct indicator to evaluate the performance of motion control system than delivery time(see [6]- [9]). The RT Ethernets whose cycle time can be smaller than 1ms with jitters less than 1ms are called isochronous RT Ethernet (see [10]- [11]), which are most likely to be used in motion control systems.

However, most of the current isochronous RT Ethernet solutions are based on modifications of the hardware [12]. Dedicated Network Interface Card (NIC) or real-time OS are mandatory for the master node in time critical applications. For example, PROFINET IRT is achieved with a special switch ASIC (see [12]); dedicated NICs are mandatory for SERCOS III and Ethernet Powerlink's master/management node in time-critical applications (see [13]); EtherCAT does not require a dedicated master node, but a hard real-time kernel is mandatory for the distributed clock synchronization, in addition the operating system can introduce jitter to the cycle time(see [14]). The dedicated hardware is closed and high cost, which is not good for the development of motion control system.

Herein, a standard hardware based real-time Ethernet named EtherMAC is proposed. EtherMAC employs the programmable gate array chips (FPGA) based slave node to manage the communication and synchronization, so that IPC master node can focus on motion control algorithms. In this way, computation capability of IPC and hard real-time property of FPGA are combined, and hard real-time performance can be achieved even the master node without dedicated NICs and hard real-time operating systems.

The rest of this paper is organized as follow. In Section 2, the protocol model of EtherMAC is introduced. In Section 3 and 4, its real-time and synchronization mechanism are described in detail. Section 5 is the slave node implementation on a FPGA chip. Then two experiments are carried out in Section 6 to test the delivery time and synchronization precision. Finally, the conclusion is given in Section 7.

2 Overview of EtherMAC

2.1 Topology of EtherMAC

Line topology network is widely used in industrial automation field for its simple structure, low cost and advantage in deterministic communication(see [11], [15]- [16]Jasperneite A Proposal for a Generic Real-TimeJasperneite A Proposal for a Generic Real-TimeJasperneite A Proposal for a Generic Real-Time). EtherMAC also takes the line topology as its fundamental topology, where the IPC with at least one standard NIC works as the master node, and the FPGA-based slave nodes with two Ethernet ports cascade with each other in a line as shown in Figure 1. All the slave nodes can be treated as a single standard Ethernet device for the master node, because they share a common Media Access Control (MAC) address.

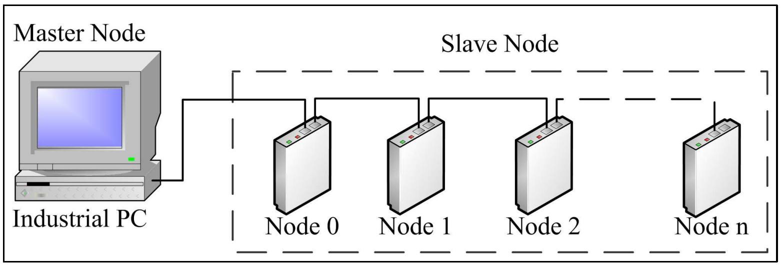


Figure 1: Topology of EtherMAC

EtherMAC employs summation frames in Figure 2 to convey application data, which is similar to INTERBUS [17] and EtherCAT [18]. The configuration and control data are conveyed down by a summation frame in the descending way, and the state data of the slave nodes are collected by another summation frame in the ascending way. In this manner, all the nodes can be updated in a single communication cycle. This is more efficient than the other RT Ethernets employing individual frame(see [8], [11]). In addition, the control and state data are separated in the descending and ascending ways, so bandwidth of Ethernet can be used more efficiently.

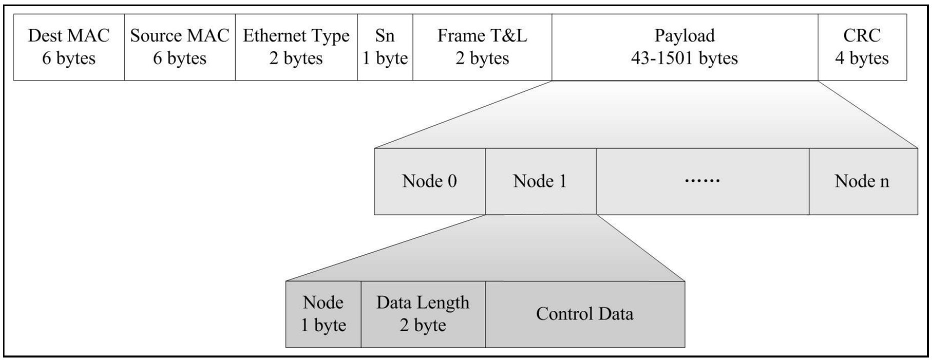


Figure 2: Frame structure of EtherMAC

2.2 Model of EtherMAC

There are three solutions for Ethernet to get real-time communication. The first class is named "on top of TCP/IP", which keeps the TCP/UDP/IP protocols unchanged and concentrates all real-time modifications in the application layer. In the second solution, application layer programs bypass the TCP/UDP/IP protocols and operate the Ethernet directly. In the third one, medium access mechanism and hardware structure of Ethernet are modified to achieve isochronous real-time performance (see [8], [12]). The architecture of EtherMAC is shown in Figure 3. The master node is implemented on a standard IPC, and TCP/IP protocols are bypassed. The slave nodes are implemented on FPGA chips with modified Data Link Layer (DLL).

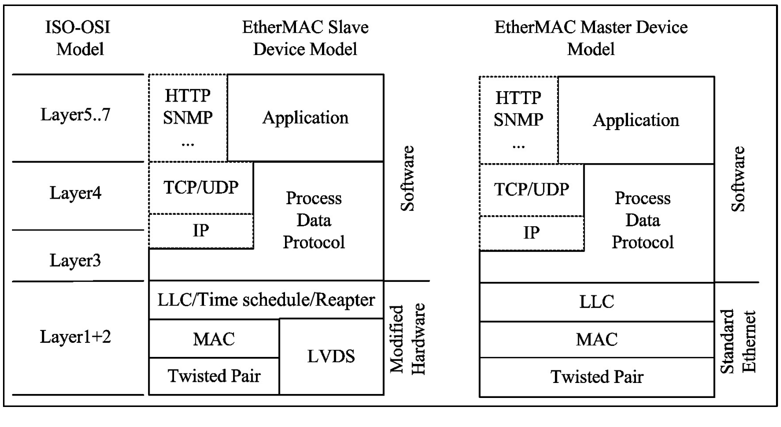


Figure 3: Protocol Suite

The modified DLL of the slave node has the following characters:

- A time schedule module is added to DLL to manage the communication cycle and synchronize the distributed clocks.
- A repeater is designed to receive and forward the summation frame on line, so that small forward delay can be achieved.

3 Real-time communication

The master-slave structure is most used in RT Ethernet, where the master node sends out the control data periodically and the slave nodes carry out the received commands. The cycle duration accuracy is determined by the master node in this structure, as a result dedicated NIC hard real-time or operating system is necessary for the master node in time critical applications. However, on one hand dedicated NIC solution is closed architecture and high cost, on the other hand the response time jitter of real-time operating system can reduce the cycle accuracy.

EtherMAC turns its time-critical tasks over to the FPGA-based slave nodes, while master node is designed to complete the configuration and complex control algorithms. The slave nodes of EtherMAC are implemented on FPGA chips, which can provide high timing precision because of their hardware property.

3.1 Communication process

The communication process of EtherMAC is shown in Figure 4. The slave nodes in the networks carry out a self-checking program after powered on. If there are no problems with the hardware, the slave nodes will get into the idle state and wait for commands from master node.

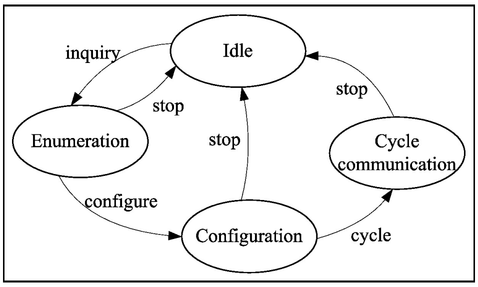


Figure 4: State machine of EtherMAC working process

1)Enumeration. The slave nodes are assigned a unique sequence number and feedback their inherent information in this stage. The detailed process is shown in Figure 5.

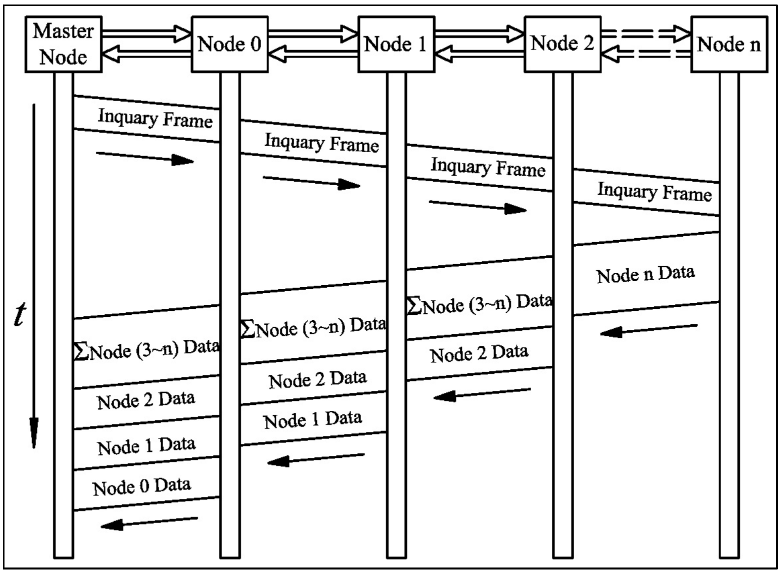


Figure 5: Enumeration Process

First, master node sends an inquiry frame to the slave nodes. Every slave node takes the data in certain place of the frame as its sequence number, increases its value by one and forwards the frame the next. The slave nodes get their sequence number in this way.

Then, the last slave node stops the forwarding process and starts to feedback after receiving the inquiry frame. It packets its inherent information into the feedback frame and sends the frame to the second last slave node. All the previous slave nodes attach their inherent information to the end of the feedback frame and forward it.

At last, the frame with the information of all slave nodes is transmitted to the master node by slave node 0. The master node compares the received information with the network configuration file to check the configuration state of the network.

Figure 5 The Inquiry process

2)Configuration. The master node starts the configuration stage if all the slave nodes in the network have been checked well in enumeration stage. Master node delivered the configuration frame to slave nodes, and slave nodes return their configuration results. Then the master node prepares the configuration data of the next cycle based on the feedback information. The forward delays measurement is also carried out in this stage, which will be introduced in the Section 4.

3)Cycle communication. The master node sends out a frame to start the cycle communication stage after the network configure done. The management right of the communication is transferred to the last slave node from then on. Different from the last two stages, the communication is started by the last slave node instead of the master node in this stage. The detailed process is shown in Figure 6.

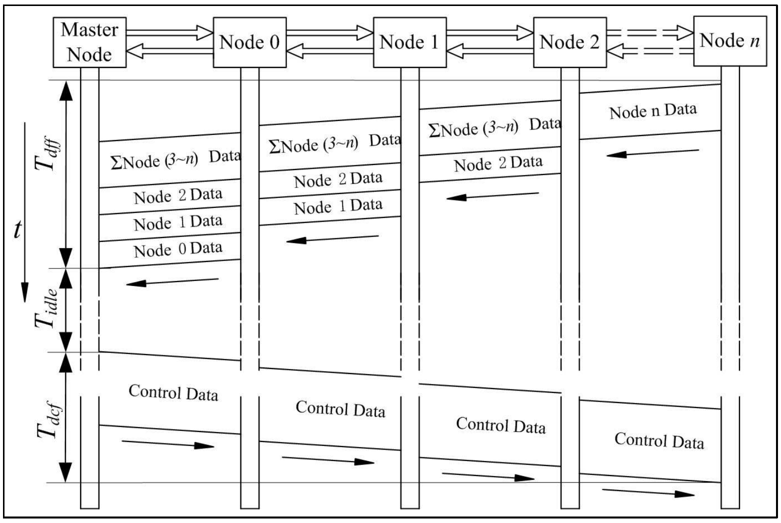


Figure 6: Cycle Communication Process

The last slave node starts a new communication cycle by sending a frame back with the set communication period. The slave nodes update their state data to master node by attaching their data to the end of the feedback frame. As soon as receives the feedback frame, master node computes the commands of the next period and sends it to slave nodes. Therefore, real-time communication can be achieved if the master node can finish all the operations before the last slave node starting the next communication cycle.

In cycle communication, the control frames are forwarded directly and without any modification in the descending way, and the feedback frames are processed online in ascending way, so that small forward delay can be achieved. This means small delivery time for line topology network. For one slave node, the total forward delay in a communication cycle is about 520ns.

3.2 Performance analysis

The delivery time of EtherMAC can be derived from its communication mechanism and the theoretical analysis in(see [6]- [8])

$$T_d = T_{frame} + T_{cable} + \sum_{i=0}^{n-1} \Delta T(i)_{delay} \tag{1}$$

where T_{cable} is propagation delay from cable, which is about 5ns/m for the category 5 cable. T_{frame} is the frame transmission time, which is in linear with the frame size. $\Delta T(i)_{delay}$ is the frame forward delay of slave node i, which is about 520ns with proposed the forward mechanism.

There are two different types of Ethernet frames in the cycle communication stages, so the minimal cycle time T_{cycmin} can be derived

$$T_{cycmin}h = T_{dcf} + T_{dff} + T_{ifg} + T_{idle} \tag{2}$$

where T_{dcf} is the delivery time of control frame, T_{dff} is the delivery time feedback frame, T_{ifg} is the inter frame gap, because the full duplex communication is employed there is only one T_{ifg}. T_{idle} is the idle time when there is no communication in the network.

In this way, the react time jitters of master node does not influence cycle time accuracy any more, and isochronous real-time communication can be achieved even the master node only with soft real-time property. The minimal cycle time can be as small as $20\mu s$ when there are only 3 slave nodes and payload is no more than 48 bytes length.

4 Synchronization of distributed slave nodes

Synchronization accuracy of RT Ethernet is its capability to synchronize the actions of the distributed devices [19]. RT Ethernet demands the synchronized actions must be repeated periodically with strict jitter bounds [3]. Poor synchronization of relevant axes in motion control systems means diminished dimensional accuracy of the work-piece or even unusable products (see [20]- [21]). Many synchronization methods have been proposed for networked control systems(see [22]- [24]), of which Precision Time Protocol (PTP) described in IEEE 1588 is one of the most used. However, PTP is difficult to implement, which consumes too many hardware and bandwidth resources in practice. Therefore, a concise synchronization algorithm is proposed to get high synchronization accuracy.

4.1 Factors result to asynchronization

Every slave nodes in the network has its own oscillator. The frequency of each oscillator cannot be exactly the same, besides the clock frequency drifts with environmental temperature and time. As a result, synchronization mechanisms are necessary for networked control systems. Broadcast frame is often used as the synchronization signal in star topology networks, but it can hardly work in line topology networks thanks to the forward delay between cascaded slave nodes. Equations in (4.1) show the time difference of the slave nodes.

$$\begin{aligned} T_1 &= T_0 + \Delta T^1_{delay} \\ T_2 &= T_1 + \Delta T^2_{delay} = T_0 + \Delta T^1_{delay} + \Delta T^2_{delay} \\ &\cdots\cdots \\ T_{n-1} &= T_0 + \sum_{i=0}^{n-1} \Delta T(i)_{delay} \end{aligned} \tag{3}$$

where T_i is the time a frame arriving at the slave node i in descending way. T^i_{delay} is the forward delay of salve node i, which accumulates with the node number and cable length. The forward delay of every slave node is not constant because of the phase separation of receiving and transmitting clock. Thus, it is impossible to get the exact value beforehand and compensate it in cycle communication.

4.2 Synchronization mechanism of EtherMAC

The distributed clocks synchronization can be divided into two stages: delay measurement and clock compensation.

Delay measurement is only carried out once at the first time of network configuration. The detailed process is shown in Figure 7. Master node sends the measure frame cyclically, and the last slave node starts the feedback process as soon as receives the frame. Every slave node records the time between it receives the measure frame and feedback frame. The measurement begins at the time of the frame arriving at the media independent interface, so the influence of software stack can be ignored. The ascending and descending way can be symmetrical by compensating the forward FIFO depth. Here the measured delay includes the slave node forward delay and cable delay. Take the phase separation of the transmitting and receiving clock into account, the average value of hundreds of record delay time is used as delay time.

In the cycle communication stage, every salve node initializes its local clock with half of the average delay time at the time of receiving the feedback frame, so that the slave nodes can share a common zero point. In this way, the distributed clock of the slave nodes can be synchronized.

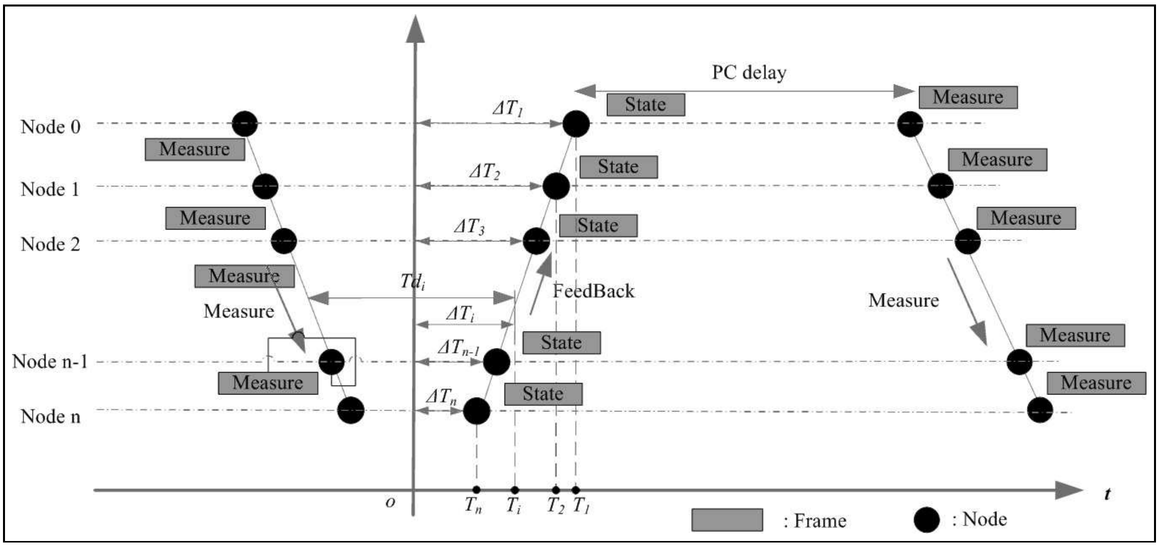


Figure 7: Synchronization Process

The forward delay and clock drift can be compensated cyclically because the feedback process is carried out every communication cycle, so high synchronization accuracy between slave nodes can be achieved in this way.

5 Implementation of slave node

Master node of EtherMAC can be implemented on any controller with standard Ethernet interfaces, and no dedicated hardware in needed. Real-time OS is necessary for some time critical applications, but the real-time demand can be greatly reduced with the above mentioned communication mechanism. Therefore, only the slave node implementation is introduced in this paper. The slave nodes are implemented on FPFA chips as shown in Figure 8.

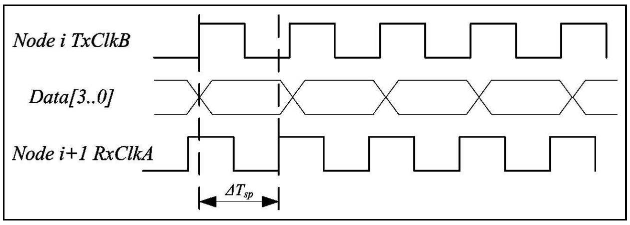


Figure 8: Implementation of EtherMAC slave node

FPGA is more than some programmable resource with a certain number of logical cells today. The loadable soft cores enable the user to create a microcontroller within an FPGA chip and equip it with ultra fast peripherals [25]. It allows the developers to customize peripherals to meet the specific requirement.

All the functions of the EtherMAC are packaged into an IP core. The application layer is implemented as software in Nios II a soft processor designed specifically for Altera FPGA chips. IO, Servo Control and the other peripherals are mounted on the Avalon bus. A Static Random Access Memory (SRAM) chip is used to running application programs, and a flash chip is used to store the configuration data of FPGA. The schematic diagram of EtherMAC IP core is shown in Figure 9.

There are two concurrent channels in descending way. One is for protocol analysis, and the other one is for frame forward. In protocol analysis channel, application data belong to the current slave node are extracted and stored in a dual-port-RAM AppCtrl Data, and the operation information including data address and length are stored in a dual-port-RAM Operation Info at the same time. After receiving the whole frame, the signal RxDone holds high for twelve system clocks to inform the upper layer program that application data have been updated. Then the application programs can get the commands of this cycle from AppCtrl Data.

The Syn signal holds high for twelve system clocks when the distributed clocks time to the set point. Upper layer programs and other peripherals modules can carry out the synchronous actions and latch the state of slave node at this moment. The state data have been latched can be upload to the master by writing them into the dual-port-RAM AppState Data.

6 Experiment

The system in Figure 10 is employed to evaluate the real-time performance of EtherMAC. The system has one master node and eleven slave nodes. The master node is an IPC with an Intel Atom D510 1.66GHz CPU and Windows 7 OS. The Network Driver Interface Specification (NDIS) layer of the system is modified to improve its real-time performance [26]. The slave nodes are connected with a backplane, which hides in the steel rail. 100BASE-T Ethernet cable is adopted when the adjacent nodes are far away from each other as in Figure 10.

Two experiments are carried out to measure the delivery time and synchronization accuracy respectively. In the first experiment, delay time Tdi in Figure 7 is measured and delivered to master. Here, the time duration between slave node 0 receiving the measurement and feedback frame is twice of the delivery time, while half of the time difference between adjacent nodes is the forward delay of a slave node. As shown in Figure 11, the delay time is linear with the sequence number of slave nodes and the forward delay of a single slave node is about 520ns.

In the second experiment, an oscilloscope is used to watch the synchronization signals. For convenience, the synchronization signals are assigned as output pins of the FPGA chips. Slave

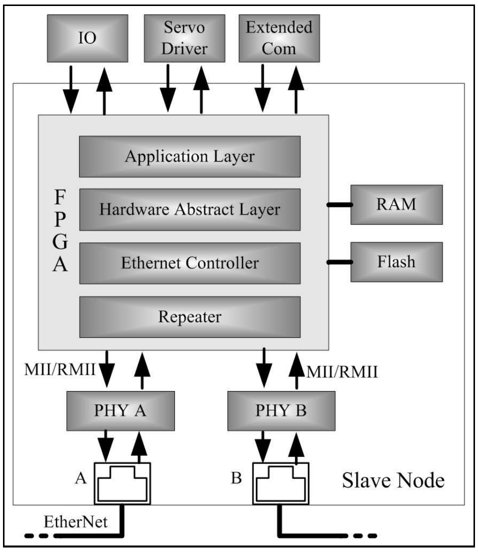


Figure 9: Hardware Architecture of EtherMAC

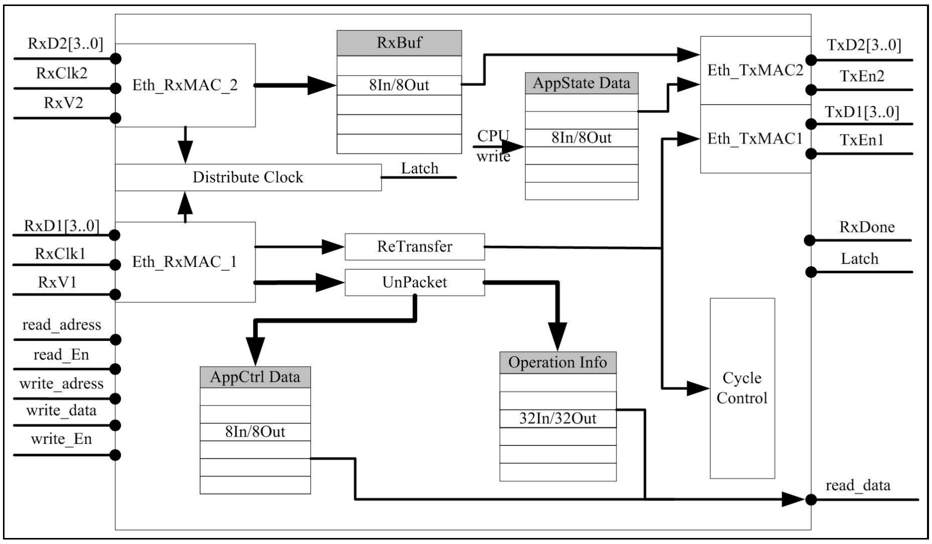


Figure 10: Distributed slave nodes

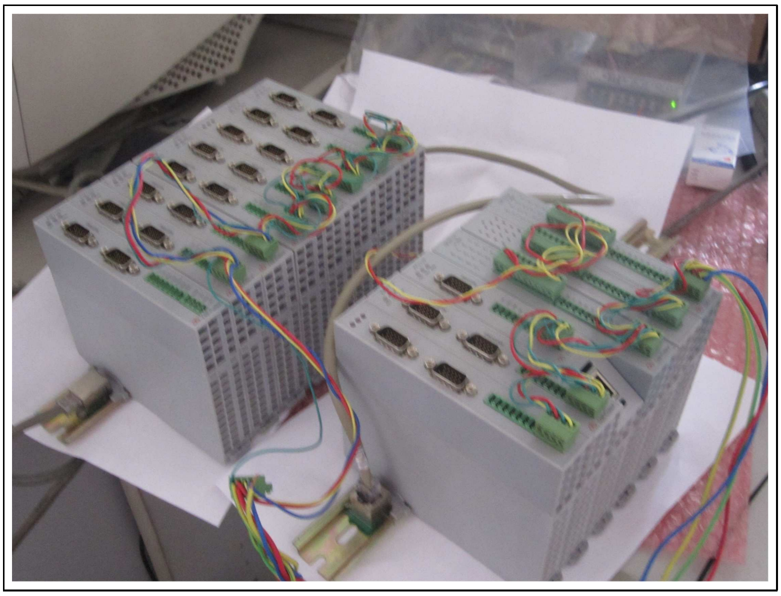


Figure 11: Delay time and jitters of Salve nodes

node 0, 5 and 10 are tested, of which slave node 0 works as the reference signal. All the triggered wave forms can superpose and stay on the screen by setting the oscilloscope works in infinite persistence mode. The results after the system worked for 30 minutes are shown in Figure 12. It is shown that the deviation of these slave nodesĄŻ synchronization signal is about 100ns with jitters no more than 100ns.

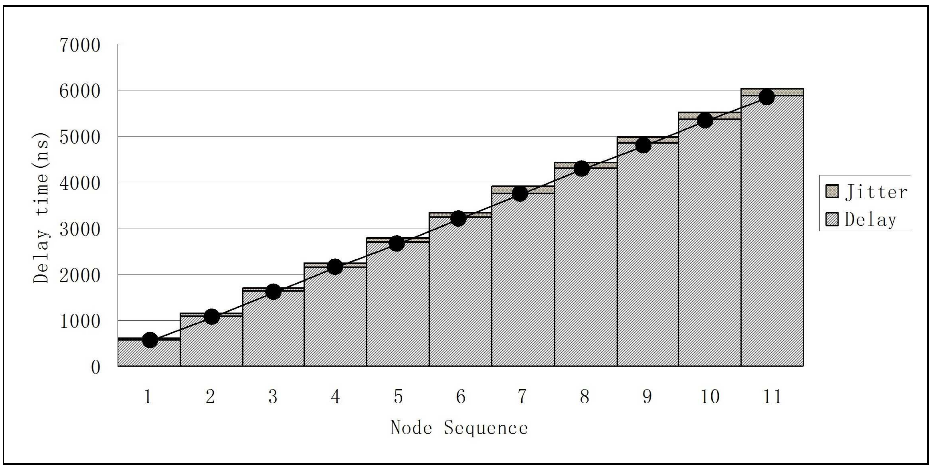


Figure 12: Jitter of the synchronization signal

7 Conclusion

RT Ethernet can help the motion control systems to get a concise architecture and good control performance [27]. However, many RT Ethernet solutions need dedicated NIC to manage communication in time critical applications. This goes against the open architecture motion control systems. Therefore, the hardware independent based EtherMAC is proposed in this paper. As most of the time critical tasks of EtherMAC are transferred to the FPGA based slave nodes, master node can focus on the implementation of motion control algorithms. In this way, a software based motion control system with open architecture and high performance can be achieved.

Bibliography

[1] M. P. Thomesse(2005); Fieldbus technology in industrial automation, *Proceedings of the IEEE*, ISSN 0018-9219, 93(6):1073-1101.

[2] M. Felser and T. Sauter(2004); Standardization of industrial ethernet - The next battlefield?, *IEEE International Workshop on Factory Communication Systems, Proceedings, 2004*, ISBN 0-7803-8734-1, 413-420.

[3] J. D. Decotignie(2005); Ethernet-based real-time and industrial communications, *Proceedings of the IEEE*, ISSN 0018-9219, 93(6): 1102-1117.

[4] P. Neumann(2007); Communication in industrial automation's what is going on? *Control Engineering Practice*, ISSN 0967-0661, 15(11): 1332-1347.

[5] Digital Data Communications for Measurement and Control, Part 2: Additional Profiles for ISO/IEC 8802-3 Based Communication Networks in Real-Time Applications, *IEC 61784-2, 65C/350/CD circulated for comments.*

[6] L. Seno, S. Vitturi, L. Peretti, M. Zigliotto, and C. Zunino (2011); Real-time Ethernet networks for motion control, *Computer Standards & Interfaces*, ISSN 0920-5489, 33(5): 465-476.

[7] L. Seno, S. Vitturi, and C. Zunino(2009); Real time Ethernet networks evaluation using performance indicators, *Emerging Technologies & Factory Automation, 2009. ETFA 2009. IEEE Conference on*, ISSN 1946-0759, 1-8.

[8] J. Jaspemeite, M. Schumacher(and K. Weber 2007); Limits of increasing the performance of industrial Ethernet protocols, *12th IEEE International Conference on Emerging Technologies and Factory Automation*, ISBN 978-1-4244-0825-2, 17-24.

[9] J. Robert, J.-P. Georges, E. Rondeau, and T. Divoux(2012); Minimum cycle time analysis of Ethernet-based real-time protocols, *International Journal of Computers, Communications and Control*, ISSN 1841-9836, 7(4): 743-757.

[10] Performance analysis of industrial ethernet networks by means of timed model-checking D Witsch, , B Vogel-Heuser, JM Faure(2006); Performance analysis of industrial ethernet networks by means of timed model-checking, *Information control problems in manufacturing 2006*, ISBN 978-0-08-044654-7, 99 C104.

[11] M. Schumacher, J. Jasperneite, and K. Weber(2008); A new approach for increasing the performance of the industrial Ethernet system PROFINET, *IEEE International Workshop on Factory Communication Systems*, ISBN 978-1-4244-2349-1, 159-167.

[12] M. Felser(2005); Real-time ethernet - Industry prospective, *Proceedings of the IEEE*, ISSN 0018-9219, 93(6): 1118-1129.

[13] J. D. Decotignie(2009); The many faces of industrial ethernet [past and present], *Industrial Electronics Magazine*, ISSN 1932-4529, 3(1): 8-19.

[14] S. Potra and G. Sebestyen(2006); EtherCAT Protocol Implementation Issues on an Embedded Linux Platform, *2006 IEEE International Conference on Automation, Quality and Testing, Robotics*, ISBN 1-4244-0360-X, 420-425.

[15] S. Ruping, E. Vonnahme, and J. Jasperneite(1999); Analysis of Switched Ethernet networks with different topologies used in automation systems, *Fieldbus Technology: Systems Integration, Networking, and Engineering*: ISBN 978-3-211-83394-0, 9:351-358.

[16] J. Jasperneite, J. Imtiaz, M. Schumacher, and K. Weber(2009); A Proposal for a Generic Real-Time Ethernet System, *IEEE Transactions on Industrial Informatics*, ISSN 1551-3203, 5(2): 75-85.

[17] H. Li, H. Zhang, and D. Peng(2009); Research and application on INTERBUS operator terminal, *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, ISBN 978-1-4244-4519-6, 309-312.

[18] D. Jansen and H. Buttner(2004); Real-time Ethernet: the EtherCAT solution, *Computing and Control Engineering*, ISSN 0956-3385, 15(1): 16-21.

[19] P. Ferrari, A. Flammini, S. Rinaldi, and G. Gaderer(2008); Evaluation of clock synchronization accuracy of coexistent Real-Time Ethernet protocols, *2008 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication*, ISBN 978-1-4244-2274-6, 87 - 91.

[20] D. Sun(2003); Position synchronization of multiple motion axes with adaptive coupling control, *Automatica*, ISSN 0005-1098, 39(6): 997-1005.

[21] M. Tomizuka, J.-S. Hu, T.-C. Chiu, and T. Kamano(1992); Synchronization of Two Motion Control Axes Under Adaptive Feedforward Control, *Journal of Dynamic Systems, Measurement, and Control*, ISSN 1528-9028, 114(2): 196-203.

[22] X. L. Zhang, X. Q. Tang, and J. H. Chen(2008); Time synchronization of hierarchical real-time networked CNC system based on ethernet/internet, *The International Journal of Advanced Manufacturing Technology*, ISSN 0268-3768, 36(11-12):1145-1156.

[23] X. Xu, Z. Xiong, J. Wu, and X. Zhu(2013); High-precision time synchronization in real-time Ethernet-based CNC systems, *The International Journal of Advanced Manufacturing Technology*, ISSN 0268-3768, 65(5-8):1157-1170.

[24] L. Zhang, Z. Liu, and C. Honghui Xia(2002); Clock synchronization algorithms for network measurements, in INFOCOM 2002. *Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, ISSN 0743-166X, 1(1):160-169.

[25] B. S. Michael Samuelian(2007); A Universal Approach for implementing Real-Time Industrial Ethernet. *White Paper: IXXAT and Altera* [Online].

[26] K. Wang, C. Zhang, X. Xu, S. Ji, and L. Yang(2012); A CNC system based on real-time Ethernet and Windows NT, *The International Journal of Advanced Manufacturing Technology*, ISSN 0268-3768, 65(9-12):1383-1395.

[27] G. Morel, P. Valckenaers, J.-M. Faure, C. E. Pereira, and C. Diedrich(2007); Manufacturing plant control challenges and issues, *Control Engineering Practice*, ISSN 0967-0661, 15(11): 1321-1331.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):51-66, February 2016.

Personnel Ranking and Selection Problem Solution by Application of KEMIRA Method

N. Kosareva, E.K. Zavadskas, A. Krylovas, S. Dadelo

Natalja Kosareva, Edmundas Kazimieras Zavadskas*,
Aleksandras Krylovas, Stanislav Dadelo
Vilnius Gediminas Technical University
Sauletekio al. 11, LT-10223 Vilnius, Lithuania
natalja.kosareva@vgtu.lt, edmundas.zavadskas@vgtu.lt
aleksandras.krylovas@vgtu.lt, stanislav.dadelo@vgtu.lt
*Corresponding author: edmundas.zavadskas@vgtu.lt

Abstract: In this study KEmeny Median Indicator Rank Accordance (KEMIRA) method is applied for solving personnel ranking and selection problem when there are two subgroups of evaluating criteria. Each stage of KEMIRA method illustrated with the examples. In the first stage Kemeny median method is applied to generalize experts' opinions for setting criteria priorities. Medians were calculated for all experts opinions generalization and for experts majority opinions generalization. In the second stage criteria weights calculated and alternatives ranking accomplished simultaneously by Indicator Rank Accordance method. The obtained solutions compared with the results received in previous work of authors.

Keywords: multiple criteria decision making, Kemeny median, criteria priority, optimization problem.

1 Introduction

Personnel ranking and selection process in companies is focused on testing and evaluating of human resource potential, skills and personal characteristics. Developed personnel selection and evaluation systems are usually oriented to companies operating cost reduction in order to optimize human resources demand and layout planning [8]. Personnel selection and placement of the necessary positions is seen as the most important factor affecting organization's security, stability and development [20]. Proper recruitment has influence on organization's climate directly or influencing it through mediators [34]. Security personnel selection process requires identification of specific criteria for the occupied position and setting of their weights values [12]. Solution of this problem is relevant to many companies and organizations. At the same time it is necessary to create unified (being in accordance with uniform standards) personnel selection algorithms [3]. Personnel selection deployment process is directed to human resource potential, skills and personal features, i.e. factors affecting professional efficiency, testing and evaluation. Research has revealed that the main factors affecting the professional ability to work are of different origin: physical, psychological, cognitive, social/behavioral, workplace factors and factors outside the workplace [18]. This investigation has not determined the strength (weight) of identified factors influence on professional working capacity. It can be assumed that these factors have a different impact on the effectiveness of different professional activities. Experts' assessment applied for weight identification is often a subjective process based on stereotypes, attitudes, sympathy and so on [15]. The results of such evaluation often do not meet the expectations of the ultimate goal. From the other hand application of specialized tests (objective assessments) for weight identification meets the uncertainty factor. It is not clear how much test results will be concerned with the quality of the work results in the future [22]. According to the authors the most reliable methodology of professional selection is integrating several fields of criteria and sub-criteria systems [41]. It was found that data obtained with the objective (internal) and subjective (external) methods correlate with each other at about 0.3 [11], so harmonization of objective and subjective criteria schemes allow to expect a higher efficiency of the staff selection process [13].

Solution of this relevant problem is often subject to the Multiple Criteria Decision Making methods (MCDM). MCDM is rapidly evolving methodologies direction, does have considerable researchers attention. Decision-making methods face new requirements for modernization. There is a variety of MCDM methods. A monograph [39] provides readers with a capsule look into the existing methods, their characteristics and applicability to analysis of Multiple Attribute Decision Making (MADM) problems. Models for MADM, transformation of attributes, fuzzy decision rules, and methods for assessing weights are presented in this work. The paper [42] presents a panorama of decision making methods in economics and summarizes the most important results and applications which have appeared in the last decade. [33] added several new important concepts and trends in the MCDM field for solving actual problems. They provided comments on a previously published paper of [42] that could be thought of as an attempt to complete the original paper. Despite the intensive development worldwide, few attempts have been made to systematically present the theoretical bases and developments of MCDM methods. The article [43] describes the situation with reviews of MCDM/MADM methods. The goal of the paper [31] is to propose an approach to resolve disagreements among MCDM methods based on Spearman's rank correlation coefficient. The authors showed that the proposed approach can resolve conflicting MCDM rankings and reach an agreement among different MCDM methods. Sometimes, there are situations where MCDM methods generate very different results. The results proposed in [35] proved that MCDM methods are useful tools for evaluating multiclass classification algorithms and the fusion approach of MCDM methods is capable of identifying a compromised solution when different MCDM methods generate conflicting rankings.

However, MCDM techniques for recruitment and layout planning areas are not consistently structured and improved, this is an obstacle to their wide application. New MCDM methods are developing and new fields are looking for their application. The fuzzy MULTIMOORA for group decision making (MULTIMOORA-FG) enables to aggregate subjective assessments of the decision-makers and offer an opportunity to perform more robust personnel selection procedures [4], [5]. The selection process often contains imprecise data and linguistic variables. The model proposed in [6] combines fuzzy analytic hierarchy process (FAHP) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) approaches to overcome these problems. The personnel selection problem is suitable to be dealt with by the linguistic VIKOR method because it includes some conflicting criteria and needs to consider the relative competitiveness of each candidate [9], [16]. The hybrid MCDM models which employ analytic network process (ANP) and modified TOPSIS and combination of the fuzzy Delphi method, ANP, and TOPSIS demonstrated the effectiveness and feasibility of the proposed models for personnel selection [14] and for investment in stock exchange [19]. The model based on fuzzy DEMATEL (Decision Making Trial and Evaluation Laboratory) and fuzzy ANP is proposed in [25] to to cope with the interdependencies between evaluation criteria. In the mentioned paper, a fuzzy DEMATEL-ANP model is proposed for selection of snipers. An approach enabling combination of Fuzzy ANP, Fuzzy TOPSIS and Fuzzy ELECTRE techniques were enabled for solving of the same problem [26]. Other authors proposed other MCDM methods, like fuzzy AHP adopted for Triangular Fuzzy Numbers (TFN) [37] or such as ELECTRE III and PROMETHEE II [27], [36] in the process of staff selection and recruitment.

Other methods used for solving similar problems are grey relational analysis (GRA) based methods [41]. A GRA-based intuitionistic fuzzy multi-criteria group decision making method for personnel selection is proposed in [23]. In this paper intuitionistic fuzzy weighted averaging

(IFWA) operator is utilized to aggregate individual opinions of decision makers and intuitionistic fuzzy entropy is used to obtain the entropy weights of the criteria.
The hybrid approach applying AHP and TOPSIS grey as an effective decision aid to improve human resources management in various areas of economic activities is proposed in [23]. A personnel selection system based AHP and Complex proportional assessment of alternatives with grey relations (COPRAS-G) method was proposed in [24]. The use of hesitant fuzzy sets as a powerful technique has been studied in [20]. This paper explores aggregation methods for prioritized hesitant fuzzy elements and their application on personnel evaluation.

The comparison of various methods in the selection of personnel may help in finding out the accuracy, appropriateness, suitability, fairness and practicality efficiently [37]. Using several independent sub-criteria for personnel selection we face with the lack of appropriate MCDM solutions. The purpose of this article is to offer methods of this problem solution.
Studies have shown that the subjective and objective assessments are different [2], [21]. In order to effective solutions, it is necessary to synthesize the objective and subjective assessments, this opens up new opportunities and improve the quality of the selection process [10]. Most previous studies did not provide clear criteria for grouping and ranking process patterns [30]. The proposed new KEMIRA method allows to combine criteria structuring and evaluating on the basis of subjective and objective levels. This method can be used to solve unstructured evaluation tasks when consensus among the experts on the importance of sub-criteria is not obligatory or when criteria of different nature (genesis) are used. It occurs in those areas of human activity where it is necessary the evaluation of experts from different specializations (personnel, finance, manufacturing and technology, etc.). Using this algorithm for decision-making, the decision-making process takes on a higher quality and greater integration levels to include in the evaluation and selection process a lot of important criteria of different nature, incomparable among themselves and couldn't be combined. For example, external and internal criteria; subjective and objective criteria; physical, mental and psychological indicators, etc. The proposed KEMIRA criteria selection and structuring model extends the MCDM approach. The advantage of proposed KEMIRA method is its efficiency for solving MCDM problems when the set of criteria is divided into a few subgroups having different origin. Nevertheless, Kemeny median method, which is a part of KEMIRA method, is applicable for much wider class of MCDM problems when it is necessary to establish priority of criteria. It must be mentioned that criteria weights are found at the same time as the process of MCDM problem solution (ranking the alternatives) goes on.

2 Problem formulation and scheme of solution

This work continues the series of works designed to solve MCDM problem of elite selection from security personnel, when two separate groups of criteria describing security guards are presented. In general case it may be more than two groups of criteria. External assessment values and internal measurements are given for each subject. Additionally, information about experts independently set preferences for criteria in each criterion group is known. The purpose of the study is selection of 10% best subjects according to the given information. Usually MCDM problems are being solved in the following order. At first stage of the problem solution data normalization procedure is performed. Then criteria weights are determined and finally MCDM methods applied for selecting best objects and/or objects ranking [29]. Each problem solution phase can be performed by different methods.

Our proposed KEmeny Median Indicator Rank Accordance (KEMIRA) [32] method consists of other three phases:

1. data normalization and standardization;
2. determination of criteria preferences reflecting the aggregated experts opinion;

3. minimization of the target functions value, determination of criteria weights and objects ranking.

The second step is determination priority of criteria by Kemeny median method. In this stage we'll use the information about criteria preferences set by experts. Note that the third step combines two phases into one - determination of criteria weights and objects ranking will be carried out simultaneously. This process is dynamic, so both goals will be achieved at the end of optimization procedure. Brief scheme of problem solution is depicted in Figure 1.

Comprehensive description of the method steps is provided below. At first introduce some notations: m – number of internal criteria (X), n – number of external criteria (Y), K – number of alternatives, S – number of experts. Initial decision-making matrix:

$$D = \begin{pmatrix} x_1^{(1)} & \dots & x_j^{(1)} & \dots & x_n^{(1)} & y_1^{(1)} & \dots & y_j^{(1)} & \dots & y_m^{(1)} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_1^{(i)} & \dots & x_j^{(i)} & \dots & x_n^{(i)} & y_1^{(i)} & \dots & y_j^{(i)} & \dots & y_m^{(i)} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_1^{(K)} & \dots & x_j^{(K)} & \dots & x_n^{(K)} & y_1^{(K)} & \dots & y_j^{(K)} & \dots & y_m^{(K)} \end{pmatrix}. \tag{1}$$

Criteria preferences estimated by experts:

Expert	x_1	...	x_l	...	x_n	y_1	...	y_l	...	y_m
1	$i_1^{(1)}$	...	$i_l^{(1)}$	...	$i_n^{(1)}$	$j_1^{(1)}$	...	$j_l^{(1)}$	...	$j_m^{(1)}$
...	...	...	...	...	...	...	...	...	...	...
s	$i_1^{(s)}$	...	$i_l^{(s)}$	...	$i_n^{(s)}$	$j_1^{(s)}$	...	$j_l^{(s)}$	...	$j_m^{(s)}$
...	...	...	...	...	...	...	...	...	...	...
S	$i_1^{(S)}$	...	$i_l^{(S)}$	...	$i_n^{(S)}$	$j_1^{(S)}$	...	$j_l^{(S)}$	...	$j_m^{(S)}$

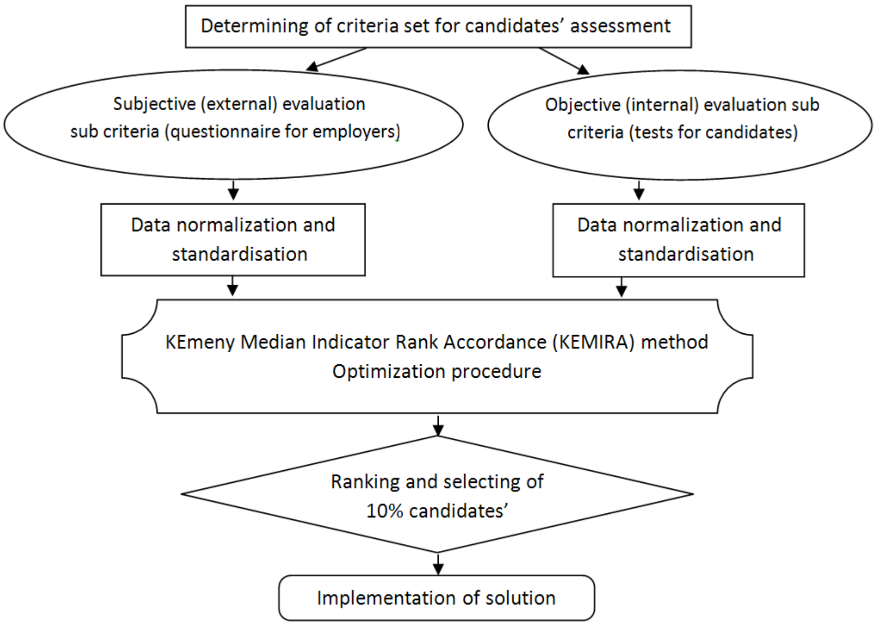

Determining of criteria set for candidates' assessment
Subjective (external) evaluation sub criteria (questionnaire for employers)
Objective (internal) evaluation sub criteria (tests for candidates)
Data normalization and standardisation
Data normalization and standardisation
KEmeny Median Indicator Rank Accordance (KEMIRA) method Optimization procedure
Ranking and selecting of 10% candidates'
Implementation of solution

Figure 1: Scheme of KEmeny Median Indicator Rank Accordance Method (KEMIRA) application for selecting the best objects.

Step 1. Normalization of elements of decision making matrix (1). All criteria are representing a benefit, i.e. the bigger is a value, the better is the respective alternative. Elements of decision-making matrix are normalized by formulas:

$$x_j^{(i)*} = \left(x_j^{(i)} - x_{min}^{(i)}\right) / \left(x_{max}^{(i)} - x_{min}^{(i)}\right), \quad y_j^{(i)*} = \left(y_j^{(i)} - y_{min}^{(i)}\right) / \left(y_{max}^{(i)} - y_{min}^{(i)}\right). \tag{2}$$

Step 2. Determining priority of criteria X and Y components. Criteria X and Y components priority is established independently by choosing priority which minimizes sum of distances to the priorities set up by all S experts:

$$R_A = \arg\min_R \sum_{j=1}^{S} \rho_A \left(R, R^{(j)}\right). \tag{3}$$

The result of this step is the median criteria components priority: $x_{j_1} \succ x_{j_2} \succ \cdots \succ x_{j_n}$, $y_{i_1} \succ y_{i_2} \succ \cdots \succ y_{i_m}$.
Step 3. Fixing initial weights of criteria satisfying median criteria components priority set in Step 2 and normalizing condition:

$$\alpha_{j_1} \succ \alpha_{j_2} \succ \cdots \succ \alpha_{j_n}, \quad \beta_{i_1} \succ \beta_{i_2} \succ \cdots \succ \beta_{i_m}, \sum_{r=1}^{n} \alpha_{j_r} = \sum_{s=1}^{m} \beta_{i_s} = 1. \tag{4}$$

Step 4. Calculation of functions $\varphi(X)$ and $\psi(Y)$ values and total value $\varphi(X) + \psi(Y)$ for each alternative.

$$\varphi(X) = \sum_{r=1}^{n} \alpha_{j_r} x_{j_r}, \ \psi(Y) = \sum_{s=1}^{m} \beta_{i_s} y_{i_s}. \tag{5}$$

Step 5. Calculation of ranks of alternatives according to the total value $\varphi(X) + \psi(Y)$: $R^{(k)}(\alpha)$ and $R^{(k)}(\beta)$, $k = 1, 2, \ldots, K$.
Step 6. Calculation number of elements having ranks lower than K_X according to the criteria X and lower than K_Y according to the criteria Y: $\left|A_{K_x} \cap B_{K_y}\right|$.
Step 7. Calculation sum of squared ranks differences for the alternatives satisfying conditions of Step 6, i.e. denote $A_{K_x} \cap B_{K_y} = \{k \in \{1, 2, \ldots, K\} : R^{(k)}(\alpha) \leqslant K_x \ , R^{(k)}(\beta) \leqslant K_y\}$. Then

$$\sum_{k \in \{A_{K_x} \cap B_{K_y}\}} \left(R^{(k)}(\alpha) - R^{(k)}(\beta)\right)^2. \tag{6}$$

Steps 4–7 are repeated until number of elements in Step 6 will reach its maximum value and the value of target function in Step 7 reach its minimum value. The obtained values of criteria weights $\alpha_{j_1}, \alpha_{j_2}, \cdots, \alpha_{j_n}, \beta_{i_1}, \beta_{i_2}, \cdots, \beta_{i_m}$ are used for calculation of total value $\varphi(X) + \psi(Y)$ and MCDM problem solution.

3 Determining priority of criteria

Say that priority of criteria $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_m)$ was estimated by S experts. Each expert sorted criteria in the priority descending order:

$$x_{i_1}^{(s)} \succ x_{i_2}^{(s)} \succ \ldots \succ x_{i_n}^{(s)}, \ y_{i_1}^{(s)} \succ y_{i_2}^{(s)} \succ \ldots \succ y_{i_m}^{(s)}, \tag{7}$$

here $s \in \{1, 2, \ldots, S\}$ – number of experts. We must ascertain generalized experts opinion. This task will be accomplished by Kemeny median method. Suppose that permutations $(j_1, j_2, \ldots, j_n)$ of the set of natural numbers $\{1, 2, \ldots, n\}$ determine priorities of vector (criteria) X components: $x_{j_1} \succ x_{j_2} \succ \cdots \succ x_{j_n}$. It means that for the set $\{x_1, x_2, \ldots, x_n\}$ a strict order relationship $R = (x_{j_1}, x_{j_2}, \ldots, x_{j_n})$ is defined. This relationship can be defined by the square matrix $A_R = \|a_{ij}\|$, which elements are: $a_{ij} = \begin{cases} 1, & \text{if } x_i < x_j, \\ 0, & \text{if } x_i \geqslant x_j. \end{cases}$, where $a_{ii} = 0$ and $a_{ij} = 1 - a_{ji}$, for $i \neq j$.

Define function

$$\rho_A\left(R^{(1)}, R^{(2)}\right) = \sum_{i=1}^{n}\sum_{j=1}^{n}\left|a_{ij}^{(1)} - a_{ij}^{(2)}\right|. \tag{8}$$

Function (8) is a certain measure of difference between two relationships and its values coincide with values of Kemeny distance function [28].

For example, if ranks of criteria $X = (x_1, x_2, x_3, x_4)$ components are established by the first expert as $R^{(1)} = (3, 2, 4, 1)$ and by the second expert as $R^{(2)} = (2, 1, 4, 3)$, consequently priorities of criteria components are set accordingly $x_4 \succ x_2 \succ x_1 \succ x_3$ and $x_2 \succ x_1 \succ x_4 \succ x_3$. Then corresponding matrices are:

$$A^{(1)} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}, \quad A^{(2)} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Function (8) in this case is gaining value $\rho_A\left(R^{(1)}, R^{(2)}\right) = \sum\limits_{i=1}^{4}\sum\limits_{j=1}^{4}\left|a_{ij}^{(1)} - a_{ij}^{(2)}\right| = 1+1+0+2 = 4.$

Suppose that S experts established priorities $R^{(1)}, R^{(2)}, \ldots, R^{(S)}$. Most consistent with these estimates will be priority R_A, which is called *median*:

$$R_A = \arg\min_{R}\sum_{j=1}^{S}\rho_A\left(R, R^{(j)}\right). \tag{9}$$

Notice that we can analyze functions ρ_A analogues, i.e. another distances between two relationships. The solution of (9) is not necessary a unique value. Sometimes we can obtain several medians which are solutions of problem (9).

If we are interested in the majority experts opinion only (not all experts) then we apply cluster analysis procedure, that helps to distinguish a cluster of experts majority. Then the median is sought among selected group of experts.

Let's us consider an example. Priority of criteria $X = (x_1, x_2, x_3, x_4)$ and $Y = (y_1, y_2, y_3)$ was estimated by 5 experts. Results of estimation presented in the Table 1. First expert set criterion X components priorities as follows: $x_4 \succ x_2 \succ x_1 \succ x_3$ and criterion Y components priorities: $y_1 \succ y_3 \succ y_2$. Notice that medians must be sought independently for criterion X and Y components. Let's calculate elements of the matrices $A_X^{(i)}$:

Table 1: Criteria X and Y components preferences established by 5 experts.

Expert	x_1	x_2	x_3	x_4	y_1	y_2	y_3
1	3	2	4	1	1	3	2
2	2	1	4	3	1	2	3
3	3	2	1	4	1	2	3
4	2	3	4	1	2	1	3
5	3	4	2	1	2	3	1

$$A_X^{(1)} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}, \quad A_X^{(2)} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad A_X^{(3)} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$A_X^{(4)} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}, \ A_X^{(5)} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

Note, that we must search the median R_{A_X} among all $24 = 4!$ components priorities options. Median is the priority, which minimize functions (9) value. In this case solution was found among priorities $R^{(1)} - R^{(5)}$ (for other priorities options value of function (9) is bigger):
$\sum\limits_{j=1}^{5} \rho_A \left(R^{(1)}, R^{(j)}\right) = 0 + 4 + 10 + 2 + 6 = 22,$ $\sum\limits_{j=1}^{5} \rho_A \left(R^{(2)}, R^{(j)}\right) = 4 + 0 + 6 + 6 + 10 = 26,$
$\sum\limits_{j=1}^{5} \rho_A \left(R^{(3)}, R^{(j)}\right) = 10 + 6 + 0 + 12 + 8 = 36,$ $\sum\limits_{j=1}^{5} \rho_A \left(R^{(4)}, R^{(j)}\right) = 2 + 6 + 12 + 0 + 4 = 24,$
$\sum\limits_{j=1}^{5} \rho_A \left(R^{(5)}, R^{(j)}\right) = 6 + 10 + 8 + 4 + 0 = 28$ ...

Median components priority is $R_{A_X} = R^{(1)} = (3, 2, 4, 1)$ or $x_4 \succ x_2 \succ x_1 \succ x_3$. Next we find criterion Y components priorities among all $6 = 3!$ options. Calculate elements of matrices $A_Y^{(i)}$:

$$A_Y^{(1)} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \ A_Y^{(2)} = A_Y^{(3)} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \ A_Y^{(4)} = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix},$$

$$A_Y^{(5)} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}, \ A_Y^{(6)} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \ A_Y^{(7)} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}.$$

Here $R^{(6)} = (3, 1, 2), R^{(7)} = (3, 2, 1)$. $\sum\limits_{j=1}^{5} \rho_A \left(R^{(1)}, R^{(j)}\right) = 0 + 2 + 2 + 4 + 2 = 10,$
$\sum\limits_{j=1}^{5} \rho_A \left(R^{(2)}, R^{(j)}\right) = \sum\limits_{j=1}^{5} \rho_A \left(R^{(3)}, R^{(j)}\right) = 2 + 0 + 0 + 2 + 4 = 8,$
$\sum\limits_{j=1}^{5} \rho_A \left(R^{(4)}, R^{(j)}\right) = 4 + 2 + 2 + 0 + 6 = 14,$ $\sum\limits_{j=1}^{5} \rho_A \left(R^{(5)}, R^{(j)}\right) = 2 + 4 + 4 + 6 + 0 = 16,$
$\sum\limits_{j=1}^{5} \rho_A \left(R^{(6)}, R^{(j)}\right) = 6 + 4 + 4 + 2 + 4 = 20,$ $\sum\limits_{j=1}^{5} \rho_A \left(R^{(7)}, R^{(j)}\right) = 4 + 6 + 6 + 4 + 2 = 22.$

Median components prioritiy is $R_{A_Y} = R^{(2)} = R^{(3)} = (1, 2, 3)$ or $y_1 \succ y_2 \succ y_3$.

4 Calculation of weights and MCDM problem solution

Suppose that there are known objective measurements (x) and subjective expert evaluations (y) of K test takers $T^{(j)}, j = 1, 2, \ldots, K : x_1^{(j)}, x_2^{(j)}, \ldots, x_n^{(j)}, y_1^{(j)}, y_2^{(j)}, \ldots, y_m^{(j)}, 0 \leqslant x_i^{(j)}, y_i^{(j)} \leqslant 1$:

$$T^{(j_1)} \succeq T^{(j_2)}, \text{ if } (\forall i) \ x_i^{(j_1)} \geqslant x_i^{(j_2)} \ \& \ y_i^{(j_1)} \geqslant y_i^{(j_2)}. \tag{10}$$

Suppose that $0 \leqslant w_{xi}, w_{yi} \leqslant 1$ are weighted coefficients: $\sum\limits_{i=1}^{n} w_{xi} = \sum\limits_{i=1}^{m} w_{yi} = 1$. Then under conditions (10) the following inequalities take place:

$$W_x^{(j_1)} \geqslant W_x^{(j_2)} \ \& \ W_y^{(j_1)} \geqslant W_y^{(j_2)}. \tag{11}$$

Here

$$W_x^{(j)} = \sum_{i=1}^{n} w_{xi} x_i^{(j)}, \ W_y^{(j)} = \sum_{i=1}^{m} w_{yi} y_i^{(j)}. \tag{12}$$

It means that if all measurements values are bigger for object $T^{(j_1)}$ than for object $T^{(j_2)}$ then each linear combination of measurements for $T^{(j_1)}$ will also be bigger than for $T^{(j_2)}$ provided that weighted coefficients are nonnegative. In practice more often we have encountering situation when results of the measurements are as follows: $x_{i_1}^{(j_1)} > x_{i_1}^{(j_2)}$ and $x_{i_2}^{(j_1)} < x_{i_2}^{(j_2)}$. Then it isn't possible to apply criteria (10) for selecting better alternative. The idea of current study is to choose such values of weights w_{xi}, w_{yi} which will guarantee proximity of values $W_x^{(j)}$ and $W_y^{(j)}$ for 10% of best security guards. The measure of closeness of these values would be sum of squares of ranks differences.

Let's denote $R_x^{(j)}$ and $R_y^{(j)}$ positive integers $R_{x,y}^{(j)} \in \{1, 2, \ldots, K\}$ satisfying condition: $R_{x,y}^{(j_1)} < R_{x,y}^{(j_2)}$, when $W_{x,y}^{(j_1)} > W_{x,y}^{(j_2)}$, i.e. they are ranks of numbers $W_{x,y}^{(j)}$. Let's define A_{K_x} and B_{K_y} as subsets of the set $\{1, 2, \ldots, K\}$: $A_{K_x} = \Big\{\{j_1, j_2, \ldots, j_{K_x}\} : R_x^{(j_i)} \leqslant K_x\Big\}$, $B_{K_y} = \Big\{\{j_1, j_2, \ldots, j_{K_y}\} : R_y^{(j_i)} \leqslant K_y\Big\}$ including objects which have lowest ranks according to the corresponding criteria (X and Y) and having K_x and K_y elements respectively. Note, that the lower is the rank, the better is the alternative.

The best alternatives selection task according to both criteria (12) is equivalent to the task of finding the intersection of sets $A_{K_x} \cap B_{K_y}$, which has the required number of elements $\left|A_{K_x} \cap B_{K_y}\right|$. If the number of elements $\left|A_{K_x} \cap B_{K_y}\right|$ is insufficient, it is necessary to increase numbers K_x and K_y. Preferences of the selected alternatives could be determined by the following criteria expressed as sum of criteria (12): $W^{(j)} = W_x^{(j)} + W_y^{(j)}$.

Consider the following example. Suppose that 5 objects have 4 attributes of one type (X) and 3 attributes of other type – (Y). Values of attributes are given in the Table 2. Let's calculate values of criteria (12) for the first object when weights are as follows:

Table 2: Attributes x_i and y_j values for the 5 given objects.

	x_1	x_2	x_3	x_4	y_1	y_2	y_3
1	0.4	0.6	0.7	0.3	0.5	0.4	0.2
2	0.5	0.3	0.7	0.3	0.6	0.7	0.2
3	0.2	0.4	0.6	0.5	0.7	0.7	0.4
4	0.5	0.3	0.5	0.4	0.6	0.8	0.2
5	0.6	0.2	0.8	0.8	0.4	0.5	0.8

$w_{x,1} = \frac{1}{2}$, $w_{x,2} = \frac{1}{4}$, $w_{x,3} = \frac{1}{8}$, $w_{x,4} = \frac{1}{8}$, $w_{y,1} = \frac{5}{8}$, $w_{y,2} = \frac{1}{4}$, $w_{y,3} = \frac{1}{8}$. We obtain criteria (12) values: $W_x^{(1)} = \frac{1}{2}0.4 + \frac{1}{4}0.6 + \frac{1}{8}0.7 + \frac{1}{8}0.3 = 0.475, W_y^{(1)} = \frac{5}{8}0.5 + \frac{1}{4}0.4 + \frac{1}{8}0.2 = 0.4375$.

Similarly calculate criteria values for the remaining 4 objects. All criteria values and their ranks are presented in the Table 3. Previously defined sets A_{K_x} and B_{K_y} are those including respectively K_x and K_y elements with lowest ranks $R_x^{(j)}$ and $R_y^{(j)}$: $A_1 = \{5\}$, $A_2 = \{1, 5\}$, $A_3 = \{1, 2, 5\}$, $A_4 = \{1, 2, 4, 5\}$, $B_1 = \{3\}$, $B_2 = \{3, 4\}$, $B_3 = \{2, 3, 4\}$, $B_4 = \{2, 3, 4, 5\}$, $A_1 \cap B_1 = \varnothing$, $A_2 \cap B_2 = \varnothing$, $A_3 \cap B_3 = \{2\}$ and so on.

Table 3: Criteria $W_x^{(j)}$ and $W_y^{(j)}$ and their ranks values for the 5 given objects.

j	$W_x^{(j)}$	$W_y^{(j)}$	$R_x^{(j)}$	$R_y^{(j)}$
1	0.475	0.4375	2	5
2	0.45	0.575	3	3
3	0.3375	0.6625	5	1
4	0.4375	0.60	4	2
5	0.55	0.475	1	4

Calculation of criteria weights is performed simultaneously with the decision of MCDM problem. Suppose that there are known criteria-referenced assessments for certain alternatives $X^{(k)} = \left(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}\right)$, $Y^{(k)} = \left(y_1^{(k)}, y_2^{(k)}, \dots, y_m^{(k)}\right)$, $k = 1, 2, \dots, K$ and criteria X, Y priorities are determined:

$$x_{j_1} \succ x_{j_2} \succ \dots \succ x_{j_n}, \quad y_{i_1} \succ y_{i_2} \succ \dots \succ y_{i_m}. \tag{13}$$

According to the established criteria X, Y priorities (13) functions $\varphi(X)$ and $\psi(Y)$ are determined as follows:

$$\varphi(X) = \sum_{r=1}^{n} \alpha_{j_r} x_{j_r}, \text{ where } \alpha_{j_1} \geqslant \alpha_{j_2} \geqslant \ldots \geqslant \alpha_{j_n} > 0, \tag{14}$$

$$\psi(Y) = \sum_{s=1}^{m} \beta_{i_s} y_{i_s}, \text{ where } \beta_{i_1} \geqslant \beta_{i_2} \geqslant \ldots \geqslant \beta_{i_m} > 0. \tag{15}$$

For identifying criteria weights we use the *heuristic* described in [13]. Require that the weighting coefficients α_j and β_i satisfy normalizing condition (4).

Denote $\mathcal{S}_{(j_1, j_2, \dots, j_n)}$ class of all convolutions constructed on the base of weighted averages and having priority feature $(j_1, j_2, \dots, j_n)$, i.e. satisfying condition (14). Analogous $\mathcal{S}_{(i_1, i_2, \dots, i_m)}$ is a class of all convolutions having priority feature $(i_1, i_2, \dots, i_m)$ (satisfying condition (15)).

Suppose that φ and ψ are criteria X and Y convolutions having corresponding priority features (13). For each alternative $\left(X^{(k)}, Y^{(k)}\right)$ we'll calculate values of both criteria convolutions $\varphi\left(X^{(k)}\right)$, $\psi\left(Y^{(k)}\right)$. Numbering them in ascending order we get ranks of alternatives: $R_x^{(k)}$, $R_y^{(k)}$.

Denote A_{K_x} and B_{K_y} sets of the best alternatives – i. e. those subsets of the set $\{1, 2, \dots, K\}$, whose elements ranks satisfy the inequalities $R_x^{(k)} \leqslant K_x$ and $R_y^{(k)} \leqslant K_y$. In the set A_{K_x} there are K_x the best alternatives according to criteria X, similarly in the set B_{K_y} there are K_y the best alternatives according to criteria Y. Numbers K_x ir K_y are chosen so that the intersection of sets $A_{K_x} \cap B_{K_y}$ have not less than 10 % of the best alternatives.

Then we search such functions φ and ψ that the number of elements of sets A_{K_x} and B_{K_y} intersection should be the maximum:

$$\max_{\substack{\varphi \in \mathcal{S}_{(j_1, j_2, \dots, j_n)} \\ \psi \in \mathcal{S}_{(i_1, i_2, \dots, i_m)}}} \left| A_{K_x} \cap B_{K_y} \right|. \tag{16}$$

Condition (16) means that we must select convolutions φ and ψ, which maximize criteria X and Y compatibility. The number of convolutions φ and ψ, maximizing (16) can be great, therefore additional optimization problem must be formulated.

Denote $R^{(k)}(\alpha)$ and $R^{(k)}(\beta)$ ranks of the numbers $\left\{\varphi\left(X^{(1)}\right), \varphi\left(X^{(2)}\right), \dots, \varphi\left(X^{(K)}\right)\right\}$ and $\left\{\psi\left(Y^{(1)}\right), \psi\left(Y^{(2)}\right), \dots, \psi\left(Y^{(K)}\right)\right\}$ respectively $(k = 1, 2, \dots, K)$.

We'll apply Indicator Rank Accordance method for minimizing ranks discrepancy function, i. e. sum of squares of the highest ranks differences according to criteria X and Y:

$$CR_{K_x, K_y}(\alpha, \beta) = \min_{\substack{\varphi \in \mathcal{S}_{(j_1, j_2, \dots, j_n)} \\ \psi \in \mathcal{S}_{(i_1, i_2, \dots, i_m)}}} \sum_{k \in \{A_{K_x} \cap B_{K_y}\}} \left(R^{(k)}(\alpha) - R^{(k)}(\beta)\right)^2. \tag{17}$$

Here K_x and K_y are chosen so that the number of elements in the intersection $A_{K_x} \cap B_{K_y}$ will be equal to the desired number of selected objects.

For example, suppose that for the data given in Table 2 five experts proposed criteria priorities presented in Table 1. In Chapter 3 for these data we have determined criteria priority features as follows: $x_4 \succ x_2 \succ x_1 \succ x_3, \quad y_1 \succ y_2 \succ y_3$.
Therefore, we'll search for weights values, satisfying conditions

$$\alpha_4 \geqslant \alpha_2 \geqslant \alpha_1 \geqslant \alpha_3, \quad \beta_1 \geqslant \beta_2 \geqslant \beta_3 \tag{18}$$

and normalizing conditions (4) which maximize number of elements in the intersection of sets (16) and minimize value of function (17). Let's choose $K_x = K_y = 3$ and initial values of coefficients as follows: $\alpha_1 = \frac{1}{12}$, $\alpha_2 = \frac{1}{3}$, $\alpha_3 = \frac{1}{12}$, $\alpha_4 = \frac{1}{2}$, $\beta_1 = \beta_2 = \beta_3 = \frac{1}{3}$. Values of functions $\varphi(X^{(i)})$ and $\psi(Y^{(i)})$ and their ranks for each alternative are given in Table 4.

Table 4: Values of functions $\varphi(X^{(i)})$ and $\psi(Y^{(i)})$ and their ranks values for the 5 given objects.

j	$W_x^{(j)}$	$W_y^{(j)}$	$R_x^{(j)}$	$R_y^{(j)}$
1	0.4417	0.33	3	5
2	0.35	0.45	5	4
3	0.45	0.54	2	1
4	0.3833	0.48	4	3
5	0.5833	0.51	1	2

$A_3 = \{1,3,5\}$, $B_3 = \{3,4,5\}$, $A_3 \cap B_3 = \{3,5\}$, $|A_3 \cap B_3| = 2$.
So, in the sum (17) we have only 2 summands: $CR_{3,3}(\alpha,\beta) = (2-1)^2 + (1-2)^2 = 2$.
In the next step we choose other values of coefficients satisfying conditions (18) and (4). If the number of elements in the intersection of sets (16) is greater or equal than 2, we calculate function (17) value. If this value is less than in previous step, values of these new coefficients are considered to be the best solution before the next step. After the last step when all coefficients satisfying conditions (18) and (4) are verified we'll receive values of coefficients $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \beta_1, \beta_2, \beta_3$ which are the solution of MCDM problem. Finally, values of functions $\varphi(X)$ and $\psi(Y)$ are calculated and objects are ranked according to the criteria $\varphi(X) + \psi(Y)$.

5 Case study

The present problem has been solved in [13]. In the mentioned paper weighted coefficients of criteria were calculated as the proportions of the collected scores to the total score. In the current research the weights obtained by KEMIRA method. Results of selecting the best 10% security guards will be compared with previous results. Moreover, guard ranking will be performed.

118 security guards were randomly selected from the company G4S Lietuva. 22 leader managers (experts) ranked the competences described below.

Personnel elite – 10% the best employees of the private security company – are selected according to six internal assessment criteria ($x_1 - x_6$) and nine external evaluation criteria ($y_1 - y_9$). 6 internal criteria are objective tests and measurements: x_1 is employee's theoretical and practical preparation; x_2 – professional activity, x_3 – mental qualities; x_4 – physical development; x_5 – motor abilities (personal physical conditions allowing to carry out physical tasks); x_6 – fighting efficiency. 9 external criteria are evaluation of subordinate by his immediate superior: y_1 – specialty knowledge, professionalism, y_2 – diligence and positive attitude to work, y_3 – behavior with colleagues and supervisors; y_4 – reliability at work; y_5 – quality of work; y_6 – workload performance; y_7 – image; y_8 – development rate; y_9 – being promising (potential to make a career). Fragment of security guards evaluation criteria structure is given in Table 5. Table 5 data are given in the paper [13]. Note that all 15 criteria are associated with benefit and their

Table 5: Security guards internal and external evaluation criteria.

Security guards	Internal criteria				External criteria			
	x_1	x_2	$\dots$	x_6	y_1	y_2	$\dots$	y_9
a_1	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
a_{118}	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$

greater value is better. Therefore criteria values were transformed to values x_i^{j*}, y_i^{j*}, belonging to the range $[0;1]$ by the transformations (2). 22 independent experts determined preferences separately for internal and external evaluation criteria. Criteria preferences established by experts are presented in the Table 6. A higher grade means that the criterion is more important.

Criterion X priority feature is determined from expert estimates of the form $(j_1, j_2, \dots, j_6)$. For example, first expert set criterion X components priorities as follows: $(x_1, x_3, x_6, x_4, x_5, x_2)$. Here x_1 is the most important and x_2 – the least important component. Criterion Y priority feature is established similarly. For example, first expert set them in this order: $(y_2, y_4, y_1, y_7, y_5, y_3, y_6, y_8, y_9)$. Generalized experts opinion expressing criteria priority feature is established by calculating median with the respective metric (8) to calculate distances between priorities. Calculation of medians was performed by full re-selection of options, which is respectively $6! = 720$ and $9! = 362880$.

Medians were calculated separately for criterion X components x_j and criterion Y components y_i by minimizing function (9) values. Two solutions were obtained for criterion X components x_j priority features: $(1,5,3,6,4,2)$, $(1,3,5,6,4,2)$. Criterion Y components y_i priority features were determined uniquely: $(2,4,5,3,1,7,6,8,9)$. Therefore, criteria X and Y components order was established respectively:

$$x_1 \succ x_5 \succ x_3 \succ x_6 \succ x_4 \succ x_2 \quad \text{or} \quad x_1 \succ x_3 \succ x_5 \succ x_6 \succ x_4 \succ x_2, \tag{19}$$

$$y_2 \succ y_4 \succ y_5 \succ y_3 \succ y_1 \succ y_7 \succ y_6 \succ y_8 \succ y_9. \tag{20}$$

Table 6: Criteria X and Y components preferences established by experts.

Expert	$x_1 \cdots x_6$	$y_1 \cdots y_9$	Expert	$x_1 \cdots x_6$	$y_1 \cdots y_9$
1	615324	794853621	12	624153	695784321
2	523164	795682341	13	645132	596873421
3	513264	594681732	14	635142	786954321
4	615342	597863421	15	413265	597863241
5	615234	796853421	16	413265	798654231
6	514263	596873421	17	614235	687945231
7	423165	485972631	18	634152	698745231
8	625341	685973421	19	415263	498671532
9	546321	794863512	20	624153	596783421
10	546123	674983521	21	526341	687954321
11	413265	586972431	22	416325	697854231

Finally, cluster analysis [17] was applied to the data presented in the Table 6 to distinguish a group of experts whose opinions are very close to each other and which form the majority of the experts. By applying Between groups, Ward's and Furthest neighbor methods the group of experts with numbers {1,4,5,8,9,10,12,13,14,17,18,20,21,22} was selected. Further only those experts priorities were examined to get criteria priority preferences, because we did not want to distort the opinion of this group by the remaining minority group of experts. Kemeny median method was applied for this reduced data array. For criterion X components x_j the solution minimizing function (9) received as follows: $(1, 3, 5, 6, 2, 4)$. Results for criterion Y components y_j were the same as in (20). Therefore, for the majority of experts criteria X and Y components order was established respectively:

$$x_1 \succ x_3 \succ x_5 \succ x_6 \succ x_2 \succ x_4 \tag{21}$$

and (20). From the priority features (19), (21) and (20) it follows, that it is necessary to search for convolutions φ and ψ in the form:

$$\varphi_1(X) = \alpha_1 x_1 + \alpha_5 x_5 + \alpha_3 x_3 + \alpha_6 x_6 + \alpha_4 x_4 + \alpha_2 x_2, \quad \alpha_1 \geqslant \alpha_5 \geqslant \alpha_3 \geqslant \alpha_6 \geqslant \alpha_4 \geqslant \alpha_2 > 0,$$

$$\varphi_2(X) = \alpha_1 x_1 + \alpha_3 x_3 + \alpha_5 x_5 + \alpha_6 x_6 + \alpha_4 x_4 + \alpha_2 x_2, \quad \alpha_1 \geqslant \alpha_3 \geqslant \alpha_5 \geqslant \alpha_6 \geqslant \alpha_4 \geqslant \alpha_2 > 0,$$

$$\varphi_3(X) = \alpha_1 x_1 + \alpha_3 x_3 + \alpha_5 x_5 + \alpha_6 x_6 + \alpha_2 x_2 + \alpha_4 x_4, \quad \alpha_1 \geqslant \alpha_3 \geqslant \alpha_5 \geqslant \alpha_6 \geqslant \alpha_2 \geqslant \alpha_4 > 0,$$

$$\psi(Y) = \beta_2 y_2 + \beta_4 y_4 + \cdots + \beta_9 y_9, \quad \beta_2 \geqslant \beta_4 \geqslant \beta_5 \geqslant \cdots \geqslant \beta_8 \geqslant \beta_9 > 0.$$

We define parameter values for our problem: number of investigated objects $K = 118$, number of criterion X components $n = 6$, number of criterion Y components $m = 9$. Values $K_x = 21$ and $K_y = 21$ were chosen to fulfill the condition $\left|A_{K_x} \cap B_{K_y}\right| = 12$, since the goal is to select the top 12 security guards. In the case of functions $\varphi_1(X)$ and $\psi(Y)$ the lowest criterion (17) value was found $CR_{21,21}(\alpha, \beta) = 331$ with the following values of weighted coefficients:

α_1	α_5	α_3	α_6	α_4	α_2	β_2	β_4	β_5	β_3	β_1	β_7	β_6	β_8	β_9
0.202	0.202	0.197	0.191	0.11	0.098	0.32	0.149	0.14	0.12	0.085	0.064	0.053	0.035	0.034

The best 12 security guards, belonging to intersection of sets $A_{21} \cap B_{21}$ and ranked according to the criterion $W_1^{(j)} = \varphi_1(x^{(j)}) + \psi(y^{(j)})$ as follows: $a_{56} \succ a_{76} \succ a_{91} \succ a_{81} \succ a_{21} \succ a_{47} \succ a_{36} \succ a_{106} \succ a_{34} \succ a_{111} \succ a_{102} \succ a_{77}$. In the case of functions $\varphi_2(X)$ and $\psi(Y)$ the lowest criterion (17) value $CR_{21,21}(\alpha, \beta) = 362$ was obtained with these weighted coefficients values:

α_1	α_3	α_5	α_6	α_4	α_2	β_2	β_4	β_5	β_3	β_1	β_7	β_6	β_8	β_9
0.194	0.192	0.188	0.182	0.132	0.11	0.29	0.24	0.094	0.089	0.082	0.078	0.055	0.055	0.017

The same 12 security guards got to the intersection of sets $A_{21} \cap B_{21}$. Alternatives ranked according to the criterion $W_2^{(j)} = \varphi_2(x^{(j)}) + \psi(y^{(j)})$ as follows: $a_{56} \succ a_{76} \succ a_{91} \succ a_{81} \succ a_{21} \succ a_{47} \succ a_{36} \succ a_{106} \succ a_{111} \succ a_{34} \succ a_{102} \succ a_{77}$. Two alternatives have exchanged places: a_{111} and a_{34}. The obtained results coincide with the results got in [13], since in the case of $\varphi_2(X)$ and $\psi(Y)$ criteria priorities were set in the same order as in the mentioned article. For the generalized opinion of the experts majority determined by cluster analysis the solution was sought between the functions of the form $\varphi_3(X)$ and $\psi(Y)$. The minimum (17) value $CR_{19,19}(\alpha, \beta) = 373$ was obtained with these weighted coefficients values:

α_1	α_3	α_5	α_6	α_2	α_4	β_2	β_4	β_5	β_3	β_1	β_7	β_6	β_8	β_9
0.2	0.19	0.19	0.19	0.133	0.097	0.25	0.249	0.12	0.1	0.081	0.07	0.05	0.05	0.03

The top 12 security guards are: a_{21}, a_{34}, a_{36}, a_{47}, a_{56}, a_{72}, a_{76}, a_{81}, a_{91}, a_{102}, a_{106}, a_{111}. The difference from the previous cases when solutions were sought amongst functions $\varphi_1(X)$ or $\varphi_2(X)$ is that a_{77} was changed by a_{72}. Alternatives ranked according to the criterion $W_3^{(j)} = \varphi_3(x^{(j)}) + \psi(y^{(j)})$ as follows: $a_{56} \succ a_{91} \succ a_{76} \succ a_{81} \succ a_{21} \succ a_{36} \succ a_{111} \succ a_{47} \succ a_{106} \succ a_{34} \succ a_{102} \succ a_{72}$. Ranking results differ from those accomplished by criteria $W_1^{(j)}$ and $W_2^{(j)}$ much more.

6 Discussion and conclusions

Many authors studied the relationship between the set of qualitative indicators of human resources and business performance. Recently, this phenomenon is gaining considerable economic importance. A lot of recruitment methods, whose main aim is to help organizations make the best personnel management decisions, are being created. Many authors have expressed concern about the human resources performance assessment process because it can be biased due to the non-compliance of methods [1]. Traditional human resource evaluation and selection methods are usually based on a statistical analysis for evaluation of objective indicators thought to reflect the realities. However, it is important to assess not only known but also unknown, hard-diagnosable factors. Solution of this problem requires modern techniques involving large amounts of uncertain and subjective information [38]. For the assessment of indicators it is necessary to apply a taxonomy based sorting principles [7]. The different types of equally important factors should be grouped into separate sub-criteria [41]. There can be two or more sub-criteria. In selection process sub-criteria are given the same weights. This paper presents an algorithm which opens possibilities for recruitment in compliance with the above requirements. Sub-criteria applied to the evaluation of candidates reflect subjective and objective information about the candidates. In the current paper two equally important hierarchical structures were generated and a new method for ranking proposed. This method combines expert (subjective) evaluation and testing indicators (objective evaluation) hierarchical layout.

In the article a new approach to MCDM problem, when objects are evaluated by two groups of criteria having different origin, is presented. In the first stage criteria components preferences are established separately in each group by applying the novel method of Kemeny medians. Method of Kemeny medians can be used when information about criteria components preferences established by independent experts is available. In the second stage criteria weights are determined and MCDM problem is solved by applying proposed Indicator Rank Accordance method. Principle of this method is to choose criteria weights values from the set of all possible values which maximize number of elements having lowest ranks according to the both criteria (16) and minimize sum of squared ranks differences calculated for two groups of criteria (17). A case study of selecting top 12 security guards was analyzed by KEMIRA method. For the proposed data method of Kemeny medians was applied twice: for all experts and for the majority group of experts distinguished by cluster analysis methods.

The proposed methodology allows to weigh and synthesize subjective (managers assessments) and objective (the candidates skills) indicators. In general, there can be more than two groups of criteria having different origin. Method of Kemeny medians is useful for a wide range of MCDM problems when priority of criteria must be established according to experts evaluation. New KEMIRA method opens up the new opportunities of application and development of decision-making methods not only in the selection of personnel. It is suggested that this type of research could be extended to other areas of human activities where MCDM problems arise (business, manufacturing, trade and etc).

Bibliography

[1] A. Afshari, M. Mojahed, R.M. Yusuff (2010); Simple additive weighting approach to personnel selection problem, *International Journal of Innovation, Management and Technology*, 1(5): 511–515.

[2] P. Arezes, M. Neves, S. Teixeira, C. Leão, J. Cunha (2013);Testing thermal comfort of trekking boots: An objective and subjective evaluation. *Applied Ergonomics*, 44(4): 557–

565.

[3] A. Arisha, W. Abo-Hamad (2013); Towards Operations Excellence: Optimising Staff Scheduling For New Emergency Department. Proceedings of the 20th International Annual EurOMA Conference - "Operations Management at the Heart of the Recovery", 9-12 June 2013, (pp. 1–10). Dublin, Ireland.

[4] A. Baležentis, T. Baležentis, W.K.M. Brauers (2012); Personnel selection based on computing with words and fuzzy MULTIMOORA. *Expert Systems with Applications*, 39(9): 7961–7967.

[5] A. Baležentis, T. Baležentis, W.K.M. Brauers (2012); MULTIMOORA-FG: a multi-objective decision making method for linguistic reasoning with an application to personnel selection. *Informatica*, 23(2): 173–190.

[6] S. Balli, S. Korukoğlu (2014); Development of a fuzzy decision support framework for complex multi-attribute decision problems: a case study for the selection of skilful basketball players. *Expert Systems*, 31(1): 56–69.

[7] J. Bednarik, W. Andreff, S. Popović, D. Jakšić, E. Kolar, G. Jurak (2013); Financial taxonomy of non-governmental sports organizations, *Kinesiology*, 45(2): 241–251.

[8] J. van den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, L. De Boeck (2013); Personnel scheduling: A literature review. *European Journal of Operational Research*, 226(3): 367–385.

[9] C. Chen, P. Pai, W. Hung (2011); Applying linguistic VIKOR and knowledge map in personnel selection. *Asia Pacific Management Review*, 16(4): 491–502.

[10] L.-C. Cheng, H.-A. Wang (2014); A fuzzy recommender system based on the integration of subjective preferences and objective information, *Applied Soft Computing*, 18: 290–301.

[11] S. Dadeło (2005); Czynniki determinujące kompetencje pracowników ochrony na Litwie. AWF Warszawa-Vilnius. [in Polish, abstract in English, in Lithuanian]

[12] S. Dadelo, Z. Turskis, E.K. Zavadskas, R. Dadelienė, (2013);Integrated multi-criteria decision making model based on wisdom-of-crowds principle for selection of the group of elite security guards, *Archives of Budo*, 9(2): 135–147.

[13] S. Dadelo, A. Krylovas, N. Kosareva, E.K. Zavadskas, R. Dadelienė (2014); Algorithm of maximizing the set of common solutions for several MCDM problems and it's application for security personnel scheduling, *International journal of computers, communications & Control*, 9(2): 140–148.

[14] M. Dağdeviren (2010); A hybrid multi-criteria decision-making model for personnel selection in manufacturing systems, *Journal of Intelligent Manufacturing*, 21(4): 451–460.

[15] A. Dahlstrom, M. Campbell, C. Hewitt (2013); The role of uncertainty and subjective influences on consequence assessment by aquatic biosecurity experts. *Journal of Environmental Management*, 127(30): 103–113.

[16] M. El-Santawy, R. El-Dean (2012); On using VIKOR for ranking personnel problem. *Life Science Journal-Acta*, 9(4): 1534–1536.

[17] V. Estivill-Castro (2002); Why so many clustering algorithms – A Position Paper. *ACM SIGKDD Explorations Newsletter* 4(1): 65–75.

[18] J. Fadyl, K. Mcpherson, P. Schlüter, L. Turner-Stokes (2010); Factors contributing to work-ability for injured workers: literature review and comparison with available measures. *Disability and Rehabilitation*, 32(14): 1173–1183.

[19] S. Fazli, H. Jafari (2002); Developing a hybrid multi-criteria model for investment in stock exchange. *Management Science Letters*, 2(2): 457–468, 2002.

[20] B. Garland B, N. Hogan, T. Kelley, B. Kim, E. Lambert E. (2013); To Be or Not to Be Committed: The Effects of Continuance and Affective Commitment on Absenteeism and Turnover Intent among Private Prison Personnel. *Journal of Applied Security Research*, 8(1): 1–23.

[21] A. Gasiorowska (2014); The relationship between objective and subjective wealth is moderated by financial control and mediated by money anxiety, *Journal of Economic Psychology*, 43: 64–74.

[22] J. Grand, J. Golubovich, A. Ryan, N. Schmitt (2013); The detection and influence of problematic item content in ability tests: An examination of sensitivity review practices for personnel selection test development. *Organizational behavior and human decision processes*, 121(2): 158–173.

[23] S. Hashemkhani Zolfani, J. Antuchevičienė (2012); Team member selecting based on AHP and TOPSIS grey. *Inzinerine Ekonomika - Engineering Economics*, 23(4): 425–434.

[24] S. Hashemkhani Zolfani, N. Rezaeiniya, M. Aghdaie, E.K. Zavadskas (2012); Quality control manager selection based on AHP-COPRAS-G method: a case in Iran. *Ekonomska istraživanja – Economic Research*, 25(1): 88–104.

[25] M. Kabak (2013); A fuzzy DEMATEL-ANP based multi criteria decision making approach for personnel selection. *Journal of Multiple-Valued Logic and Soft Computing*, 20(5-6): 571–593.

[26] M. Kabak, S. Burmaoğlu, Y. Kazançoğlu (2012); A fuzzy hybrid MCDM approach for professional selection. *Expert Systems with Applications*, 39(3): 3516–3525.

[27] A. Kelemenis, D. Askounis, A new TOPSIS-based multi-criteria approach to personnel selection. *Expert Systems with Applications*, 37(7): 4999–5008, 2010.

[28] J. Kemeny, J. Snell (1963); *Mathematical Models in the Social Sciences*, New York.

[29] M. Kendall (1970); *Rank correlation methods*, fourth ed., Charles Griffin & Co., London.

[30] V. Keršulienė, Z. Turskis (2013); An Integrated Multi-criteria Group Decision Making Process: Selection of the Chief Accountant. The 2-nd International Scientific conference "Contemporary Issues in Business, Management and Education 2013", *Procedia - Social and Behavioral Sciences*, 110: 897–904.

[31] G. Kou, Y. Lu, Y. Peng, Y. Shi (2012); Evaluation of classification algorithms using MCDM and rank correlation. *International Journal of Information Technology & Decision Making*, 11(1): 197–225.

[32] A. Krylovas, E.K. Zavadskas, N. Kosareva, S. Dadelo (2014); New KEMIRA method for determining criteria priority and weights in solving MCDM problem. *International Journal of Information Technology & Decision Making*, 13(6): 1119–1133.

[33] J. Liou, G. Tzeng (2012); Comments on "Multiple criteria decision making (MCDM) methods in economics: an overview". *Technological and Economic Development of Economy*, 18(4): 672–695.

[34] P. Papsiene, S. Vaitkevicius (2014); Human Resource Assessment Impact to Organization Climate: Case of Lithuanian Public Sector Organizations, *Inzinerine Ekonomika – Engineering Economics*, 25(2): 223–230.

[35] Y. Peng, G. Kou, G. Wang, Y. Shi (2011); FAMCDM: A Fusion Approach of MCDM Methods to Rank Multiclass Classification Algorithms. *Omega*, 39(6): 677–689.

[36] B. Rouyendegh, T. Erkan (2013); An Application of the fuzzy ELECTRE method for academic staff selection. *Human Factors and Ergonomics in Manufacturing & Service Industries*, 23(2): 107–115.

[37] B. Rouyendegh, T. Erkan (2012); Selection of academic staff using the fuzzy analytic hierarchy process (FAHP): a pilot study. *Tehnicki vjesnik–Technical Gazette*, 19(4): 923–929.

[38] A. Salehi, M. Izadikhah (2014); A novel method to extend SAW for decision-making problems with interval data, *Decision Science Letters*, 3(2): 225–236.

[39] G. Tzeng, J. Huang (2011); *Multi Attribute Decision Making: Methods and Applications*, CRCPress.

[40] D. Yu, W. Zhang, Y. Xu (2013); Group decision making under hesitant fuzzy environment with application to personnel evaluation, *Knowledge-Based Systems* 52: 1–10.

[41] E.K. Zavadskas, Z. Turskis, J. Tamošaitiene, V. Marina (2008); Multicriteria selection of project managers by applying grey criteria, *Technological and Economic Development of Economy*, 14(4): 462–477.

[42] E.K. Zavadskas, Z. Turskis (2011); Multiple criteria decision making (MCDM) methods in economics: an overview. *Technological and economic development of economy*, 17(2): 397–427.

[43] E.K. Zavadskas, Z. Turskis, S. Kildienė (2014); State of art surveys of overviews on MCDM/MADM methods. *Technological and economic development of economy*, 20(1): 165–179.

[44] S. Zhang, S. Liu (2011); A GRA-based intuitionistic fuzzy multi-criteria group decision making method for personnel selection. *Expert Systems with Applications*, 38(9): 11401–11405.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):67-76, February 2016.

Predictive Input Delay Compensation with Grey Predictor for Networked Control System

A. Kuzu, S. Bogosyan, M. Gokasan

Ahmet Kuzu*
1. Istanbul Technical University
Electric-Electronics Faculty
Control and Automation Engineering Department
Turkey, 34469 Maslak, Istanbul
kuzuah@itu.edu.tr
2. TUBITAK BILGEM
Turkey, 41100 Gebze, Kocaeli
*Corresponding author: ahmet.kuzu@tubitak.gov.tr

Seta Bogosyan
ECE Department
University of Alaska Fairbanks, Fairbanks, USA
sbogosyan@alaska.edu

Metin Gokasan
Istanbul Technical University
Electric-Electronics Faculty
Control and Automation Engineering Department
Turkey, 34469 Maslak, Istanbul
gokasan@itu.edu.tr

Abstract: The performance of networked control systems is affected strictly by time delay. Most of the literature in the area handle the problem from a stability perspective. However, stability optimized algorithms alone are not sufficient to reduce synchronization problems caused by time delay between the action and reaction in geographically distant places, and the effect and performance of other system components should also be taken into account. In teleoperation applications the reference is often provided by a human, known as the operator, and due to the nature of the human system, references provided by the human operator are of a much lower bandwidth when compared to common control reference inputs. This paper focuses on the operator, and proposes an approach to predict the manipulator's motion (created by the operator) ahead of time with an aim to reduce the time delay between the master and slave manipulator trajectories. To highlight the improvement offered by the developed approach, hereby called Predictive Input Delay Compensator (PIDC), we compare the performance with the only other study in the literature that handles this problem using the Taylor Series approach. The performance of these two approaches is evaluated experimentally for the forward (control) path on a PUMA robot, manipulated by a human operator and it has been demonstrated that the efficient latency in the forward path is decreased by 100ms, on average, reducing the forward latency from 350ms to 250ms.

Keywords: communication network delay, delay regulator, Grey predictor, Taylor series, teleoperation.

1 Introduction

Because of their huge potential to contribute to human life in many different ways, teleoperation and bilateral control systems have been attracting significant interest in control and communication communities. Telesurgery using remote medical robots, exploration in hazardous environments using teleoperating robots could be good examples of such promising applications. The ultimate aim of a networked control system is the synchronization of the position and/or force between the master and slave in geographically distant motion control systems. Currently, the most widely used network communication medium is the Internet. However, the Internet brings a variable delay between the transmitted channels. This makes control implementations over the Internet challenging. The problem of variable can be eliminated via delay regulators [1] resulting in constant, and often relatively long delays.

Time delay compensation problem in bilateral control systems has been addressed by many different approaches. To name a few of the major methods, scattering variables [2], wave variables [3], Smith-Predictor (SP) [4], Astrom's modified SP [5], sliding-mode control (SMC) [6] and , via the design of communication disturbance observer (CDOB) [7]. Moreover optimal control methods are used to find an optimal solution in terms of stability and performance constraints of the system [8].

The focus of this study will again be on the performance improvement of master-slave position tracking in networked robot control systems. While the ultimate goal of the networked control is full synchronization, network delay between master and slave is a major obstacle for the desired performance, and network delays happen randomly. In teleoperation applications the reference is often provided by a human, known as the operator. Due to the nature of the human system, references provided by the human operator are of a much lower bandwidth when compared to common control reference inputs, and this can sometimes be problematic. All of the above mentioned studies discuss system stabilization under network delay [9], but do not address the operator delay, which also contributes to the delay between master and slave. Meanwhile, the prediction of the input delay (in this case, created by the human operator) has the potential to reduce this network latency. However, to the authors' best knowledge, the only study in the literature addressing this concept is [9], which uses a Taylor series based analytical approach to handle this problem. Taylor series simply performs the extrapolation of position based on velocity, meanwhile acceleration has significant effect on both velocity and position, and affects the prediction error negatively.

In this study, we propose a method based on Grey Prediction for Predictive Input Delay Compensation, and demonstrate experimentally the advantages of the proposed method over the one using Taylor Series in predicting the operator's motion. The Grey prediction not only performs extrapolation, but unlike the Taylor Method fits a differential equation to the system dynamics. As a result, grey prediction is more effective in considering the transients, hence, the acceleration.

The organization of the paper is as follows. Section II presents the benchmark Taylor Series based predictor. Section III presents Grey Prediction. Section IV introduce the Networking Control System and the application of the both two predictor on it. Section V introduce the experimental setup and results with conclusions in Section VI.

2 Taylor based PIDC as Benchmark System

In this section, we will first discuss the benchmark PIDC approach based on Taylor Series. Subsystems of the human, such as skeleton, muscle, and neural systems behave similar to mass-spring-damper like structures, hence result in a high time constant for the operator. This also makes it acceptable to assume the human motion output to be continuously differentiable in time [9].

By accepting this assumption, future signal values can be predicted using simple geometric approaches. The prediction formula is

$$\lim_{T \to 0} q_{mi}(n+H) \approx (H+1)q_{mi}(n) - Hq_{mi}(n-1) \tag{1}$$

Here $q_{mi}(n+H)$ denotes H step further value , $q_{mi}(n)$ denotes current value and finally $q_{mi}(n-1)$ denotes previous value. We must mention that, there are just two error source which is neglected. One is high order terms and the second is discretization.

3 Grey based PIDC

Grey system theory [10] is developed for systems characterized by uncertain information. Grey Prediction is a scientific quantitive prediction method which is based on the theoretical treatment of the original data to determine the future output of the system [11]. Basically, it can be defined as a local curve fitting extrapolation method, which requires four data sets only. In Grey Systems, $GM(n,m)$ denotes a grey model. Here n denotes the order of the difference equation, and m is the number of the variables. The commonly used Grey Model is of the $GM(1,1)$ type. It represents the first order derivative, and one input variable is used for prediction purposes. The process of the Grey Prediction can be given as below [?]:

Step 1: Collecting the original data sequence, and using generalized coordinate,

$$q_{mi}^{(0)} = \left\{ q_{mi}^{(0)}(1), q_{mi}^{(0)}(2), \ldots, q_{mi}^{(0)}(N) \right\}, \; N \geq 4 \tag{2}$$

where

$$q_{mi}^{(0)}(N) = q_{mi}(n) \tag{3}$$
$$q_{mi}^{(0)}(N-1) = q_{mi}(n-1) \tag{4}$$
$$\ldots$$
$$q_{mi}^{(0)}(1) = q_{mi}(n-N+1) \tag{5}$$
$$\tag{6}$$

Here N denotes buffer size, q denotes generalized coordinate, m subscript denotes master side not slave side, i subscript denotes i^{th} joint angle. For instance q_{m3} denotes master manipulators 3^{rd} joint angle. Moreover, $q_{mi}^{(0)}$ denotes zero order AGO of q_{mi}.

Step 2: Conducting an accumulated generation operation, AGO, on the original data sequence in order to diminish the effect of data uncertainty;

$$q_{mi}^{(1)} = \left\{ q_{mi}^{(1)}(1), q_{mi}^{(1)}(2), \ldots, q_{mi}^{(1)}(N) \right\}, \; N \geq 4 \tag{7}$$

Where

$$q_{mi}^{(1)}(k) = \sum_{i=1}^{k} q_{mi}^{(0)}(i), k = 1, 2, \ldots, N \tag{8}$$

Here the $q_{mi}^{(1)}$ denotes first order AGO of q_{mi}.

Step 3: Establishing the Grey difference equation and then calculating its background values;

$$q_{mi}^{(0)}(k) = -a_i {z_i}^1(k) + b_i \tag{9}$$

$$ {z_i}^{(1)}(k) = 0.5 \left\{ q_{mi}^{(1)}(k) + q_{mi}^{(1)}(k-1) \right\} \tag{10}$$

Here the a_i denotes developing coefficient in Grey Theory, b_i denotes Grey input.
Step 4: Constructing data matrix B and data vector Y;

$$B = \begin{bmatrix} -{z_i}^{(1)}(2) & 1 \\ -{z_i}^{(1)}(3) & 1 \\ \vdots & \vdots \\ -{z_i}^{(1)}(N) & 1 \end{bmatrix} \tag{11}$$

$$Y_i = \left[q_{mi}^{(0)}(2), q_{mi}^{(0)}(3), \ldots, q_{mi}^{(0)}(N)\right]^T \tag{12}$$

Step 5: Resolving the matrix;

$$Y_i = B_i \hat{a}_i \tag{13}$$

$$\hat{a}_i = B_i^T B_i^{-1} B_i^T Y_i = \begin{bmatrix} a_i \\ b_i \end{bmatrix} \tag{14}$$

Step 6: Deriving the solution to the Grey difference equation;

$$q_{mi}^{(1)}(k+1) = \left[q_{mi}^{(0)}(1) - \frac{b_i}{a_i}\right] e^{-a_i k} + \frac{b_i}{a_i} \tag{15}$$

Step 7: Conducting the inverse accumulated generation operation to obtain a prediction value

$$q_{mi}^{(0)}(k+1) = \left[q_{mi}^{(0)}(1) - \frac{b_i}{a_i}\right] e^{-a_i k}(1 - e^{a_i}) \tag{16}$$

Step 8: By Substituting k with $N + H - 1$

$$q_{mi}^{(0)}(N + \mathrm{H}) = \left[q_{mi}^{(0)}(1) - \frac{b_i}{a_i}\right] e^{-a_i(N+H-2)}(1 - e^{a_i}) \tag{17}$$

Step 9: By Rearranging formula

$$q_{mi}(n + \mathrm{H}) = \left[q_{mi}(n - N + 1) - \frac{b_i}{a_i}\right] e^{-a_i(N+H-2)}(1 - e^{a_i}) \tag{18}$$

4 A Configuration for the Networked Control System

Here we will introduce our proposed networked control system configuration to explain the requirement and the performance measure of the Predictive Input Delay Compensator ($PIDC$) algorithms. However first, we will introduce our standard configuration without the $PIDC$. In the standard configuration, the operator forces the master manipulator to a desired posture, which in turn will dictate the slave motion. In order for the slave to track the master motion in the closest possible way, on the master side, an Astrom Smith Predictor (ASP) generates the control signal for the model plant. Then the control signal generated on the master side, is transmitted to the slave side passing through a Delay Regulator Send unit (DRS_m)" through the Internet to Delay Regulator Receive unit (DRR_s). On the slave side, a Model Tracking Control (MTC) algorithm inputs the received control to an other model process (same as the model plant at master side) and forces the slave manipulator to track the trajectory of the model plant.

The angular displacement output of the MTC is fed back to the ASP passing through a Delay Regulator Send unit (DRS_s) through the Internet to Delay Regulator Receive unit (DRR_m). [1]

Here $\tau_{oq\{1,2,3\}}$ denote the joint torques generated by the operator, $\tau_{qm\{1,2,3\}}$ denote the joint torques applied to the manipulator after the addition of $g_{qm\{1,2,3\}}$ gravitational compensation terms. $\tau_{mq\{1,2,3\}}$ denote the torque signals fed to DRS_m to be sent to slave side. $\bar{\tau}_{mq\{1,2,3\}}$ denote the delay regulated torque signals coming through the Internet from the master to the slave. $\tau_{cq\{1,2,3\}}$ denote the joint torques generated by MTC, $\tau_{qs\{1,2,3\}}$ denote the joint torques applied to the manipulator after the addition of $g_{qs\{1,2,3\}}$ gravitational compensation terms. Finally $q_{s\{1,2,3\}}$ denote the slave manipulator's joint angle (actual) positions.

In this study, the Predictive Input Delay Compensator ($PIDC$) unit is added to this configuration between the operator and ASP. With this addition, it is now possible to predict and compensate the delay caused by the operator, which will allow sending the master side information to the slave side with less delay. This reduced delay increases the synchronization between the master and slave trajectories. To reduce nonlinearities of the master and slave manipulators, namely to increase the compliance on the master side and compensate the gravity on the slave side, gravity compensation blocks GC_m and GC_s added, respectively. The overall master-slave architecture is given in Fig. 1. However the function detail of each block is outside the scope of this study, and will not be discussed.

5 Experimental System and Results

Next, experiments have been performed to conduct a comparative performance evaluation for the Taylor based and Grey prediction approaches. The 6-DOF PUMA560 Industrial Robot is used for experimentation. The manipulator is operated as a 3-DOF system by the operator as can be seen in Fig. 2.

The well-known Euler-Lagrange based dynmaic model of the manipulator has the following general form:

$$M(q).\ddot{q} + V(q, \dot{q}).\dot{q} + G(q) = \Gamma \tag{19}$$

where,

q	: $nx1$ position vector
$M(q)$	: nxn inertia matrix of the manipulator
$V(q, \dot{q})$	: $nx1$ vector of Centrifugal and Coriolis terms
$G(q)$	: $nx1$ vector of gravity terms
Γ	: $nx1$ vector of torques

In this study the system will be taken on the consideration as an independent joint control system. This approach allows each manipulator joint to be controlled independently as a SISO system, with the nonlinearities and couplings taken as a disturbance affecting each joint actuator.

For our experimental system, since the human speed is considerably low, the main nonlinearities come from the gravity effect $G(q)$. For that reason, in our experiments we apply system gravity compensation $\tilde{G}(q) \approx G(q)$, to cancel and/or reduce the gravity effect on the experimental system to a negligible level.

The use of an independent joint control system approach, simplifies the system to be estimated to a one-degree-of-freedom process.

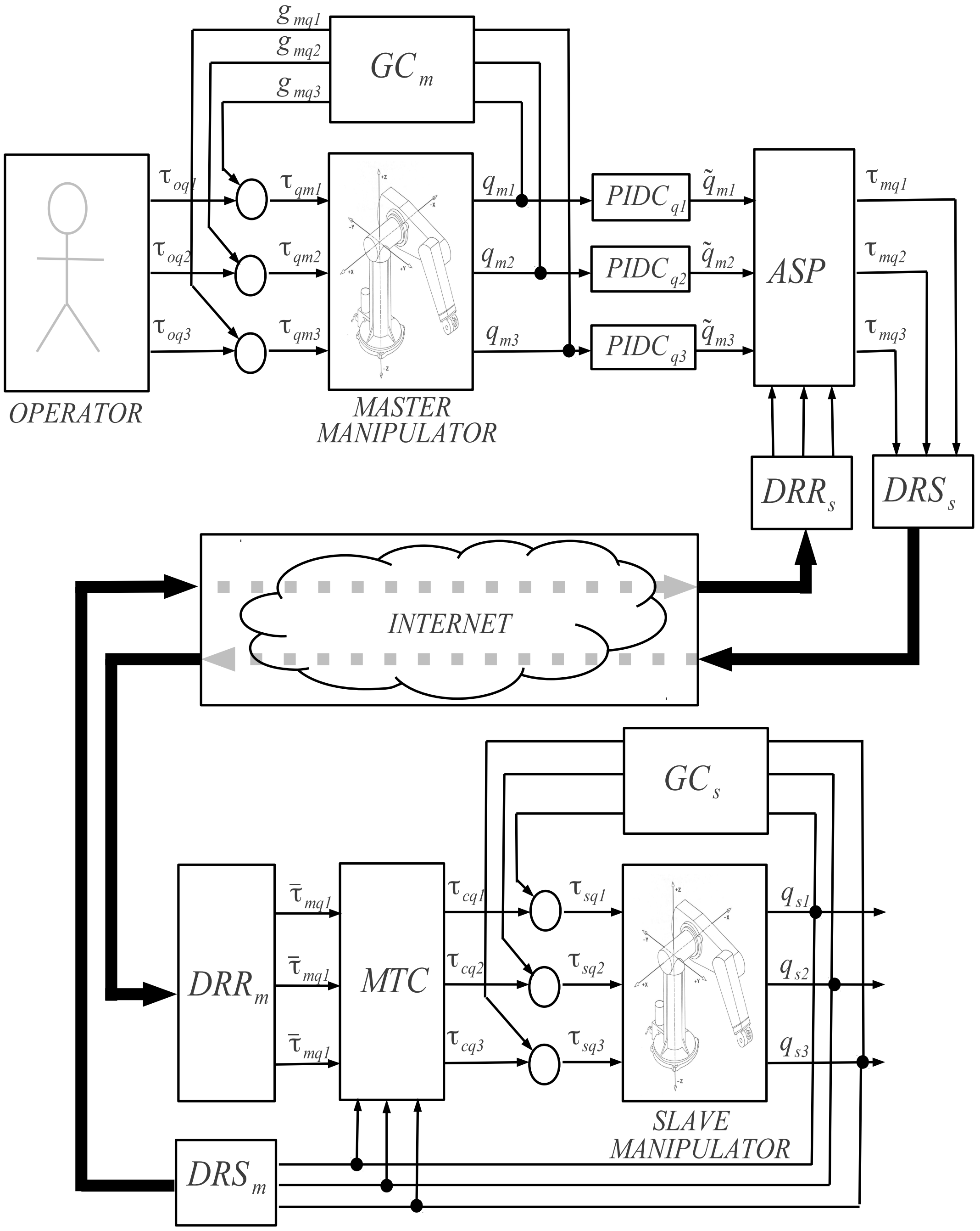


Figure 1: Extended Control Scheme of Networked Control

Experiments are performed for three cases. Then, $\tilde{q}_{mi}$ is also evaluated for three separated case. These cases and evaluation formulas are;

Case 1: No PIDC

$$\tilde{q}_{mi}(n) = q_{mi}(n) \tag{20}$$

Case 2: Taylor Series based Benchmark PIDC

$$\tilde{q}_{mi}(n) = (H+1)q_{mi}(n) - Hq_{mi}(n-1) \tag{21}$$

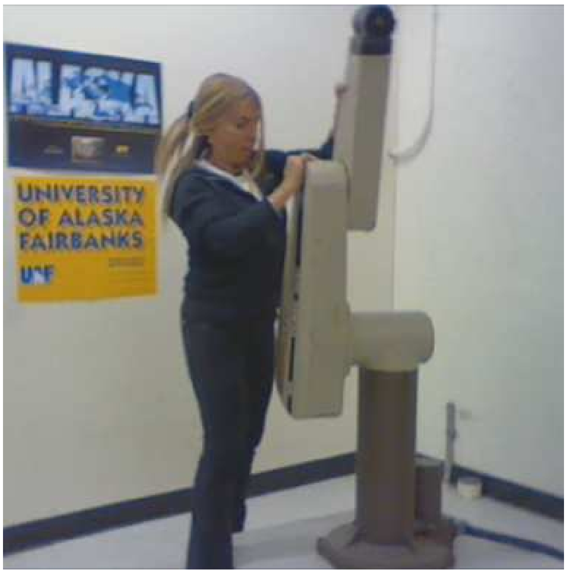
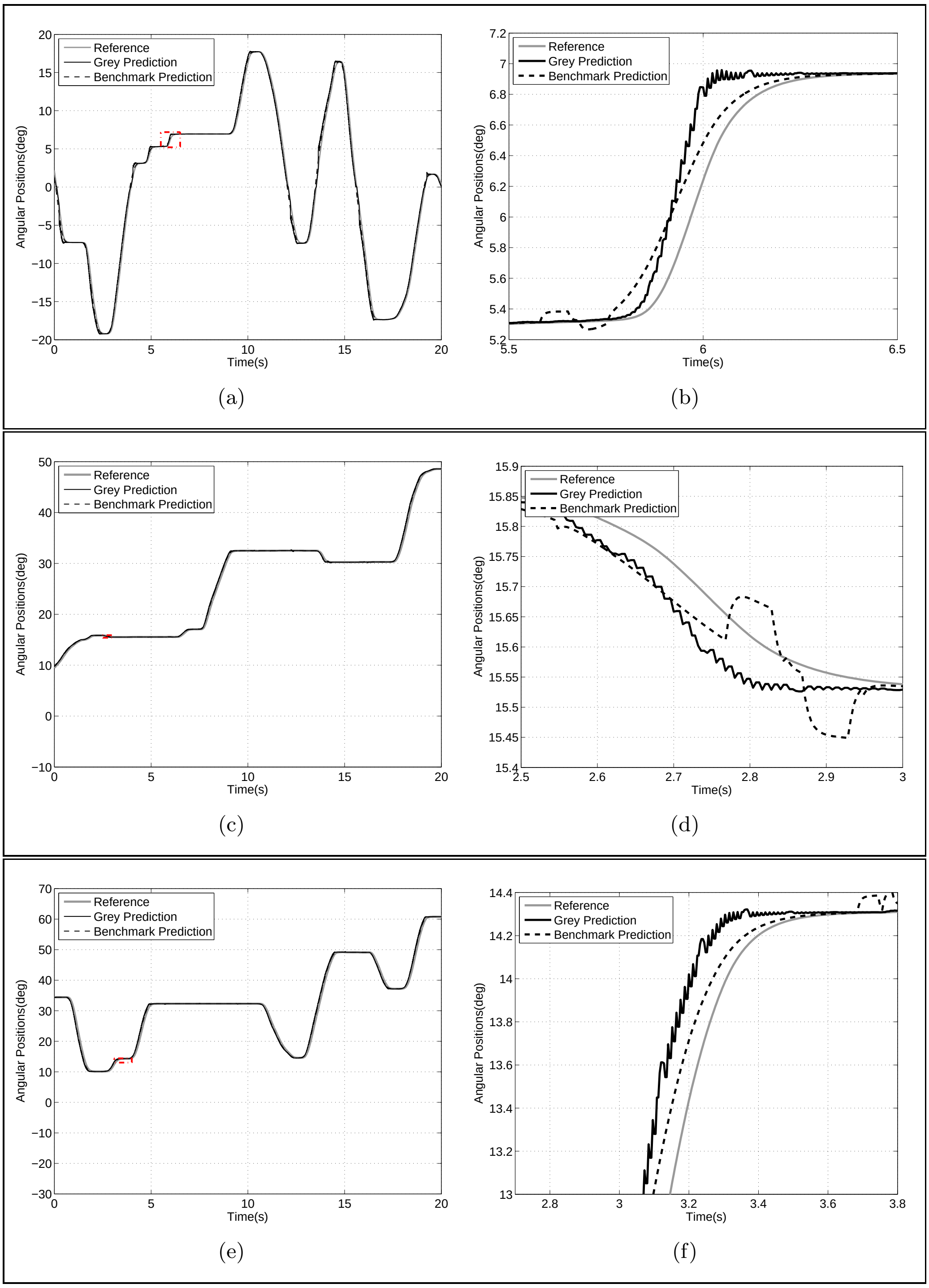


Figure 2: Experimental Setup

Case 3: Grey Theory based Proposed PIDC

$$\tilde{q}_{mi}(n) = \left[q_{mi}\left(\mathrm{n-N+1}\right) - \frac{b_i}{a_i} \right] e^{-a_i(n+H-1)}(1-e^{a_i}) \tag{22}$$

The results of those experiments are seen in Fig. 3. Here, Fig. 3a and Fig. 3b depict the results for the first joint , Fig. 3c and Fig. 3d for the second joint, Fig. 3e and Fig. Fig. 3f for the third joint. The performance of each joint is further demonstrated by also highlighting the zoomed version of the region marked in red. The figures in the right are zoomed versions for the highlighted sections in the diagrams on the left side. For each figure, the grey line represents the operator motion, which is taken as the reference motion to be predicted. The solid line depicts the Grey Predictor's output, and finally dashed line demonstrates the output of the benchmark Taylor Series based predictor. In each figure, it is easily seen that when the angular velocity is low, both algorithms demonstrate similar performance. However when the acceleration increases, the performances show differences. Only operation intervals where there is significant operator motion have been selected in the zooms of Fig. 3b, Fig. 3d and Fig. 3f. In all three figures, we see that the Grey Prediction method achieves a faster prediction of $100ms$ on average when compared with the Taylor based approach. On the other hand, the benchmark method demonstrates a prediction performance that varies between $10ms$ and $100ms$, and demonstrates a poor performance in tracking transients as indicated by the high amplitude oscillation observed in Fig. 3b starting at $5.7ms$, and in Fig. 3d starting at $2.8ms$ for the benchmark system output. Hence, it can be said that the proposed Grey based $PIDC$ demonstrates a faster and more accurate prediction performance then the Taylor based $PIDC$.



(a) (b)

(c) (d)

(e) (f)

Figure 3: Measured and Predicted Angular Positions of each joint and zoomed versions a) joint 1, b) zoomed area of joint 1, c) joint 2, d) zoomed area of joint 2, e) joint 3, f) zoomed area of joint 3

6 Conclusion

In this study, a Grey system theory based *PIDC* is developed and implemented for the prediction of the master manipulator motion in order to reduce the transmission latency between the master and slave. Our philosophy is to reduce the latency in every way possible within our capability, considering network latency is unavoidable and random.

Experiments are conducted on a PUMA 560 manipulator which is just compensated for gravitational force to allow easy manipulation for the operator. The operator randomly manipulates the arm, while both the benchmark and proposed schemes predict the future trajectory of the robot motion created by the operator. The proposed approach outperforms the Taylor Series based benchmark approach, by predicting the joint motions approximately $100ms$ ahead on average, while the benchmark's predictor performance varies between $2.8ms$-$100ms$. Based on these results, it can be concluded that Grey Prediction meets our motion prediction requirements better then the Taylor Series based approach, which is currently the only other study in the literature to address input delay compensation.

Acknowledgment

This study has been supported by National Science Foundation-Computer & Information Sciences and Engineering (NSF-CISE) , and NSF Office of International Science & Engineering (NSF-OISE). The authors declare that there is no conflict of interests regarding the publication of this article.

Bibliography

[1] Kuzu, A. et al (2011); Control and measurement delay compensation in bilateral position control, *Mechatronics (ICM), 2011 IEEE International Conference on*, April 2011, 1003–1010.

[2] Mori, T (2012); Modified bilateral control by using intervention impedance based on passivity of flexible master-slave manipulators and its design methods, *Control, Automation and Systems (ICCAS), 2012 12th International Conference on*, 748–753.

[3] Hashemzadeh, F (2012); A new method for bilateral teleoperation passivity under varying time delays, *Mathematical Problems in Engineering*, Hindawi Publishing Corporation, http://dx.doi.org/10.1155/2012/792057.

[4] Lai, C.L. and Hsu, P.-L. (2010); Design the remote control system with the time-delay estimator and the adaptive smith predictor, *Industrial Informatics, IEEE Transactions on*, 6(1): 73–80.

[5] Astrom, K. (1994); A new smith predictor for controlling a process with an integrator and long dead-time, *Automatic Control, IEEE Transactions on*, 39(2): 343–345.

[6] Gadamsetty, B. et al (2010); Sliding mode and ekf observers for communication delay compensation in bilateral control systems, *Industrial Electronics (ISIE), 2010 IEEE International Symposium on*, 328–333.

[7] Natori, K. et al (2010); Time-delay compensation by communication disturbance observer for bilateral teleoperation under time-varying delay, *Industrial Electronics, IEEE Transactions on*, 57(3): 1050–1062.

[8] Yashiro, D. and Ohnishi, K. (2008); L2 stable four-channel control architecture for bilateral teleoperation with time delay, *10th IEEE International Workshop on Advanced Motion Control*, 324–329.

[9] Baran, E. and Sabanovic, A. (2012); Predictive input delay compensation for motion control systems, *Advanced Motion Control (AMC), 2012 12th IEEE International Workshop on*, March 2012, 1–6.

[10] Ju-Long, D. (1982); Control problems of grey systems, *Systems & Control Letters*, 1(5): 288–294.

[11] Liu, S. and Lin, Y. (2006); *Grey information: theory and practical applications*, Springer.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):77-89, February 2016.

Towards Pricing Mechanisms for Delay Tolerant Services

L. Marentes, T. Wolf, A. Nagurney, Y. Donoso

Luis Marentes
Department of Systems and Computing Engineering
Universidad de los Andes, Bogota, Colombia, South America
la.marentes455@uniandes.edu.co
*Corresponding author

Tilman Wolf
Department of Electrical and Computer Engineering
University of Massachusetts, Amherst
Amherst, MA, United States
wolf@ecs.umass.edu

Anna Nagurney
Operations and Information Management, University of Massachusetts, Amherst
Amherst, MA, United States
nagurney@isenberg.umass.edu

Yezid Donoso
Department of Systems and Computing Engineering
Universidad de los Andes, Bogota, Colombia, South America
ydonoso@uniandes.edu.co

Abstract: One of the applications of Delay Tolerant Networking (DTN) is rural networks. For this application researchers have argued benefits on lowering costs and overcoming challenging conditions under which, for instance, protocols such as TCP/IP cannot work because their underlying requisites are not satisfied. New responses are required in order to understand the true adoption opportunities of this technology. Constraints in service level agreements and viable alternative pricing schemes are some of the new issues that arise as a consequence of the particular operation mode. In this paper, we propose a novel model for pricing delay tolerant services, which adjusts prices to demand variability subject to constraints imposed by the DTN operation. With this model we also show how important parameters such as channel rental costs, cycle times of providers, and market sensitivities affect business opportunities of operators.

Keywords: Delay Tolerant Networks Pricing, Capacity Management, Economic Models.

1 Introduction

Networks deployed in low income isolated areas operate in challenging environments characterized by discontinuous power supply and long distance wireless link connections. Their architectures are adapted to reduce costs, but they are prone to fail due to severe bandwidth restrictions, an operation highly dependent on weather conditions, and low levels of redundancy on key elements. These facts lead to intermittent connectivity between end to end points, which breaks the main assumption for the proper operation of the TCP/IP protocol suite.

The research networking community introduced the concept of intermittent connected networks (ICN) to cope with this environment. According to [15], ICN can be defined as:

> An infrastructure-less wireless network that supports the proper functionality of one or several wireless applications operating in stressful environments, where excessive delays and unguaranteed continuous existence of end-to-end path(s) between any arbitrary source-destination pair, result from highly repetitive link disruptions.

The core element is the uncertainty in connectivity partially solved by two mechanisms: store-and-forward and flooding. In the first mechanism, the network's nodes are capable of storing data during periods of disconnection and forwarding to other nodes closer to the destination when a link is established. In the second mechanism, flooding copes with the routing problem. As nodes cannot maintain origin-destination paths, the network is programmed to send many copies of the same data towards different nodes to increase the probability of reaching the final destination.

For isolated networks the first project introducing this concept was [19] with a two-fold objective: to reduce operational cost using offline data mule connections implemented by existing transportation methods such as buses, motorcycles, bicycles, etc., and to provide an infrastructure capable of overcoming interruptions. Then the idea was extended to use mechanic backhauls and the Delay Tolerant Network architecture (DTN) proposed by the Internet Engineering Task Force (IETF) to handle ICNs [4].

In recent years prototypes such as [5, 9] using DTN or similar architectures have appeared. Potential operators deploying networks in isolated regions would fulfil a mix of delay tolerant and real time services such as video conferencing for medical procedures. We envision that real time services are going to be operated using technologies resulting from projects deploying low-cost infrastructure for cellular networks, see [1,21], or by wireless channels connected to infrastructure provided by states, i.e, the e-Mexico system [23], the National broadband plan in India [24], the broadband infraCo in South Africa [22], to name a few.

Although most of these proposals cover technical parts of the network, the economics of the services being fulfilled are not considered. Therefore, their sustainability is strongly compromised. In economic terms, providers establish service level agreements (SLAs) to offer alternatives regarding the quality of services and prices that customers are willing to pay. Right now, these agreements are based on measures including, but not limited to, packet loss ratio, connection latency and bandwidth. Once the agreement is established the provider charges services according to the rules defined in the SLAs. Nonetheless, potential operators in isolated areas cannot directly use these measurements; they are based on the assumption of end to end connectivity which is not guaranteed in the rural environment. So one of the problems faced by these operators is how to establish SLAs for a diverse set of customers in networks delivering delay tolerant and real time services.

In this paper we propose, as a first step, an integrated pricing and resource assignment policy for a delay tolerant service. This proposal is based on two service agreement elements and it optimizes the expected profits from the operator's point of view. Furthermore, using real data from a rural network we test the validity of this model to adjust to variability on traffic demand and study its behavior under different scenarios.

The remainder of this paper is organized as follows. Section II provides an overview of previous proposals on Internet service usage pricing. Section III introduces a novel pricing model tailored to a delay tolerant service. This model is based on traffic management and average time of delivery. Scenario test-bed settings and results are presented in section III. Finally, section IV summarizes and concludes the paper.

2 Related Work

From the introduction of the Internet, authors like [10] have argued the necessity of creating and using charging models to induce an optimal network resource usage. For Internet connectivity in low income regions, which can be isolated, an optimal resource assignment is even more important, howsoever, there is no room for light users to subsidize heavy users, nor enough capacity to deliver non-priority services without affecting high-priority services, the profit enablers for operators [2]. To the best of our knowledge this paper is the first to introduce charging methods for networks using the delay tolerant architecture. Despite its novelty, the proposal takes elements of general charging models for the Internet and, in particular, it recaptures conclusions to increase operator performance under time-variance consumption patterns, see [11, 16, 20] for an overview of alternative research approaches.

Hande et al [8] introduce models to understand the relationship between the revenues of a monopolist Internet Service Provider(ISP) and the price sensitivity of users under a two part tariff. They conclude that in markets with high price sensitivity, normally found in low income regions, most of the revenues come from the usage based portion of the price. Even more, the paper suggests that the loss in revenue can be mitigated if the ISP implements charging based on consumption. Following these conclusions, the proposed model assumes that the ISP collects the usage part of the tariff and it has previous knowledge of the demand function based on the price sensitivity. This paper extends the formulation by including quality attributes on the demand function and costs. These extensions help to provide insights regarding ISP profitability and service quality experience.

Jian L. et al [12] introduce a price scheme to be implemented by a monopolist operator having a set of users, whose utilities not only depend on the traffic quantity, but on the specific access time. They show that the level of revenue extracted by the ISP depends on the information available. With perfect information, the ISP can extract the maximal revenue. When the ISP has only information about the traffic in different time-slots, a more realistic scenario, the loss in revenue compared with the maximal value is in general not bounded. Contrary to their proposal, we partially discard the additional utility received by users when access the network in a specific time-slot.

The present proposal utilizes a demand changing during the day and forecast prices. We expect that users being charged move to their preferred time-slot as they know pricing information, which somehow follows the results presented by [7]. For us, based on an actual deployment, the authors verify that repeating a game, in which users take into account net prices to decide either to wait or to use the network, helps not only to decrease congestion periods but to increase the network usage.

3 Proposed Pricing Model

Operators have two modes for connecting to the Internet. The first mode is a real time connection with cost and bandwidth capacity. Its cost is measured in value per unit of traffic volume and its bandwidth is measured in traffic volume per unit of time. The second mode is a mechanic backhaul connection with a by trip transit time (delivery time), cost, and capacity. Operators can divide traffic in both connections. Users receive in real time part of the content, for instance the web page's text, and, after the delivery time, the rest, images and videos. This way of operation is beneficial for users and providers. Users consume a reduced service instead of no service and providers increase the use of the network.

Information is relevant to users during a certain period of time. This is closely related to the value given by users to services, which is partially determined by whether the operator might or not deliver the information within that period. Then, the delivery time becomes an intrinsic quality characteristic for delay tolerant services to be modelled as part of the demand.

In this proposal the idea of Grade of Service (GoS) was adapted to make quality operational for delay tolerant services. GoS is defined, see [6], as a set of traffic engineering variables used to measure the adequacy of a group of resources to fulfil a service under a specified condition. An important concept behind GoS is the average behavior, which is used to aggregate the performance of different resources into a unique measure. Following this idea, DTN operators are interested in measuring the adequacy of the resources (the two connection modes) for user's preferences on time of delivery. We propose that operators might use the weighted average time required to get answers to a user's request as a measure of the service's quality.

Our proposed model for pricing delay tolerant services includes price and average delivery time as factors determining aggregated demand. It creates an optimal policy for price and average delivery time searching for the maximum amount of profits. In the following subsections we present the main assumptions: the mathematical representation for the situation being modelled and the development of its solution.

3.1 Assumptions

Demand

In this paper demand constructions suggested by authors in [3] are used. They fixed the relationship between demand, price and time by the following linear function.

$$D(P,T) = a - \beta_P P - \beta_T T, \tag{1}$$

where a is the potential market size, β_P and β_T are the corresponding sensitivity of demand to price and to delivery time. The potential market size is generalized because demand fluctuates within the hours of days and between days. We propose a as a function of time, which is continuous and differentiable. The end form aggregated demand used in this proposal is:

$$D(t,P,T_{avg}) = a(t) - \beta_P P - \beta_T T_{avg} \tag{2}$$

Costs

The mechanic backhaul connection is charged by a third party. The vendor charges by the distance travelled and not by how much information is being travelled. From the operator's point of view this charge is a fixed cost by trip.

The real time connection is used for fulfil priority and non-priority services –delay tolerant services belong to the second category–, and it has a channel rental cost.

Constant capacity

The model assumes that connections cannot increase capacity dynamically as more demand is requested from users. The mechanic backhaul capacity is determined by the connection bandwidth and the total time that the vehicle is connected to the main gateway.

Constant consumption

Requests generated by users in the network are dissimilar in terms of data to be transferred. However, the operator needs to establish prices before observing the actual use of the network. The estimate of the average download and upload consumption is used to charge users. The estimate is an input parameter for the model and it is included in the potential market size function $a(t)$.

3.2 Mathematical formulation

The operator needs a rule that defines traffic going to be served by the mechanic backhaul connection ($F_1(t)$) or by the real time connection ($F_2(t)$) and the price to charge. See figure 1. Each connection has a common set of parameters defined by the tuple (T, C) where T means time of delivery, and C means cost. The real time connection has an instant capacity K_2. The mechanic backhaul connection has a by trip capacity CAP_{mb}. The operator wants to maximize profits while maintaining the service mean delivery time (T_{avg}). The operator cannot differentiate among customers willing to pay more for the time provision.

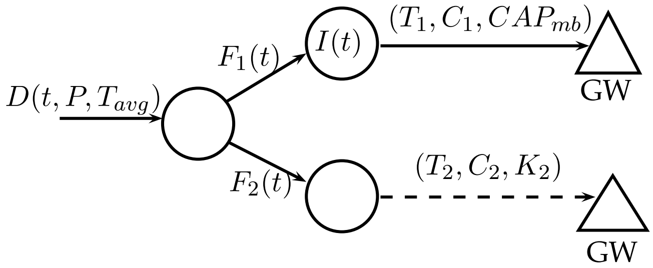


Figure 1: Elements in the general model

The general model could be stated as:

$$\text{Max} \int_0^{T_1} P(t)(F_1(t) + F_2(t)) - C_1F_1(t) - C_2F_2(t)dt$$

Subject to

$$I'(t) = -F_1(t) \tag{3a}$$
$$I(T_1) \geqslant 0 \tag{3b}$$
$$F_1(t) + F_2(t) = a(t) - \beta_T T_{avg} - \beta_P P(t) \tag{3c}$$
$$F_2(t) \leqslant K_2 \tag{3d}$$
$$(Z(t) - 1)F_1(t) \leqslant F_2(t) \tag{3e}$$
$$F_1(t) \geqslant 0 \tag{3f}$$
$$F_2(t) \geqslant 0 \tag{3g}$$
$$P(t) \geqslant 0 \tag{3h}$$
$$I(0) = CAP_{mb} \tag{3i}$$

The first constraint indicates that the instantaneous change in the backlog of traffic going to be served by the mechanic backhaul connection is equal to the flow using that channel. The second constraint establishes that the aggregate capacity for services using the mechanic backhaul connection cannot be more than the capacity available. In particular, this condition must be true at the end of the cycle time.

The third constraint establishes that demand must go by either of the channels. It constitutes the relationship between channel use and price and service mean delivery time. Constraint four indicates that the quantity of flow using the real time connection must be less than or equal to the capacity available.

Requests made at time t and using the mechanic backhaul must at least have a travel time equal to the vehicle cycle time T_1. Moreover, when the arrival time coincides with the vehicle being in transit, it has to wait the remaining cycle time and another cycle. Therefore, the traffic going by the mechanic backhaul as part of a request made at time t must to wait $2T_1 - t$.

Then, the mean delivery time for services being handled at time t is given by:

$$\overline{L}(t) = \frac{(2T_1 - t)U_1(t) + T_2(t)U_2(t)}{U_1(t) + U_2(t)} \tag{4}$$

We can assume that $T_2(t) = 0$ for all t. This measure according to the SLA maintained by the operator has to be less than or equal to T_{avg}. Defining the decreasing function $Z(t) = {}^{2T_1 - t}/_{T_{avg}}$ the expression is reduced to the following inequality which corresponds to constraint number five. The reader must note that it is valid only for the set $\{t \leq T_1 : Z(t) \geq 1\}$.

$$(Z(t) - 1)F_1(t) \leqslant F_2(t) \tag{5}$$

Finally, we define $b(t) = a(t) - \beta_T T_{avg}$ as the function representing the potential market at time t after discounting the sensitivity of users for the average time.

3.3 Optimal Policy

In this subsection we develop the optimal policy for problem (3). First, the optimal policy for the formulation without the time average constraint is developed; then as a second step, the policy is extended to include the average time constraint.

The optimal policy when providers can break the time average constraint

We introduce a new constraint in $F_1(t)$ in order to make easier the optimal policy develop. This constraint limits the amount of flow using the mechanic backhaul connection at any time, which is written as $F_1(t) \leq K_1$. Multiplier functions $\theta(t), \lambda_1(t), \lambda_2(t), \lambda_3(t), \lambda_4(t), \lambda_5(t), \lambda_6(t)$ are respectively associated with (3a), (3c), $F_1(t) \leq K_1$, (3d), (3f), (3g), (3h) to calculate the Hamiltonian (H) and Lagrangian (LH), which are, see [14]:

$$\begin{aligned}
f(t, I, F_1, F_2, P) &= P(t)(F_1(t) + F_2(t)) - C_1F_1(t) - C_2F_2(t) && \text{(6a)}\\
g(t, F_1) &= -F_1(t) && \text{(6b)}\\
h_1(t, P, F_1, F_2) &= b(t) - \beta_P P(t) - F_1(t) - F_2(t) && \text{(6c)}\\
h_2(t, F_1) &= K_1 - F_1(t) && \text{(6d)}\\
h_3(t, F_2) &= K_2 - F_2(t) && \text{(6e)}\\
h_4(t, F_1) &= F_1(t) && \text{(6f)}\\
h_5(t, F_2) &= F_2(t) && \text{(6g)}\\
h_6(t, P) &= P(t) && \text{(6h)}\\
H(t, I, F_1, F_2, P, \theta) &= f(.) + \theta(t)g(.) && \text{(6i)}\\
LH(t, I, F_1, F_2, P, \theta, \boldsymbol{\lambda}) &= f(.) + \theta(t)g(.) + \boldsymbol{\lambda h} && \text{(6j)}
\end{aligned}$$

from which we obtain the necessary conditions for optimality:

$$LH_P = F_1(t) + F_2(t) - \lambda_1(t)\beta_P + \lambda_6(t) = 0, \tag{7a}$$
$$LH_{F_1} = P(t) - C_1 - \theta(t) - \lambda_1(t) - \lambda_2(t) + \lambda_4(t) = 0, \tag{7b}$$
$$LH_{F_2} = P(t) - C_2 - \lambda_1(t) - \lambda_3(t) + \lambda_5(t) = 0, \tag{7c}$$
$$LH_{\lambda_1} = b(t) - \beta_P P(t) - F_1(t) - F_2(t) = 0, \tag{7d}$$
$$\theta'(t) = -H_I = 0, \quad I'(t) = -F_1(t), \qquad \theta(T_1) \geq 0, \theta(T_1)I(T_1) = 0, \tag{7e}$$
$$\lambda_4(t) \geq 0, \lambda_4(t)F_1(t) = 0, \quad \lambda_5(t) \geq 0, \lambda_5(t)F_2(t) = 0, \; \lambda_6(t) \geq 0, \lambda_6(t)P(t) = 0, \tag{7f}$$
$$\lambda_2(t) \geq 0, \lambda_2(t)(K_1 - F_1(t)) = 0, \quad \lambda_3(t) \geq 0, \lambda_3(t)(K_2 - F_2(t)) = 0, \tag{7g}$$

From (7) and integrating $\theta'(t) = 0$ we have $\theta(t) = \theta_0$. Without loss of generality we are going to restrict the optimal path to prices greater than zero, so $\lambda_6(t) = 0$. The reader must to observe:

1. Assuming $C_1 < C_2$, the unconstrained problem has as optimal price $P_1^*(t) = \frac{b(t) + (C_1 + \theta_0)\beta_P}{2\beta_P}$. It is important to note that the function $f(t)$ is strictly concave in the parameter P, $f_{PP} = -2\beta_P < 0$, and it is concave in the parameter F_1, Therefore, this value is a global optimum. Let $\overline{D}_1(t) = \frac{b(t) - (C_1 + \theta_0)\beta_P}{2}$ the demand corresponding to $P_1^*(t)$.

2. Because the mechanic backhaul channel has lower cost ($C_1 < C_2$), the optimal solution uses it as much as possible until the point $\overline{D}_1(t)$. Indeed, if $K_1 > 0 \Rightarrow F_1(t) > 0$.

Now, we calculate the optimal paths as functions of the maximal instantaneous capacity for the mechanic backhaul K_1. All cases are developed in the following paragraphs.

Case 1. $K_1 > \overline{D}_1(t)$. The constraint on the mechanic backhaul traffic is not binding. From observation two we have: $F_2^*(t) = 0, \lambda_2^*(t) = 0, \lambda_3^*(t) = 0$, and $\lambda_4^*(t) = 0$. Replacing these results on the set of equations 7, we can calculate the remaining optimal paths as: $F_1^*(t) = \frac{b(t) - (C_1 + \theta_0)\beta_P}{2}, \lambda_1^*(t) = \frac{b(t) - (C_1 + \theta_0)\beta_P}{2\beta_P}$, and $\lambda_5^*(t) = C_2 - C_1 - \theta_0$.

Case 2. $K_1 \leq \overline{D}_1(t)$ and $F_2^*(t) = 0$. From these conditions and observation two we conclude $\lambda_3^*(t) = 0$ and $\lambda_4^*(t) = 0$. These let us to find optimal values for the rest of the functions. Those values are: $P_2^*(t) = \frac{b(t) - K_1}{\beta_P}, \lambda_1^*(t) = \frac{K_1}{\beta_P}, \lambda_2^*(t) = \frac{b(t) - 2K_1 - (C_1 + \theta_0)\beta_P}{\beta_P}, \lambda_5^*(t) = \frac{2K_1 + C_2\beta_P - b(t)}{\beta_P}$.

Case 3. $K_1 \leq \overline{D}_1(t)$ and $F_2^*(t) > 0$. In this case $P_3^*(t) = \frac{b(t) + C_2\beta_P}{2\beta_P}, F_1^*(t) = K_1, F_2^*(t) = \frac{b(t) - C_2\beta_P - 2K_1}{2}, \lambda_1^*(t) = \frac{b(t) - C_2\beta_P}{2\beta_P}, \lambda_2^*(t) = C_2 - C_1 - \theta_0, \lambda_3^*(t) = 0, \lambda_4^*(t) = 0, \lambda_5^*(t) = 0$. Additionally, it is defined $\overline{D}_2(t) = \frac{b(t) - C_2\beta_P}{2}$.

Case 4. $K_1 + K_2 \leq \overline{D}_2(t)$. From observation one, the objective function concavity in $P(t)$ for a given t, and the fact that it is increasing in $F_1(t)$ and $F_2(t)$ until the point $\overline{D}_2(t)$, it can be concluded as the optimal decision to assign K_1 and K_2 to $F_1(t)$ and $F_2(t)$ respectively. With these assignments we have $\lambda_4^*(t) = 0, \lambda_5^*(t) = 0$. Replacing this information on the set of equations, we have $P_4^*(t) = \frac{b(t) - K_1 - K_2}{\beta_P}, \lambda_1^*(t) = \frac{K_1 + K_2}{\beta_P}, \lambda_2^*(t) = \frac{b(t) - 2(K_1 + K_2) - (C_1 + \theta_0)\beta_P}{\beta_P}, \lambda_3^*(t) = \frac{b(t) - 2(K_1 + K_2) - C_2\beta_P}{\beta_P}$.

Case 5. $K_1 = 0, F_2(t) > 0, K_2 \leqslant \overline{D}_2(t)$. The objective function in t is increasing in $F_2(t)$ as long as $K_2 \leqslant \overline{D}_2(t)$. Therefore, $F_2^*(t) = K_2, \lambda_5^*(t) = 0$. From $K_1 = 0$, we assign to the multiplier function $\lambda_4^*(t) = 0$. Replacing these results in the set of equations and resolving for the rest of functions, we have: $P_5^*(t) = \frac{b(t) - K_2}{\beta_P}, \lambda_1^*(t) = \frac{K_2}{\beta_P}, \lambda_2^*(t) = \frac{b(t) - 2K_2 - (C_1 + \theta_0)\beta_P}{\beta_P}, \lambda_3^*(t) = \frac{b(t) - 2K_2 - C_2\beta_P}{\beta_P}$.

Case 6. $K_1 = 0, F_2(t) > 0, K_2 > \overline{D}_2(t)$. From these conditions, then $\lambda_3^*(t) = 0, \lambda_4^*(t) = 0$, and $\lambda_5^*(t) = 0$. Additionally, replacing these conditions, the optimal paths are: $P_6^*(t) = \frac{b(t) + C_2\beta_P}{2\beta_P}, \lambda_1^*(t) = \frac{b(t) - C_2\beta_P}{2\beta_P}, \lambda_2^*(t) = C_2 - C_1 - \theta_0$.

The optimal θ_0 value is required for all cases, from sub-equation (3a) it is equal to: (where χ represents the indicator function).

$$I(T_1) = \int_0^{T_1} -F_1(t)dt = \int_0^{T_1} -K_1\chi_{K_1 \leq \overline{D}_1}dt + \int_0^{T_1} -\overline{D}_1\chi_{K_1 > \overline{D}_1} \tag{8}$$

Let $A_D = \int_0^{T_1} b(t)dt$ the total estimated market during the cycle time. From observation two, we conclude that operators want to send the maximal value by the mechanic backhaul channel. In case of enough mechanic backhaul capacity, i.e, $A_D - C_1\beta_P T_1/2 \leq CAP_{mb}$, it must be assigned $\theta_0 = 0$ and $K_1 = \max\{b(t) - C_1\beta_P/2 : t \in [0, T_1]\}$. When $A_D - C_1\beta_P T_1/2 > CAP_{mb}$, the constant value θ_0 can be obtained observing:

1. It can be verified that profits for cases 1 and 2 are greater or equal than profits for case 3, whenever $0 \leq \theta_0 \leq C_2 - C_1$.

2. The amount of flow not sent because not enough mechanic backhaul capacity is $\theta_0\beta_P T_1/2$. This issue is the result of using the optimal point $b(t) - (C_1 + \theta_0)\beta_P/2$ instead of $b(t) - C_1\beta_P/2$.

3. From items 1 and 2, we conclude that the flow potentially lost is constrained in $[0, (C_2 - C_1)\beta_P T_1/2]$. If $\frac{A_D - C_2\beta_P T_1}{2} \leq CAP_{mb} \leq \frac{A_D - C_1\beta_P T_1}{2}$, then $\theta_0 = \frac{A_D - 2CAP_{mb}}{\beta_P T_1} - C_1$ and $K_1 = \max\{\frac{b(t) - C_1\beta_P}{2} : t \in [0, T_1]\}$. If $CAP_{mb} < \frac{(A_D - C_2\beta_P T_1)}{2}$, then $\theta_0 = C_2 - C_1$ and $K_1 = \frac{CAP_{mb}}{T_1}$ must be assigned.

The optimal policy for providers with the time average constraint

We use the Lagrangian LH defined in the previous section and include the constraint $h_4 = F_2(t) - (Z(t) - 1)F_1(t)$. Again, a continuous multiplier function $\lambda_7(t)$ is associated with this constraint. The necessary optimality conditions for the modified Lagrangian is given by 7 and:

$$\lambda_7(t) \geq 0, \quad \lambda_7(t)(F_2(t) - (Z(t) - 1)F_1(t)) = 0 \tag{9}$$

Constraint (3e) is active when $F_2(t) = (Z(t) - 1)F_1(t), Z(t) \geq 1$ and $F_2(t)$ is constrained by K_2. Then combining constraints (3e) and (3d) we have $K_2 \geq (Z(t) - 1)F_1(t)$. From this inequality and $K_1 \geq F_1(t)$, we conclude that $F_1(t)$ is constrained by $\frac{K_2}{Z(t)-1}$ as long as this value is less than K_1. Moreover, this value is the binding constraint for $F_1(t)$ between the time 0 and time t^1, defined as:

$$t^1 = \max\left(0, 2T_1 - T_{avg}\left(\frac{K_2}{K_1} + 1\right)\right) \tag{10}$$

Now, we complete the optimal policy as a function of the instantaneous capacities K_1 and K_2. If the constraint (3e) is active and $t \in [0, t^1]$, then the optimal policy is the result of the capacity K_2 and the value $\overline{D}_3(t) = \frac{b(t)Z(t) - \overline{C}_1\beta_P - C_2(Z(t)-1)\beta_P}{2Z(t)}$ is attained, which is the maximum profit. See table 1, where $\overline{C}_1 = C_1 + \theta_0$. For $t \in [t^1, t^2 = 2T_1 - T_{avg}]$ the optimal policy is governed by values K_1 and $\overline{D}_3(t)$, see table 1. Finally, for $t > t^2$ the optimal policy follows the results presented in the previous subsection.

Table 1: Optimal assignment policy for $t \in [0, t^2]$

Case	t	$G(t)^*$	$P(t)^*$	$F_1(t)^*$	$F_2(t)^*$	$\overline{D}(t)$
$\frac{K_2 Z(t)}{Z(t)-1} \geq \overline{D}_3(t)$	$t \leq t^1$	$\frac{(Z(t)-1)(b(t)Z(t)-\beta_P(\overline{C}_1+C_2(Z(t)-1)))}{2Z(t)^2}$	$\frac{b(t)}{\beta_P} - \frac{G^* Z(t)}{\beta_P(Z(t)-1)}$	$\frac{G^*}{Z(t)-1}$	G^*	$\overline{D}_3(t)$
$\frac{K_2 Z(t)}{Z(t)-1} \leq \overline{D}_3(t)$	$t \leq t^1$	K_2	$\frac{b(t)}{\beta_P} - \frac{G^* Z(t)}{\beta_P(Z(t)-1)}$	$\frac{G^*}{Z(t)-1}$	G^*	$\frac{G^* Z(t)}{Z(t)-1}$
$Z(t)K_1 \leq \overline{D}_3(t)$	$t^1 \leq t \leq t^2$	K_1	$\frac{b(t)-Z(t)G^*}{\beta_P}$	G^*	$(Z(t)-1)G^*$	$Z(t)G^*$
$Z(t)K_1 > \overline{D}_3(t)$	$t^1 \leq t \leq t^2$	$\frac{(b(t)Z(t)-\overline{C}_1\beta_P-C_2(Z(t)-1)\beta_P)}{2Z(t)^2}$	$\frac{b(t)-Z(t)G^*}{\beta_P}$	G^*	$(Z(t)-1)G^*$	$\overline{D}_3(t)$

4 Results and discussion

4.1 Scenario and parameter settings

Overall performance evaluation is done using the software created for network rural planning in [17]. The real data presented in [13] and [18] are used as estimators for investment, maintenance costs, data traffic, and elasticities. The evolving potential market size $a(t)$ and the sensitivity of demand to price β_P were calculated for business days and weekends. Data used for simulation is summarized in tables 2 and 3. Parameters taking values in a range are presented as three arguments: the initial, final, and increment values.

With the proposed pricing model, only one delay tolerant service can be handled. Authors in [13] report two kinds of traffic that potentially could belong to this category: Http and e-mail, with e-mail being more delay tolerant than Http. Because the same elasticity value is reported for both services, we can consolidate their traffic and manage them as one service. Extending the model to more than one service and real time technology requires additional control variables and modifying the inventory constraint. This extension is left for future research.

Table 2: Network general parameters

Parameter	Value
Channel(Kbps)	32
Annual investment rate(%)	2,14,6
Monthly fixed cost (Usd)	330
Cost by channel (Usd)	90,170,10
Mechanic Backhaul cost (Usd)	10,70,15
Initial working Hour	6
Final working Hour	19
Cycle time (T_1)	7
Time average (T_{avg})	7
Investment periods	60

Table 3: Service parameters

Parameter	Delay Tolerant	Real Time
Elasticity	1.45	1.337
Demand contribution (%)	75	25
β_T	0.1,1,0.01	N/A
Base price ($^{Usd}/_{unit}$)	0.016	0.016,0.05,0.016

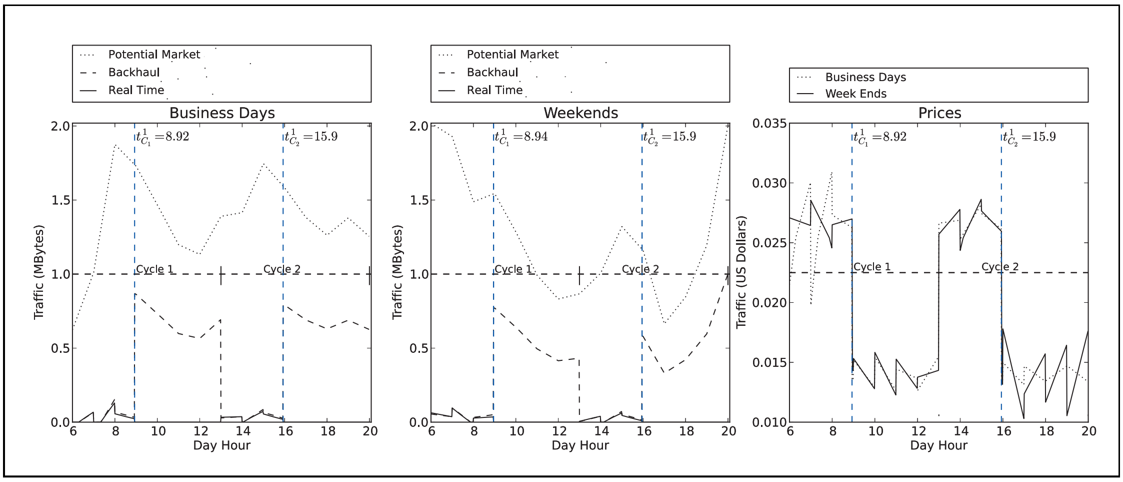


Figure 2: Traffic and price behavior

4.2 Results

The evolution of traffic by connection and price is presented in Figure 2. The two figures on the left show the potential market size and the optimal use of the real and mechanic backhaul connections. The two cycles are delimited and for each of them the time interval $t \in [0, t^1]$ in which constraint 3e is active and determined by K_2, and when it is determined by K_1, $t \in [t^1, t^2]$. Traffic use is strongly affected by the average time constraint regardless of business days or weekends. As long as the policy has to use the real time connection, it increases prices to cover link cost (see right graph). Hence, the optimal demand decreases limiting the network usage. Once the time constraint is more relaxed, the mechanic backhaul traffic increases as well as optimal demand. In fact, for the hypothetical scenario of $T_1 <= t^2$ the maximum mechanic backhaul connection use is determined by the potential market.

The last sub-figure in 2 corresponds to the optimal price. As it can be seen the figure presents jumps which are the result of changing β_P every hour within the days. Explicitly, we find the size of jumps to be proportional to demand changes between hours. From our tests we inference that maintaining a constant β_P eliminates jumps in prices, but under some scenarios decreases profits for operators. These jumps introduce comprehension complexities for users and operators, so we argue that maintaining a constant sensitivity to price for weekdays and weekends is the best decision for the whole system.

Using mixed technologies, the behavior of prices and demand might indicate a cost of real time channels too high for isolated areas. As this cost decreases, the model starts to decrease prices and the demand is stimulated for both channels. So we can suggest that mixed providers could be profitable in areas where WIMAX technologies could be installed (rental costs less that 90 USD dollars for 32 kbps month).

Second, operators desire to know how traffic and income are affected by different parameters used in the model. The impact of average time, cost of the real time connection, and market's time sensitivity are presented in Figure 3. If the operator only considers the time average parameter, his optimal decision would be to increase it as much as possible. However, this decision is correct as long as users do not have great sensitivity to time, low β_T values. If demand is closely related with time, high β_T values, the operator must establish an equilibrium between income gained as more traffic uses the mechanic backhaul connection and the corresponding demand lost.

The real time connection cost is a critical parameter. When the time average constraint is active, it determines the optimal demand $\overline{D}_3$, price, and quantity send by the two connection types. In the simulated test-bed, the most extreme case is assumed in which the operator has to use a VSAT connection. Surprisingly, these results indicate that deployment regions closer to connected Internet areas can benefit from mechanic backhaul connections when services require high volume traffic and users are to a certain degree insensible to delay.

There exists a correlation between real time costs and the traffic pattern used. It can be seen in figure 3 that a decrease in channel rental costs for business days produces more income than the same increase in weekends. The demand lost during the interval $[0, t^1]$ explains this outcome, which is the consequence of the variation in the potential market size throughout time. Therefore, we can conclude the need for using continuous models for adapting policies to demand variations.

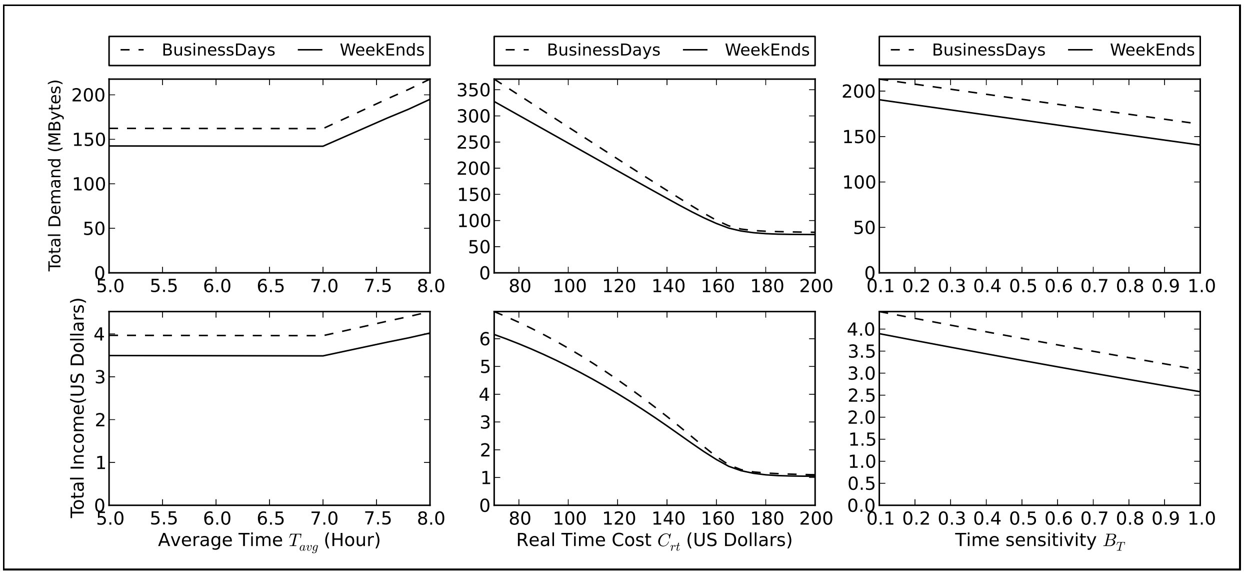


Figure 3: Change in aggregated results due to parameter shifts

5 Conclusions

We propose a model with price and delay time as demand predictors in which operators compromise themselves to deliver requests to maximize their profits. The results presented suggest that continuous models, like the one proposed, are pertinent to establish prices for delay tolerant services as long as they can handled variability of demand. Contrary to our first assumption, we observed that networks markets located close to Internet connected points can benefit from these services. Although real time channel lease cost continues to be a critical factor determining the demand offered to the market, results suggest that mechanic backhaul connections somehow mitigate the situation. In this paper we do not research, to name a few, the consequences of monopolistic operators' behavior, the extent to which sustainability of network deployments is reached, and the presence of economies of scale. All these research directions remain open.

Acknowledgment

This research is funded by Colciencias grant 567. This support is gratefully appreciated.

Bibliography

[1] Anand, A.; Pejovic, V.; Belding, E.; Johnson, D. (2012); VillageCell: Cost Effective Cellular Connectivity in Rural Areas; *Proceedings of the Fifth International Conference on Information and Communication Technologies and Development*; ISBN:978-1-4503-1045-1.

[2] Bayle, J.; Nagel, J; Raghavan S. (2008); Ex-post Internet charging; *Telecommunications Modeling, Policy, and Technology*; ISBN:978-0-387-77779-5 59-78.

[3] Boyaci, T.; Ray, S. (2003); Product Differentiation and Capacity Cost Interaction in Time and Price Sensitive Markets, *Manufacturing & Service Operations Management*; ISSN: 1523-4614, 5(1): 18-36.

[4] Cerf, V. et al (2007); Delay-Tolerant Networking Architecture.

[5] Coutinho, M. et al.(2011); A New Proposal of Data Mule Network Focused on Amazon Riverine Population; *Proceedings of the 3rd Extreme Conference on Communication: The Amazon Expedition*; ISBN:978-1-4503-1079-6.

[6] Gosztony, G. (1991); CCITT Work in Teletraffic Engineering, *IEEE Journal on Selected Areas in Communications*, ISSN:0733-8716, 9(2): 131-134.

[7] Ha, Sangtae. et al. (2012), TUBE: Time Dependent Pricing for Mobile Data; *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*; ISBN: 978-1-4503-1419-0.

[8] Hande, P.; Chiang, M.; Calderbank, R.; Zhang, J. (2010); Pricing under Constraints in Access Networks: Revenue Maximization and Congestion Management, *IEEE INFOCOM 2010 proceedings*; ISSN:0743-166X 1-9.

[9] Husni, E. (2011); Rural Internet Service System Based on Delay Tolerant Network (DTN) Using Train System; *International Conference on Electrical Engineering and Informatics (ICEEI)*, ISSN:2155-6822.

[10] Mackie-Mason J.; Varian H. (1998); The economics of the Internet; *Internet Economics*; ISBN:0-262-63191-1 27-62.

[11] Gizelis, C.; Vergados, D. (2011); A Survey of Pricing Schemes in Wireless Networks; *IEEE Communications Surveys and Tutorials*; ISSN:1553-877X 13(1): 126-145.

[12] Jiang, L.; Parekh, S.; Walrand J. (2008); Time dependent network pricing and bandwidth trading; *Network Operations and Management Symposium Workshops*; ISBN:978-1-4244-2067-4.

[13] Johnson, D.; Belding, E.; Almeroth, K. Van Stam, G. (2010); Internet Usage and Performance Analysis of a Rural Wireless Network in Macha, Zambia; *Proceedings of the 4th ACM Workshop on Networked Systems for Developing Regions*; ISBN:978-1-4503-0193-0.

[14] Kamien, M.; Schwartz, N. (1991); *Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management*, ISBN: 978-044-401-609-6.

[15] Khabbaz, M.; Assi, C.; Fawaz, W. (2012); Disruption-Tolerant Networking: A Comprehensive Survey on Recent Developments and Persisting Challenges; *IEEE Communications Surveys Tutorials*, ISSN:1553-877X, 14(2) 607-640.

[16] Ezziane, Z. (2005); Charging and Pricing Challenges for 3G Systems; *IEEE Communications Surveys Tutorials*; ISSN:1553-877X 7(4) 58-68.

[17] Marentes, L.; Donoso, Y. (2013); Assigning Capacity and Prices for Telecommunication Services to Increase Possibilities of Investment In Rural Networks; *Global Information Infrastructure and Networking Symposium (GIIS)*;

[18] Mo, J.; Kim, W.; Park, H. (2013); Internet Service Pricing: Flat or Volume?; *Journal of Network and Systems Management*; ISSN:1064-7570 21(2) 298-325.

[19] Pentland, A. and Fletcher, R. and Hasson, A. (2004); DakNet: Rethinking Connectivity in Developing Nations; *Computer*, ISSN:0018-9162, 37(1):78-83.

[20] Sen S.; Joe-WongC.; Ha, S.; Chiang M. (2012); Incentivizing Time-Shifting of Data: A Survey of Time-Dependent Pricing for Internet Access; *IEEE Communications Magazine*; ISSS:0163-6804 50(11) 91 - 99.

[21] Zheleva, M.; Paul, A.; Johnson, D.; Belding, E. (2013); Kwiizya: Local Cellular Network Services in Remote Areas; *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*; ISBN:978-1-4503-1672-9.

[22] Broadband InfraCo; http://www.infraco.co.za/SitePages/Home.aspx

[23] E-Mexico System; http://www.mexicoconectado.gob.mx/

[24] Bharat Broadband Network Limited; http://www.bbnl.nic.in/content/faq/what-is-bbnl.php

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):90-104, February 2016.

ANN Based Inverse Dynamic Model of the 6-PGK Parallel Robot Manipulator

L. Moldovan, H.-S. Grif, A. Gligor

Liviu Moldovan*, Horatiu-Stefan Grif, Adrian Gligor
Petru Maior University of Tirgu-Mures
Romania, 540088 Tirgu-Mures, Nicolae Iorga, 1
liviu.moldovan@ing.upm.ro, horatiu.grif@ing.upm.ro, adrian.gligor@ing.upm.ro
*Corresponding author: liviu.moldovan@ing.upm.ro

Abstract: This paper presents an inverse dynamic model estimation based on an artificial neural network of a complete new parallel robot manipulator prototype 6-PGK with six degrees of freedom, built at Petru Maior University of Tirgu-Mures. The model estimation of the parallel robot manipulator is performed with a feedforward artificial neural network. In the control engineering domain there are control structures that need the direct or inverse model of the process for ensuring the process control at the imposed performances. Usually, the determination of the direct/inverse mathematical model is a difficult or impossible task to be achieved. In these cases different non-parametric or parametric, off-line or on-line identification methods are used. A solution that may support the on-line parametric methods is represented by the feedforward artificial neural networks. By implementing feedforward artificial neural networks as a nonlinear autoregressive model with exogenous inputs, the authors investigate the possibility of choosing the optimum parameters that characterize the neural network so that it approximates as better as possible the model of the 6-PGK prototype robot. Finally an innovative algorithm is developed for obtaining the optimal configuration parameters set of the feedforward artificial neural network. The proposed algorithm helps in setting the optimal parameters of the neural network that offer high opportunities to provide satisfactory identification of the robot model. Experimental results obtained by a structure derived from the proposed solution demonstrate a good approximation related to the studied system, which is characterized by nonlinearities and high complexity.

Keywords: 6-DOF parallel robot manipulator, inverse dynamics, nonlinear model, unmodeled dynamics, feed-forward artificial neural network.

1 Introduction

In robotics, the class of parallel robots represent a constructive solution suitable for many industrial, medical or domestic applications, like flight simulators and entertainment rides [1], micromanufacturing [2], tool machine [3], pick and place operation [4], earthquake motion simulator [4], medical haptic devices [6], laparoscopic surgery [7], or even space docking technology [8].

Even if parallel robots have a reduced workspace [9], their benefits over their serial counterparts are various: stability and rigidity of contacts during haptic interaction [10], high accuracy, high stiffness, high payload capability, low moving inertia [11], good dexterity, compact size, large power to weight ratio [12]. Along with these advantages, there still exist many difficulties in the actual control process.

A classic approach in controlling robots consists in the employment of the dynamic model. Various methods have been proposed to derive the dynamic model of the parallel manipulator, like the generalized momentum approach [13], principle of virtual work [14] screw theory [15] and Hamilton's principle [16], the recursive matrix method [17] or the Lagrangian approach [11], which describes the dynamics of mechanical system from the concepts of work and energy [13].

All these approaches are equivalent as they are describing the same physical system, and lead to equivalent dynamic equations, which present different levels of complexity and associated computational loads.

The analytical calculus involved in the dynamic equation is very tedious, thus presenting an elevated risk for errors. Furthermore, the duration of numerical computations is getting longer when the number of mechanism's degrees of freedom is increased.

The dynamic model based control arises numerous difficulties because of the time-varying and coupling of the equations. Unfortunately, the present-day commercial controllers cannot provide satisfying performance for control.

In recent years an important credit in obtaining new enhancements is granted to the support offered by artificial intelligence techniques.

The role of artificial intelligence related to the parallel robots aims with predilection the modeling and control domain in terms of kinematics or dynamics. Moreover these strategies are designed to improve the classical control strategies, in terms of increasing the current performances, to add new properties such as robustness and improvement of the real-time behavior using fuzzy systems or different types of neural networks (NNET).

Compared to the mathematical model of the kinematics, neural networks provide significantly better mapping between manipulator's task space and joint space, as demonstrated by Uzunovic et al. [2].

Achili et al. in [18] proposed an adaptive force/position controller for parallel robot with constrained motions based on multi-layer perceptron neural network. The solution presents the major advantage of obtaining the control law without the prior knowledge of the inverse dynamic model and being able to take into account the endogenous disturbance (uncertainties and nonlinearities related to the robot dynamics) and to compensate exogenous disturbances.

Peng et al. [19] design a control system for a parallel mechanism consisting from a combination of linear controllers and two neural networks. The aim of this control system is to compensate the nonlinearities from the model of the system and the dynamic model estimation for improved accuracy of trajectory.

Le et al. [11] investigate the enhancement of the classical nonlinear PD torque controller by using neural networks for on-line self gain tuning. The proposed solution consists in a self gain tuning control law for minimizing the error in tracking trajectories of the parallel manipulators characterized by simple structure suitable for low computation effort in real-time implementations.

Yu and Weng in [20] propose a robust control structure for parallel manipulators based on H_∞ tracking adaptive fuzzy integral sliding mode control scheme mainly able to handle the nonlinear unmodeled dynamics and to compensate external disturbances.

All these solutions benefit from the advantages of various artificial intelligence techniques, but underline the difficulty of implementing them, starting from the the structure, the different parameters values or choice of training methods, and finally reaching the problem of required computational effort.

Recent research in the field of robot control is conducted in new directions based on bio-inspired approach that proved their efficiency on other fields of robotics, like the simplified fuzzy controller using neural network based on genetic algorithm learning for mobile robots [21], the communications and social structure of whale pods applied to cooperative robot structures [22] or fast genetic algorithm for robot soccer and planet exploration citelee. We consider that this direction of research may find applicability in the case of parallel robots, for the potential offered in solving many complex problems such as those in the case of parallel robot control discussed in this paper.

From the above explored literature we conclude that in the case of parallel robots there is a need to find adequate solutions for the real time control which involve increasing the trajectory following performance and low computational effort.

The objective of the paper is to build and implement an artificial neural network based on the inverse dynamic model estimation of a complete new parallel robot manipulator 6-PGK type with 6 degrees of freedom built at Petru Maior University of Tirgu-Mures.

2 The 6-PGK Parallel Robot Dynamics

2.1 The 6-PGK Parallel Robot Manipulator Design

The six controlled degrees of freedom (DOF) parallel robot configuration is a structure proposed by Stewart and Hunt and widely referred as Stewart platform [23–26]. The parallel kinematic link system proposed by D. Stewart is one of a 6-DOF robot manipulator used as a flight simulator [25, 26].

The basic geometric structure of the 6-PGK parallel robot manipulator developed at Petru Maior University consists in a mobile platform (supporting the end-effector) connected to the adjacent links at six distinct points (i=1,2,. . . ,6) at the level III by cardanic kinematic pairs (Figure 1) [27]. Six legs connect the platform to the base. Each leg has an actuated translational kinematic pair at level I and a spherical kinematic pair at level II. The translational kinematic pairs form the angles λ_i with the vertical axis in points B_i. The location (position and orientation) of the end-effector and of the manipulated object are expressed in the fixed reference frame $OXYZ$ by the generalized coordinates of the manipulated object X_p, Y_p, Z_p, ψ, ϕ, θ denoted q_{p_i} (i=1,2,. . . ,6), and in the mobile frame $oxyz$ by the x_p, y_p, z_p coordinates. The displacements in the actuated translational kinematic pairs (actuator displacements) are the generalized coordinates of the parallel robot q_i (i=1, . . . , 6).

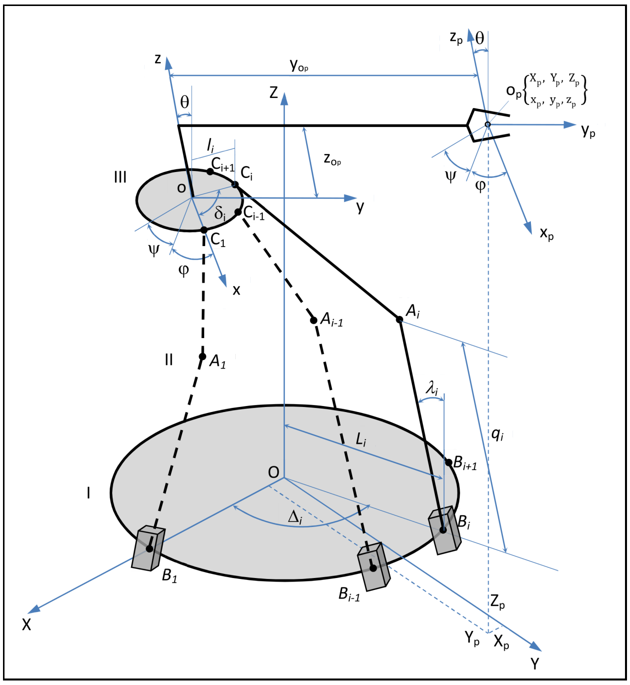


Figure 1: The structure of the 6-PGK parallel robot [27]

2.2 Parallel Robot Manipulator Dynamics Modeling

The dynamics modeling in case of robot manipulators refers to establishing a mathematical model that relates the time evolution of its end-effector in terms of positions, velocity and acceleration with the forces and moments acting upon it [28].

In order to get the position of the mobile platform with respect to generalized coordinates of the robot, the in-out equations are employed, which solve the positional problem of the robot. They relate the generalized coordinates of the robot to the generalized coordinates of the manipulated object [27]:

$$q_i^2 - 2b_i q_i + c_i = 0, \qquad (i = 1, \dots, 6) \tag{1}$$

where b_i, c_i are coefficients according to the robot mechanical parameters [27].

Determination of the velocity and acceleration of the mobile platform $[\dot{q_p}]$ with respect to generalized coordinates of the robot is performed using the kinematics equations:

$$[\dot{q_p}] = [J][\dot{q}] \tag{2}$$

obtained by deriving the in-out equations (1) with respect to time, where $[J]$ is the Jacobian matrix [27].

The dynamic modeling determines the generalized forces $[Q_j]$ that are required by the actuators to balance the forces applied to the robot.

In the case of the 6-PGK parallel robot, by employing the Lagrange method, the dynamic equations are obtained [27]:

$$\begin{aligned}
&\left\{\frac{1}{2}\sum_{k=1}^{3}\sum_{l=1}^{3} J_{kl}\left(\frac{\partial\omega_k}{\partial\dot{q}_{p_j}}\dot{\omega}_l + \dot{\omega}_k\frac{\partial\omega_l}{\partial\dot{q}_{p_j}}\right) + M\sum_{k=1}^{3}\ddot{q}_{p_k}\frac{\partial\dot{q}_{p_k}}{\partial\dot{q}_{p_j}} + \sum_{i=1}^{6} m_i\ddot{q}_i\frac{\partial\dot{q}_i}{\partial\dot{q}_{p_j}}\right\} + \\
&\left\{\begin{array}{r}
\frac{1}{2}\sum_{k=1}^{3}\sum_{l=1}^{3}\dot{J}_{kl}\left(\frac{\partial\omega_k}{\partial\dot{q}_{p_j}}\omega_l + \omega_k\frac{\partial\omega_l}{\partial\dot{q}_{p_j}}\right) - \frac{1}{2}\sum_{k=1}^{3}\sum_{l=1}^{3}\frac{\partial J_{kl}}{\partial q_{p_j}}\omega_k\omega_l + \\
\frac{1}{2}\sum_{k=1}^{3}\sum_{l=1}^{3} J_{kl}\left[\frac{d}{dt}\left(\frac{\partial\omega_k}{\partial\dot{q}_{p_j}}\right)\omega_l + \omega_k\frac{d}{dt}\left(\frac{\partial\omega_l}{\partial\dot{q}_{p_j}}\right) - \frac{\partial\omega_k}{\partial q_{p_j}}\omega_l - \omega_k\frac{\partial\omega_l}{\partial q_{p_j}}\right] + \\
\sum_{i=1}^{6} m_i\left[\dot{q}_i\frac{d}{dt}\left(\frac{\partial\dot{q}_i}{\partial\dot{q}_{p_j}}\right) - \dot{q}_i\frac{\partial\dot{q}_i}{\partial\dot{q}_{p_j}}\right]
\end{array}\right\} + \\
&\left\{Mg\frac{\partial Z_p}{\partial q_{p_j}} + \sum_{i=1}^{6} m_i g\frac{\partial q_i}{\partial q_{p_j}}C\lambda_i - m_0 g\left(x_p\frac{\partial\gamma\prime}{\partial q_{p_j}} + y_p\frac{\partial\gamma\prime\prime}{\partial q_{p_j}} + z_p\frac{\partial\gamma\prime\prime\prime}{\partial q_{p_j}}\right)\right\} = Q_j \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (j = 1, \dots, 6)
\end{aligned} \tag{3}$$

where Q_j are the generalized forces due to the non-conservative forces, m_i are masses of the motor links, J_{kl} are the elements of the inertia matrix of the mobile platform of mass m_0 and manipulated object of mass m_p, their sum is $M = m_0 + m_p$, then $\alpha\prime$, …, $\gamma\prime\prime\prime$ are terms of the rotational matrix that relates the coordinates fixed in mobile frame to the base coordinates and ω_x, ω_y, ω_z are the angular velocity terms in $OXYZ$ system.

The generalized motor forces are deduced from equation [27]:

$$Q_j = \sum_{i=1}^{6} l_{ij} Q_{m_i} \tag{4}$$

where $l_{i,j}$ are terms of the inverse Jacobian matrix.

The inverse dynamic model is useful for the robotic system control. It consists in determination in real time of the generalized motor forces Q_{m_i} from equations (4), by solving the positional problem from equation (1), the kinematic problem from equation (2) and the dynamic model from equation (3).

Parallel robots with a high number of degrees of freedom, in particular six degrees like the 6-PGK parallel robot, are modeled by complex dynamic equations, comprising a large number of parameters. Solving them requires crossing a high number of iterations and selection of admissible real solutions. The complexity of the program, the large number of loops, testing all solutions, selecting the nearest current configuration solution, can induce errors in configuration identification. Therefore in current industrial exploitation, parallel robots control requires the use of simple and effective control methods.

3 FANN Based Dynamic Model of the 6-PGK Parallel Robot Manipulator

3.1 FANN Structure for Robot Dynamic Modeling

For the 6-PGK parallel robot manipulator modeling a feed-forward artificial neural network (FANN) with one hidden layer has been designed. The FANN is implemented as a nonlinear autoregressive model with exogenous inputs (NARX) model. The NARX model is based on the linear ARX model, which is commonly used in time-series modeling. [30]

The numbers of the inputs of the used FANN is determined by the numbers of the generalized coordinates of the robot, the number of the generalized forces and the number of theirs considered past values. The numbers of the outputs of the used FANN is given by the numbers of the generalized forces from the dynamical model of the robot.

By using the compact structure of an artificial neuron presented in [29] we can summarize the mathematical processing that occurs inside of an artificial neuron from the used FANN as:

$$a = F\left(\sum_{j=1}^{n} \omega_j x_j + \theta\right) \tag{5}$$

where are denoted by: x_j $(j = 1, \dots, n)$ - the inputs of the neuron, ω_j $(j = 1, \dots, n)$ - the weights of the inputs x_j, θ - the weight of the offset input, n - the numbers of the inputs of the neuron, F is the activation function, a - the activation of the neuron.

The type of the activation function and the numbers of the inputs corresponding to each neuron depend on the layer to which it belongs. For the neurons from the hidden layer are used two types of the activation function: the hyperbolic tangent function (HTF) type and the radial basis function (RBF) type [30]. The numbers of the inputs of the neurons from this layer is given by the maximum lags associated with robot output (n_Q) and input (n_{q_p}) signals described by relation $6(n_{q_p}+n_Q)$, which also represents the number of the inputs of the FANN. The activation function of the neurons from the output layer is the linear type function and the number of the inputs of these neurons equals the number of the neurons from the hidden layer of the FANN. For the training of the FANN the gradient descent backpropagation algorithm was used [30]. The training of the used FANN was achieved by presenting to their inputs and outputs the training set of the N pairs of the vectors:

$$\{\left(\underline{x}^i(1), \underline{a}_d^o(1)\right), \left(\underline{x}^i(2), \underline{a}_d^o(2)\right), \dots, \left(\underline{x}^i(N), \underline{a}_d^o(N)\right)\}$$

where $\underline{x}^i(k)$ is an input training vector $(k = 1, \dots, N)$ with $6(n_{q_p}+n_Q)$ elements and $a_d^o(k)$ is the desired target output vector (with 6 elements) of the FANN corresponding to the input training vector $x^i(k)$.

3.2 FANN Optimal Parameters Determination

Analytical determination of the FANN parameters is an uncommon task, thus the empirical approach is more often used. The optimum determination however is a more complex task, so in this section a tested methodology for determination of these parameters is proposed.

Usually, the investigation of a system starts from the searching of the parameters that influence the behavior of the studied system. The same approach was established for the present work. In the case of the proposed structure of neural network the following four parameters were considered: learning rate (γ), the number of the neurons on the hidden layer (n_h), the number of the past values of the generalized forces (n_Q) and the number of the past values of the generalized coordinates (n_q). In order to study the influence of these parameters on the neural network behavior for building the FANN training data sets the generalized coordinates trajectory of the robot and the corresponding trajectories for generalized motor forces will be used.

An adequate solution for the process of searching through the experimental set can be performed by visual analysis of all the available results (large sets of plotted trajectories). But this approach presents a major drawback from time needed for analysis and required storage resources. In order to facilitate and automatize the evaluation process, quality measures can be used. An option is represented by the mean squared error or the mean absolute error. However, both measures present major difficulty in the case of establishing of a threshold value to aid in deciding the quality of the robot model identification. In order to overcome the mentioned drawbacks for the evaluation process the authors propose an automated method and the usage of a quality measure based on signal-to-noise ratio (SNR). For the 6-PGK prototype robot the SNR is given by the following equation:

$$SNR_i = 10 \cdot log_{10} \frac{\sum\limits_{j=1}^{N} \Big(Q_{mi}(j) \Big)^2}{\sum\limits_{j=1}^{N} \Big(Q_{mi}(j) - Q_{mi_{FANN}}(j) \Big)^2} \; \big[dB\big], \qquad (i = 1, \ldots, 6) \tag{6}$$

where Q_{mi} represents the generalized motor trajectory forces given by the inverse dynamical model of the robot, $Q_{mi_{FANN}}$ represents the generalized motor trajectory forces given by the FANN and N is the number of the trajectory samples.

The proposed SNR quality measured was derived from the signal-to-noise ratio used for evaluation of unidimensional electrical signals.

The algorithm for FANN optimal parameters determination is presented below.

General Algorithm for Choosing the Structure of the FANN

Let us consider the following parameters which influence the behavior of the FANN: learning rate (γ), the number of the neurons (n_h) from hidden layer, the number of the past values for the input (n_{q_p}) and output (n_Q) variables employed to build the robot regressors vectors (input vectors of the FANN) used in training and simulation of the FANN. These parameters are referred as "FANN parameters". The steps of the proposed algorithm for choosing the structure of the FANN are:

Step 0. Select (experimentally): the number of the initial sets of values for the FANN weights and the corresponding weights values (that are random real numbers in the range $[-0.5,\ 0.5]$), having built a collection of initial sets of values for the FANN weights, the signal-to-noise ratio threshold SNR_{thold}, the sets of values for the FANN parameters. To each set of values of the FANN parameters is attached a set of counter variables of the same size as the size of the value of the considered set. Thus for a FANN parameter to each value (from the set of values of the considered FANN parameter) it corresponds a counter variable. The value of the counter variables is initialized with zero;

Step 1. Set the following values of the FANN parameters with start values $n_h = n_{hStart}$, $n_Q = n_{QStart}$, $n_{q_p} = n_{qpStart}$. These values are drawn from the sets of the FANN parameter values;

Step 2. Keeping constant the values of the FANN parameters $n_h = n_{hStart}$, $n_Q = n_{QStart}$, $n_{q_p} = n_{qpStart}$ establish the optimal learning rate γ_{opt} based on the Algorithm for Evaluation and Determination of the Optimum Value for the FANN parameters;

Step 3. Keeping constant the value of the parameters γ=γ_{opt}, n_Q =n_{QStart}, n_{qp} =$n_{qpStart}$ establish the optimal number of hidden layer neurons ($n_{h\,opt}$) based on the Algorithm for Evaluation and Determination of the Optimum Value for the FANN parameters;

Step 4. Keeping constant the value of the FANN parameters $gamma = \gamma_{opt}$, $n_h = n_{h\,opt}$ establish optimal values of the number of the past samples for the input and output variables of the robot model (n_{Qopt}, n_{qpopt}) based on the Algorithm for Evaluation and Determination of the Optimum Value for the FANN parameters to which the following changes are made corresponding to the steps *Step* 4′′, *Step* 6′′ and *Step 7*;

Step 5. Keeping constant the value of the FANN parameters $gamma = \gamma_{opt}$, $n_h = n_{h\,opt}$ and $n_q = n_{q\,opt}$ it is tested the estimation and identification algorithm considering other trajectories for input and output variables of the robot (or model of the robot).

Algorithm for Evaluation and Determination of the Optimum Value for the FANN

For each value from the set of values corresponding to the FANN parameter set to be evaluated, the following steps will be executed separately for the each set of the initial weights:

Step 1′*.* Initialize the weights of the FANN with the corresponding initial selected set of weights;

Step 2′*.* Run the estimation and identification algorithm calculating the signal-to-noise ratio (SNR) based on the relation (6) for each output path (trajectory corresponding to a generalized forces), thus corresponding to each set of initial weights, a set of six values of the noise-to-signal ratio will be obtained;

Step 3′*.* Compute the average of the signal-to-noise ratio SNR_{med} as mean value of the six signal-to-noise ratio determined in *Step 2*;

Step 4′*.* Compare the value of SNR_{med} with the threshold value SNR_{thold}. If the average is greater than the threshold then increment the counter variable corresponding to the current value of the FANN parameter and go to *Step* 5′;

Step 4′′*.* Compare the value of the SNR_{med} with the threshold value SNR_{thold}. If the average value is greater than the threshold value, then increment the counter variable corresponding to the current values combination for the two FANN parameters;

Step 5′. If the sets of initial values for weights are not exhausted select a new set of values for weights and jump to *Step* 1′. If the sets of initial values for weights were exhausted continue with *Step* 6′;

Step 6′. Using the values of the counter variables, the success rates corresponding to each FANN parameter value is computed by

$$SR_i = \frac{counter_i}{no. of\, the\, initial\, weights\, sets} \cdot 100\,[\%]$$

where $i = 1, \ldots, no.$ *of the FANN parameter values.* Go to *Step* 7′;

Step 6″. Using the values of the counter variables, the success rates corresponding to each combination of the FANN parameters values are computed and go to *Step* 7″;

Step 7′. Choose the FANN parameter value for which the success rate is the largest. Finally this value will be considered as the optimal value for the evaluated FANN parameter;

Step 7″. Choose a pair of values of the two FANN parameters for which the success rate is the largest. Finally the values of the selected pair will be considered as the optimal for the two parameters of the FANN evaluated.

3.3 Experimental Results and Discussions

For the implementation and testing of the parallel robot modeling, Matlab platform and its Neural Network Toolbox were used. In order to investigate the performance of the FANN based modeling the model of the parallel robot based on equations 1 and 2 was implemented.

In Figure 2 six desired trajectories are presented for the generalized coordinates of the manipulated object (X_p, Y_p, Z_p, ψ, ϕ, θ denoted q_{p_i}, $i = 1, \ldots, 6$). These trajectories are placed in parallel surfaces having the same 3D spatial shape at different elevation. These trajectories represent the input trajectories for the Inverse Dynamic Model of the 6-PGK Parallel Robot Manipulator.

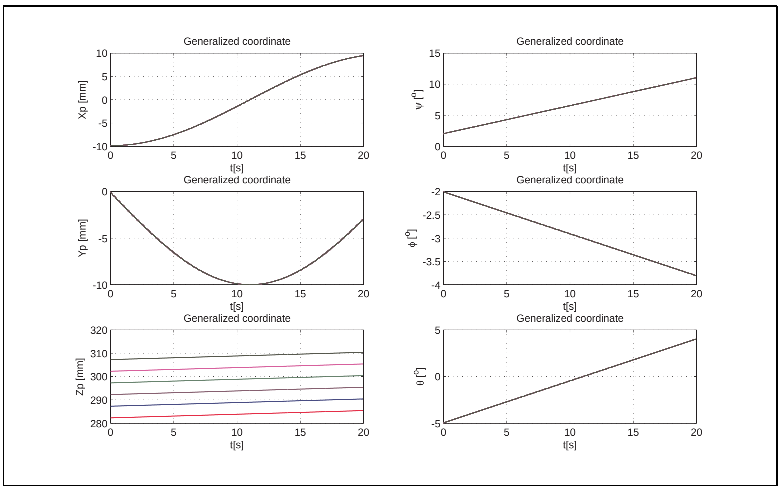


Figure 2: The generalized coordinates trajectories of the 6-PGK parallel robot

In order to study the influence of the FANN parameters on the neural network behavior for building the FANN training data sets, one of the spatial generalized coordinates trajectory was used, which has the red Z_p line from Figure 2 and the corresponding red trajectories for generalized motor forces from Figure 3.

In Figure 3 six desired trajectories for the generalized motor forces (Q_{mj}, $j = 1, \ldots, 6$) are presented. These trajectories represent the output trajectories of the Inverse Dynamic Model of the 6-PGK Parallel Robot Manipulator corresponding to the input trajectories from Figure 2.

The training data are set up using the trajectories presented in Figures 2 and 3, trajectories that consist from 2001 points each.

The estimation and identification algorithm is applied in each time step and consists of two phases:
- the estimation phase - based on the curent input vector, the FANN computes the curent estimated vector of generalized forces;
- the identification phase - based on curent input/output trainig vectors the FANN is trained.

An input training vector $\big(\underline{x}_i(k), k = 0, \ldots, N-1\big)$ and an output training vector $\big(a_d^o(k), k = 0, \ldots, N-1\big)$, from the training vectors set, are constructed as follow:

$$\begin{aligned} \underline{x}^i(k) &= \big[x_1^i(k-1) \ \ldots \ x_6^i(k-1) \ \ldots \ x^i_{6(n_{q_p}+n_Q-1)+1}(k-n_Q) \ \ldots \ x^i_{6(n_{q_p}+n_Q)}(k-n_Q)\big] \\ &= \big[X_p(k-1) \ \ldots \ \theta(k-1) \quad \ldots \quad X_p(k-n_{q_p}) \ \ldots \ \theta(k-n_{q_p}) \\ &\qquad \ldots \ Q_{m1}(k-1) \ \ldots \ Q_{m6}(k-1) \quad \ldots \quad Q_{m1}(k-n_Q) \ \ldots \ Q_{m6}(k-n_Q)\big] \end{aligned} \tag{7}$$

$$\underline{a}_d^o(k) = \big[a_{d1}^o \quad a_{d2}^o \quad \ldots \quad a_{d6}^o\,\big] = \big[Q_{m1}(k) \quad Q_{m2}(k) \quad \ldots \quad Q_{m6}(k)\big] \tag{8}$$

The order of the vectors in the vectors training set corresponds to the order of the time sample from the time axis. Taking into account this observation, the input training vector from (7) and the training output vector from (8) can be expressed in relation with time:

$$\begin{aligned} \underline{x}^i(t) = \big[x_1^i(kT-T) \quad \ldots \quad x_6^i(kT-T) \quad \ldots \quad & x^i_{6(n_{q_p}+n_Q-1)+1}(kT-n_QT) \\ & \ldots \quad x^i_{6(n_{q_p}+n_Q)}(kT-n_QT)\big] \end{aligned} \tag{9}$$

$$\underline{a}_d^o(t) = \underline{a}_d^o(kT) = \big[Q_{m1}(kT)\, Q_{m2}(kT) \ \ldots \ Q_{m6}(kT)\big] \tag{10}$$

where T is the sampling time.

The results obtained by using the above algorithms are presented as follows. The experimental study was started from the following FANN parameters values: $n_h = 5$, $n_Q = 1$, $n_{q_p} = 1$, 20 initial weights sets values, $SNR_{thold} = 44.5[dB]$, $\gamma = 0.01, 0.05, 0.1, \ldots, 0.98, 1$. During the *Step 2* of the general algorithm the success rates of γ were computed and then synthesized in Figure 4. The analysis of Figure 4 led to the selecting of the value of 0.25 as the optimal value for the γ parameter. During *Step 3* the success rates of n_h were computed and then synthesized in Figure 5. The analysis of Figure 5 led to the selecting of the value of 7 as the optimal value for the n_h parameter. Further, after the evaluation of *Step 4* the results are synthesized in Table 1. The values of the lag for n_Q are taken successively from 1 to 5. The values for n_{q_p} are taken successively from 1 to n_Q.

Analyzing the experimental results presented in Table 1 were selected as optimal values $n_{q_p} = 1$ and $n_Q = 1$.

In case of the FANN with RBF the proposed algorithm generate the following parameters: $\gamma = 0.25$, $n_h = 6$, $n_Q = 1$ and $n_{q_p} = 1$.

Comparing data from Table 1 and Table 2, Figures 4, 5, 6 and 7 shows that FANN with RBF offer a better chance of achieving a good estimation even if the learning rate hasn't a 100% success rate.

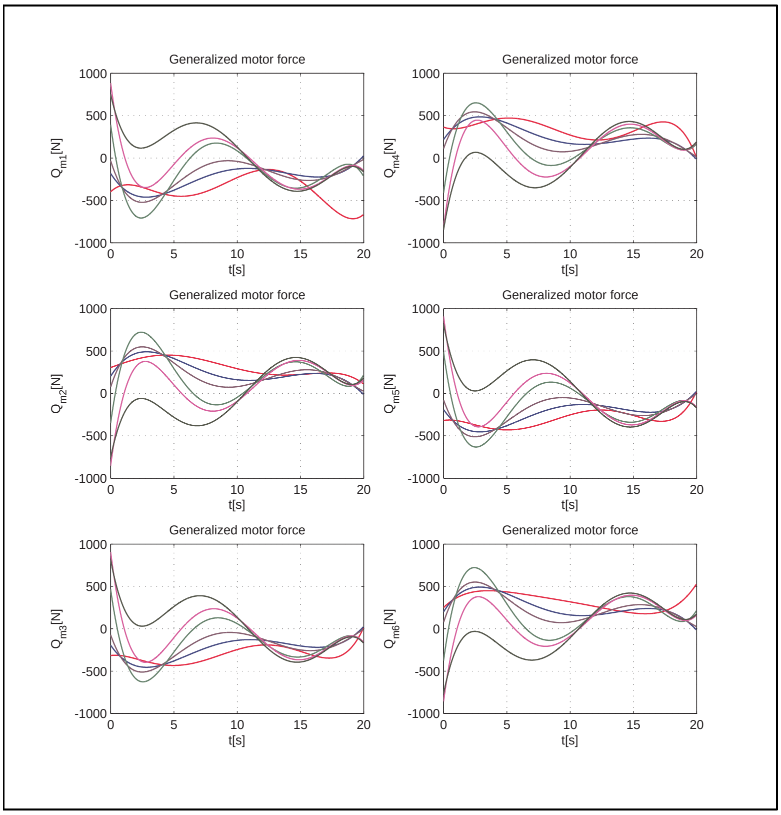


Figure 3: The generalized motor forces of the 6-PGK parallel robot

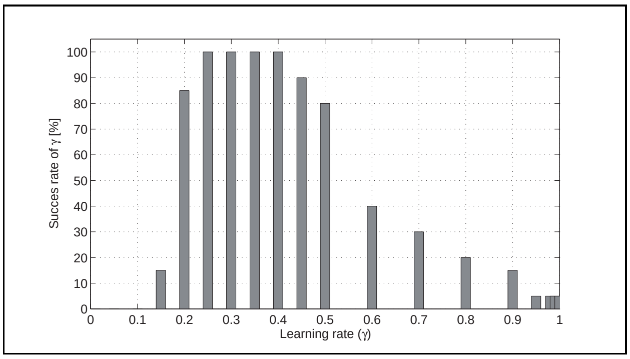


Figure 4: Success rates of γ parameter in case of using HTF in hidden layer

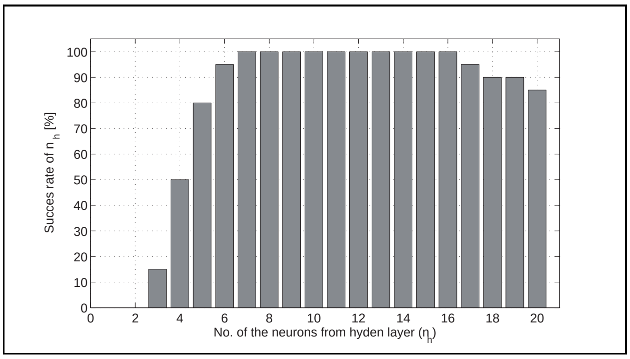


Figure 5: Success rates of n_h parameter in case of using HTF in hidden layer ($\gamma = 0.25$)

Table 1: The success rates corresponding to (n_Q, n_{q_p}) (FANN: HTF, $n_h = 7$, $\gamma = 0.25$)

Success rates [%]		n_{q_p}				
		1	2	3	4	5
n_Q	1	100	-	-	-	-
	2	100	75	-	-	-
	3	90	70	35	-	-
	4	60	55	15	20	-
	5	45	35	20	20	10

Table 2: The success rates corresponding to (n_Q, n_{q_p}) (FANN: RBF, $n_h = 6$, $\gamma = 0.25$)

Success rates [%]		n_{q_p}				
		1	2	3	4	5
n_Q	1	100	-	-	-	-
	2	95	85	-	-	-
	3	85	85	65	-	-
	4	75	70	45	40	-
	5	60	65	40	30	15

In order to evaluate the behavior of the built FANN, in Figure 8 a test motor forces trajectories of the parallel robot and the estimated trajectory generated by the FANN are presented comparatively.

In Figure 9 the estimation error trajectories related to the trajectories from Figure 8 is shown. The experimental results demonstrate a good approximation related to the studied system which is characterized by nonlinearities and high complexity.

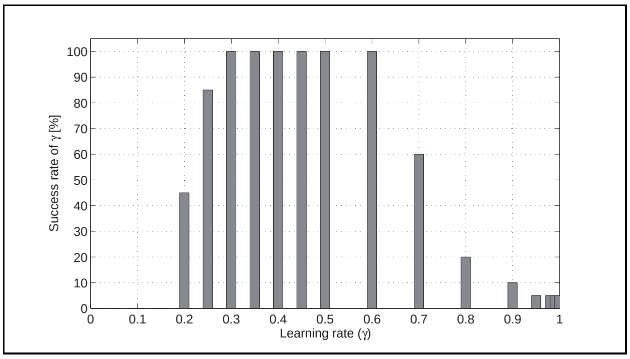


Figure 6: Success rates of γ parameter in case of using RBF in hidden layer

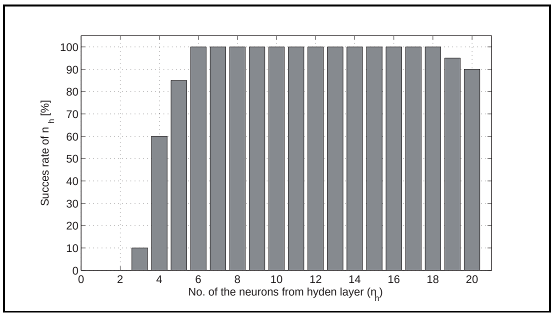


Figure 7: Success rates of n_h parameter in case of using RBF in hidden layer($\gamma = 0.25$)

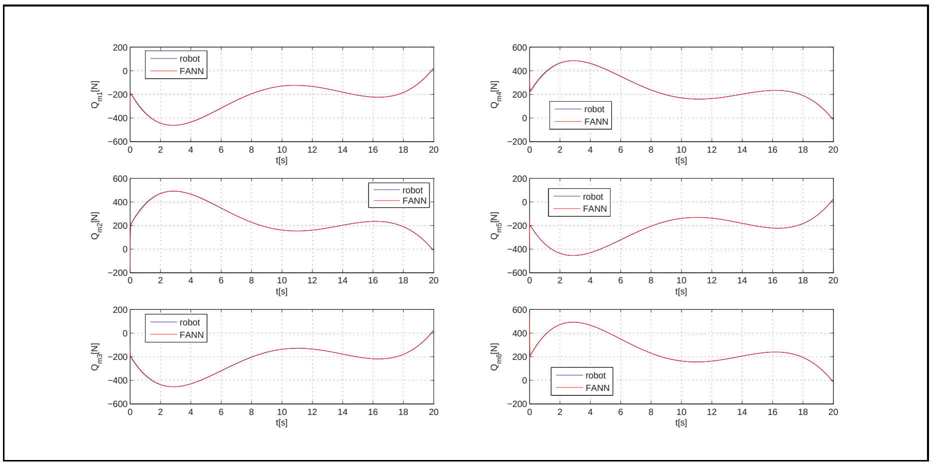


Figure 8: The FANN estimated motor forces trajectory and the motor forces trajectory of the 6-PGK parallel robot (FANN: RBF, $n_h = 6$, $\gamma = 0.25$)

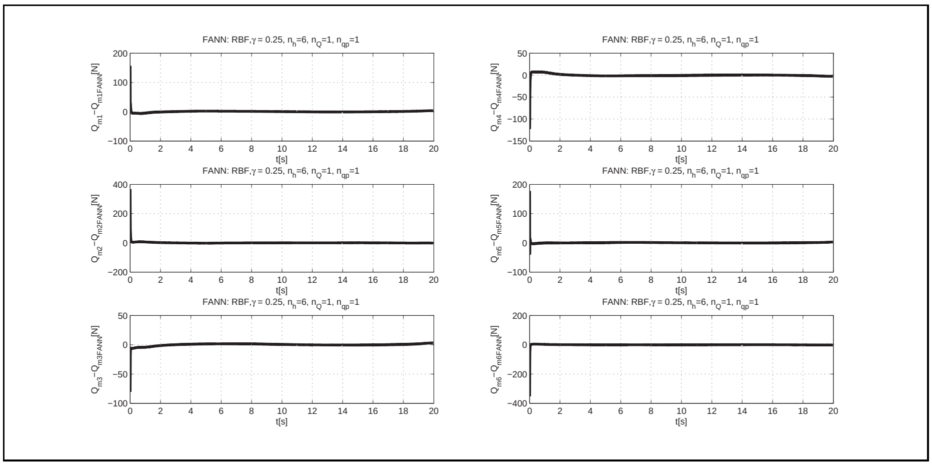


Figure 9: Estimation error trajectory corresponding to Figure 8

4 Conclusions

This paper proposed to build an on-line feedforward artificial neural network with the aim of estimating the inverse dynamic model of the 6-PGK prototype parallel robot.

The presented solution was adopted in order to perform an on-line parametric identification of the inverse dynamic model of the robot. The chosen solution was adopted mainly for the use when the robots are characterized by the nonlinearities and high mathematical model complexity.

The implementation of the feedforward artificial neural networks allowed to obtain a nonlinear autoregressive model with exogenous input behavior, for which a new method for finding optimum parameters that approximate as better as possible the model of the 6-PGK robot was obtained. In addition, the proposed solution offers, with the best estimation results, the minimal computational load structure of the neural network.

Even if the FANN has a minimal internal structure, a slower training learning method, it offers a robust and efficient estimation method of the parallel robot motor trajectories, despite its complex and nonlinear mathematical model.

Bibliography

[1] Hamlin, G.J.; Sanderson, A.C. (1997); TETROBOT: a modular approach to parallel robotics, *IEEE Robotics & Automation Magazine*, ISSN 1070-9932, 4(1): 42-50.

[2] Uzunovic, T.; Golubovic, E.; Baran, E. A.; Sabanovic, A. (2013); Configuration space control of a parallel Delta robot with a neural network based inverse kinematics, *IEEE Proc. of 2013 8th Intl. Conf. on EEE (ELECO)*, 497-501.

[3] Abdellatif H.; Heimann, B. (2010); Experimental identification of the dynamics model for 6-DOF parallel manipulators, *Robotica*, 28:359-368. doi:10.1017/S0263574709005682.

[4] Yang, Z.; Wu, J.; Mei, J.; Gao, J.; Huang, T. (2008); Mechatronic model based computed torque control of a parallel manipulator, *International Journal of Advanced Robotic Systems*, 5(1):123-128.

[5] Zhang, J.; Yu, H.; Gao, F.; Zhao, X.; Ma, C.; Huang, X. (2010); Application of a novel 6-DOF parallel robot with redundant actuation for earthquake simulation, *IEEE Proc. on Intl. Conf. on Robotics and Biomimetics (ROBIO)*, 513-518.

[6] Najafi, F.; Sepehri N. (2008); A novel hand-controller for remote ultrasound imaging, *Mechatronics*, 18: 578-590.

[7] Ibrahim, K.; Ramadan, A.; Fanni, M.; Kobayashi, Y.; Abo-Ismail, A.A.; Fujie, M.G. (2012); Design and workspace analysis of a new endoscopic parallel manipulator, *IEEE Proc of 2012 12th Intl. Conf. on Control*, Automation and Systems (ICCAS), 688-693.

[8] Jian, X.; Zhiyong, T.; Zhongcai, P.; Shao, H.; Lanbo, L. (2013); Adaptive controller for 6-DOF parallel robot using TS fuzzy inference, *International Journal of Advanced Robotic Systems*, 10(119):1-9.

[9] Moldovan, L. (2008); Geometrical Method for Description of the 6-PGK Parallel Robot's Workspace, *IEEE Proc. of First Intl. Conf. on Complexity and Intelligence of the Artificial and Natural Complex Systems, Medical Applications of the Complex Systems, Biomedical Computing*, 2008. CANS'08, 45-51.

[10] Constantinescu, D.; Salcudean, S.E.; Croft E.A. (2005); Haptic Rendering of Rigid Contacts Using Impulsive and Penalty Forces, *IEEE Transactions On Robotics*,21(3):309-323.

[11] Le, T. D.; Kang, H. J.; Suh, Y. S.; Ro, Y. S. (2013); An online self-gain tuning method using neural networks for nonlinear PD computed torque controller of a 2-dof parallel manipulator, *Neurocomputing*, 116: 53-61.

[12] Lu, Y.; Li, X.P. (2014); Dynamics analysis for a novel 6-DoF parallel manipulator I with three planar limbs, *Advanced Robotics*, 28(16): 1121-1132. DOI: 10.1080/01691864.2014.908743.

[13] Lopes, A.M. (2010); Complete dynamic modelling of a moving base 6-dof parallel manipulator, *Robotica*, 28:781-793, doi:10.1017/S0263574709990506.

[14] Staicu, S.; Liu X.-J.; Wang, J. (2007); Inverse dynamics of the HALF parallelmanipulator with revolute actuators, *Nonlinear Dyn.*, 50: 1- 12.

[15] Gallardo J.; Rico, J.; Frisoli, A.; Checcacci, D.; Bergamasco M. (2003); Dynamics of parallel manipulators by means of screw theory, *Mech. Mach. Theory*, 38: 1113-1131.

[16] Miller, K. (2004); Optimal Design and Modeling of Spatial Parallel Manipulators, *The Int. J. of Robotics Research*, 23(2): 127-140.

[17] Staicu, S.; Zhang, D.; Rugescu, R. (2006); Dynamic modelling of a 3-DOF parallel manipulator using recursive matrix relations, *Robotica*, 24:125-130, doi:10.1017/S0263574705001670.

[18] Achili, B.; Daachi, B.; Amirat, Y.; Ali-Cherif, A.; DaĂ˘chi, M. E. (2012); A stable adaptive force/position controller for a C5 parallel robot: a neural network approach, *Robotica*, doi:10.1017/S0263574711001354, 30(7):1177-1187.

[19] Peng, Z.; Liu, F.; Yang, L. (2010); Control based on double neural networks-PI for parallel mechanism, *Robotics and Computer-Integrated Manufacturing*, 26(3): 250-252.

[20] Yu, W. S.; Weng, C. C. (2014); H^{∞} tracking adaptive fuzzy integral sliding mode control for parallel manipulators; Fuzzy Sets and Systems, ISSN 0165-0114, 248: 1-38.

[21] Obe, O.; Dumitrache, I. (2012); Adaptive Neuro-Fuzzy Controler With Genetic Training For Mobile Robot Control, *International Journal od Computers Communications & Control*, 7(1): 135-146.

[22] Resceanu, I. C.; Resceanu, C. F.; Bizdoaca, N. G. (2012); Cooperative Robot Structures Modeled After Whale Behavior and Social Structure, *International Journal od Computers Communications & Control*, 7(5):945-956.

[23] Liu, X.; Wang, Q.; Malikov, A.; Wang, H. (2012); The design and dynamic analysis of a novel 6-DOF parallel mechanism, *International Journal of Machine Learning and Cybernetics*,, 3(1): 27-37.

[24] Pei, Z.; Zhang, Y.; Tang; Z. (2007); Model reference adaptive PID control of hydraulic parallel robot based on RBF neural network, *IEEE Proc. of Intl. Conf.on Robotics and Biomimetics, ROBIO 2007*, 1383-1387.

[25] Beji, L.; Pascal, M. (1999); The kinematics and the full minimal dynamic model of a 6-DOF parallel robot manipulator, *Nonlinear Dynamics*, 18(4): 339-356.

[26] Ayas, M. S.; Sahin, E.; Altas, I. H. (2014); Trajectory tracking control of a Stewart platform, *Proc. of IEEE 2014 16th Intl. Conf. and Exposition on Power Electronics and Motion Control (PEMC)*, 720-724.

[27] Moldovan, L. (2008); Trajectory Errors of the 6-PGK Parallel Robot, *IEEE Proc. of First Intl. Conf. on Complexity and Intelligence of the Artificial and Natural Complex Systems, Medical Applications of the Complex Systems, Biomedical Computing*, 2008. CANS'08, 31-37.

[28] Mendes Lopes, A.; Almeida, F. (2009); The generalized momentum approach to the dynamic modeling of a 6-dof parallel manipulator, *Multibody System Dynamics*, 21(2): 123-146.

[29] Grif, S. (2014); Automatic daylight control system based on neural estimator, *Procedia Technology*, 12: 759-765.

[30] Demut, H.; Beale, M.; Hagan, M. (2007); Neural Network Toolbox For Use with Matlab. User's Guide. Version 5, 2007, The MathWorks, Inc.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):105-112, February 2016.

Membrane Computing and Economics: A General View

G. Păun

Gheorghe Păun
Institute of Mathematics of the Romanian Academy
PO Box 1-764, 014700 Bucureşti, Romania
gpaun@us.es

Abstract: Three are the points we briefly discuss here: using membrane computing tools for efficient computing/optimization, the possibilities of using "general" membrane computing (P systems using multisets of symbol objects processed by biochemical-like evolution rules) as a framework for modeling economic processes, and the numerical P systems, a class of computing devices explicitly defined with a motivation related to economics. The discussion is rather informal, only pointing out research directions and providing bibliographical information.

Keywords: membrane computing, P system, economics, numerical P system.

1 A Quick Glimpse to Membrane Computing

Membrane computing is a branch of natural computing aiming to abstract computing models from the structure and the functioning of the living cells – at least this was the initial motivation of the research in this area, [12]. Basically, in the compartments of a cell-like membrane structure one places multisets (sets with multiplicities associated with their elements) of *objects*, represented by symbols of a given alphabet, and *evolution rules*, in the form of "reactions" transforming certain objects in other objects (multiset rewriting rules, written in the form $u \to v$, where u and v are multisets); the objects can also pass through membranes, from a compartment to another one. Starting from an initial configuration (of multisets and membranes) and applying the evolution rules according to specified strategies (semantics), one obtains transitions among configurations, hence *computations*. In this way we get a computing device, usually called a *P system*. Results are associated with computations, in various ways.

Most investigated in membrane computing are P systems with the rules applied in the non-deterministically maximally parallel way (a multiset of rules is non-deterministically chosen and applied in each multiset, such that there is no applicable multiset which contains strictly the chosen multiset) and the result associated only with halting computations, those which reach a configuration where no evolution rule can be applied.

There are many variations of this basic model. First, one can consider various forms of the rules, as well as ways of controlling their application: priority, promoters, inhibitors, periodical change of rules in time, etc. Then, other ways of using the rules can be considered: sequential, limited parallelism, minimal parallelism, asynchronous systems, etc. Also the result of a computation can be defined in various ways (internally, externally, in the form of a number or of a string), not necessarily for halting computations.

Further suggestions come from biology. For instance, instead of multiset rewriting rules, as those corresponding to biochemical reactions, we can consider symport or antiport rules, which move multisets of objects across membranes, [11]. Similarly, we can consider rules for handling also the membranes, not only the objects: membranes can be created, dissolved, divided, the relationships between membranes can change in timne (e.g., by exocytosis, phagocytosis, etc.), and so on.

Then, an extension can be considered from hierarchical arrangements of membranes, like in a cell, described by trees, to a tissue-like arrangement of membranes, described by a general graph. All ingredients mentioned above can be extended to this case.

In short, a framework for processing multisets (of symbol objects) in a distributed parallel way.

Instead of symbol objects, we can also consider more complex objects, strings for instance – and then the evolution rules should be chosen suitably.

Most of the classes of the P systems suggested above are equivalent as computing power with Turing machines, and this happens even for rather restricted (as the number of membranes or the complexity of the evolution rules) systems.

Furthermore, due to their massive parallelism, and to the possibility of creating an exponential workspace in a linear time, certain classes of P systems can solve computationally hard problems (typically, **NP**-complete problems) in a feasible (polynomial) time. (This has been called in [14] *fypercomputing*, as an extension from hypercomputing, with "f" coming from "fast".) This is based on a time-space trade-off, with the space created by operations like membrane creation, membrane division, string duplication. Depending on the used ingredients, even (polynomial) characterizations of **PSPACE** are obtained.

Two classes of P systems different from the previous ones are the *numerical P systems*, which will be discussed below, and the *spiking neural P systems* (in short, SN P systems), [7], inspired from the way the neurons cooperate in a brain, communicating through spikes, electrical impulses of identical shape. The results are the same also for SN P systems: computational universality and the possibility of *fypercomputing* in the case when neuron division or similar operations are present. Another interesting idea leading to fypercomputations in this area is to use (arbitrarily large) pre-computed resources, with a limited amount of information present in the initial configuration.

Besides these theoretical directions of investigations, concerning the power and the efficiency of P systems, well developed is the application area, naturally, starting with applications in biology and biomedicine. Many other applications are already reported, in ecology, linguistics, computer graphics, cryptography, approximate optimization, robot control – as well as to economics.

The literature of membrane computing is rather large. Details can be found at the domain web site from `http://ppage.psystems.eu`. An introduction can be found in [13], a comprehensive coverage (at the level of 2009) in [18], while applications can be found in [4], [6], [20]. Many collective volumes, PhD theses and downloadable papers can be found at the above mentioned web address.

2 Using Membrane Computing Tools in Economics

Two ideas mentioned in the previous section suggest already that membrane computing can be useful for economics, especially in the computability, the operation research, and the optimization. The P systems as models will be discussed below, here we have in mind P systems as tools for handling "classic" models.

First, we mentioned the many results where **NP**-complete problems are solved in polynomial time by certain classes of P systems. Most of the non-trivial practical problems appearing in economics (as well as in other areas) are **NP**-complete. Thus, it would be great to solve them in a polynomial time by using a "cellular computer". However, the solutions provided by membrane computing are still *in info*, theoretical, as there is no implementation of P systems *in vitro/vivo*, not even *in silico* (such that the massive parallelism can be effectively used). There are implementations of P systems on parallel hardware, but not at the level of making practical the theoretical fypercomputing. On the other hand, it is highly probably that even a bio-implementation of a P system will not help too much, it will be similar to the case of DNA computing, with the 1994 Adleman experiment, which remained at the level of a *demo*, of solving toy problems, without practical consequences.

Much more practical is the possibility of using the so-called *membrane algorithms* in solving computationally hard optimization problems. This is a rather developed direction of research in membrane computing, with a large bibliography and very encouraging results. Basically, this is nothing else than distributed evolutionary computing, but with the distribution controlled by means of membranes, and with further ingredients from membrane computing used, such as membrane creation, division, dissolution, communication among membranes, neural-like organization of cells, etc. Many engineering applications of membrane algorithms were reported – the reader is advised to consult [4], [6], [18], the forthcoming volume [20], as well as the bibliography from the membrane computing web site.

3 P Systems as Models for Economics

We pass now to a more particular approach, namely to consider membrane computing as a framework for building models for economic processes.

Of course, the first question which could be formulated is: why? There are so many (mathematical) frameworks already used in economics, why considering a new, somewhat exotic (for instance, by its biological inspiration), one? The answer is complex. In general, new tools are always good to be checked, maybe they will bring new possibilities to address old problems in a new framework, ideally, to also formulate new questions in the new framework. Membrane computing is not only *natural* in a direct sense, but also sufficiently developed at the theoretical, abstract level that it is just expected to be applicable in many areas, apparently far away from biology. Then, the biological metaphor is very general and very useful, as indicated by the many applications of membrane computing in areas rather different from one another.

More specifically, there are several features of P systems which make them attractive as models, for economics like for biology, too. They pertain to discrete mathematics, so that they are adequate to situations when we have to deal with small numbers of objects and agents. Applying tools of continuous mathematics (especially differential equations) in situations which are clearly of a discrete type leads to wrong results. Then, P systems are easily scalable, easily understandable, easily programmable – three appealing properties. The behavior of a (non-trivial) P system is “emergent", impossible to be predicted by simply examining the rules. (The universality also implies, through Rice theorem, that no non-trivial question about a P system can be answered algorithmically, hence a computer simulation is needed.) Finally, there are many economic issues which can be easily and transparently captured in terms of membrane computing. Membranes can describe the organization, the multiset rewriting rules and the symport/antiport rules can describe production and commercial operations.

Only a hint, recalling some lines from [16].

Assume that units of row material a are introduced on a market by a supplier S and are used by producers P_i, each having a number of production units b_i, to manufacture goods d_i; these goods are then sold by P_i at price p_i to retailers R_j, each having a number of capacity units c_j; in turn, the retailers sell the goods to a general consumer C, which introduces the need for d, denoted by $\bar{d}$, on the market. For instance, by rules

$$S \to Sa^n, \quad C \to C\bar{d}^m,$$

the supplier introduces n units of a and the consumer introduces the need for m copies of d.

Producer P_i can produce one d by consuming one a and one b_i,

$$b_i a \rightarrow d_i,$$

while retailer R_j takes one order for d by means of

$$c_j \bar{d} \rightarrow \bar{d}_j.$$

(The good d and the need for d are indexed with the label of the producer – who fix its price – and of the retailer, respectively.) Then, the producers and the retailers interact: by

$$d_i \bar{d}_j \rightarrow b_i c_j \ \ [p_{i,j}],$$

one copy of good d passes from P_i to R_j, because R_j has an order for this good, and this happens with the probability $p_{i,j}$, which depends on the previous transactions between producer P_i and retailer R_j; units of capacities, b_i and c_j, are free again.

We have not yet included money here, but this can be easily done. Instead of $b_i a \rightarrow d_i$, we have to consider the rule

$$b_i a u_i^r \rightarrow d_i u_S^r,$$

with the meaning that P_i has paid r monetary units (denoted by u_i when belonging to P_i) for a, thus S has gained r copies of u_S (notation for a monetary unit in possession of S). In turn, instead of $d_i \bar{d}_j \rightarrow b_i c_j \ \ [p_{i,j}]$, we have to use

$$d_i \bar{d}_j v_j^{p_i} \rightarrow b_i c_j u_i^{p_i} \ \ [p_{i,j}],$$

with the meaning that p_i monetary units have passed from the "account" (= multiset) of R_j to that of P_i. The retailer will then sell the good at a price (if possible) greater than p_i to the general consumer C, and the cycle is repeated.

The model can still be enlarged: for instance, the number of a's and $\bar{d}$'s introduced on the market can fluctuate, the prices producers and retailers set can vary in time; then, both producers and retailers can make investments (transform money into production/storing units), maybe in a prudent manner (with low probability associated with investment rules and higher probability associated with saving rules, of the form $u_i \rightarrow u_i$, just keeping the money unused), with or without limits on the total number of investments, and so on.

It is important to note that coefficients $p_{i,j}$ written in square brackets on the right hand side of the rules are not "pure" probabilities, but they also express the trust of retailers when buying from producers, depending on the history of their collaboration. These coefficients can be changed over time, which adds to the complexity of the model.

The reader is refereed to [15], [16], [17] for further details, illustrations, and discussions.

Several other titles are provided in the bibliography, where economic processes are approached in terms of P systems.

After writing such a model, it is necessary to simulate it on a computer, hence a program is needed (ideally, implemented on a parallel support). Of course, the model is not used as a computing one, but the evolution itself of the model is of interest (this is similar to the applications of P systems in biology).

We are convinced that this direction of research deserves further efforts.

4 Numerical P Systems

This class of P systems was introduced (in paper 16 from the list in the next section) with an explicit economic inspiration. Briefly speaking, in the regions of a cell-like membrane structure, we have *numerical variables* (not "chemicals", as in the P systems discussed above), evolving by means of *production-repartition programs.* The variables from region i are denoted by

$x_{1,i}, x_{2,i}, \ldots, x_{k_i,i}$. If the model is used as a computing devices, the variables take integer values, but for applications the values can be real numbers.

A production-repartition program (from region i) is of the form

$$F_i(x_{1,i}, \ldots, x_{k_i,i}) \longrightarrow c_1|v_1 + c_2|v_2 + \cdots + c_{n_i}|v_{n_i},$$

where $F_i(x_{1,i}, \ldots, x_{k_i,i})$ is the "production function" and $c_1|v_1 + c_2|v_2 + \cdots + c_{n_i}|v_{n_i}$ defines the "repartition protocol". Denote $C_i = \sum_{s=1}^{n_i} c_s$ and denote by $x_{i,j}(t)$ the value of variable $x_{i,j}$ at step $t \geq 0$ ($t = 0$ corresponds to the initial values of variables).

At any time $t \geq 0$, we compute $F_i(x_{1,i}(t), \ldots, x_{k_i,i}(t))$. The value $q = F_i(x_{1,i}(t), \ldots, x_{k_i,i}(t))/C_i$ represents the "unitary portion" to be distributed to variables $v_1, \ldots, v_{n_i}$, according to coefficients $c_1, \ldots, c_{n_i}$ in order to obtain the values of these variables at time $t+1$. Specifically, v_s will receive $q \cdot c_s, 1 \leq s \leq n_i$. If a variable receives such "contributions" from several neighboring compartments, then they are added in order to produce the next value of the variable. A variable used in a production function is consumed, reset to 0; if a variable does not appear in a production function, then its value remains unchanged – of course, it can change by receiving "production portions" (note that they can also be negative) from its region or from the neighboring regions. In this way, we pass from given values of the variables to next values. The process can be iterated. In this way, we get a computation. The positive values of a specified variable form the set of numbers *generated* by a numerical P system.

Again, further ingredients can be added – for instance, a control on the use of a program, such as the "enzymatic control" considered in several papers dealing with robot controllers based on numerical P systems, or restrictions can be considered – particular functions as production functions. As expected, these systems are both powerful (Turing complete) and efficient.

In the robot control area, numerical P systems were used as devices for computing functions of several real variables, and this seems to be the most plausible idea also for economic applications. Anyway, again we expect applications in the two directions mentioned also for usual P systems: as tools for efficient computing and as models.

Both at the theoretical level and in what concerns the applications, numerical P systems are waiting for further attention, and we believe that the effort will be rewarding. In order to help the reader in this direction, we provide below a list of all titles we know at this moment related to numerical P systems (but further research are in development).

5 A Bibliography of Numerical P Systems

1. O. Arsene, C. Buiu, N. Popescu: SNUPS, A simulator for numerical membrane computing. *Intern. J. of Innovative Computing, Information and Control*, 7, 6 (2011), 3509–3522.

2. C. Buiu, O. Arsene, C. Cipu, M. Pătraşcu: A software tool for modeling and simulation of numerical P systems. *Biosystems*, 103, 3 (2011), 442-447.

3. C. Buiu, C.I. Vasile, O. Arsene: Development of membrane controllers for mobile robots. *Information Sciences*, 187 (2012), 22–51.

4. M. Garcia-Quismondo, L.F. Macias-Ramos, M.J. Pérez-Jiménez: Implementing enzymatic numerical P systems for AI applications by means of graphic processing units. *Topics in Intelligent Engineering and Informatics*, vol. IV, Springer, Berlin, 2013, 137–159.

5. A. Leporati, A.E. Porreca, C. Zandron, G. Mauri: Improved universality results for parallel enzymatic numerical P systems. *Intern. J. Unconventional Computing*, 9 (2013), 384–404.

6. A. Leporati, G. Mauri, A.E. Porreca, C. Zandron: Enzymatic numerical P systems using elementary arithmetic operations. *Membrane Computing. Proc. 14th Intern. Conf., CMC 2013, Chişinău, August 2013*, LNCS 8340, Springer, Berlin, 2014, 249–264.

7. D. Llorente–Rivera, M.A. Gutiérrez–Naranjo: The pole balancing problem with enzymatic numerical P systems. *Proc. BWMC 2015*, Sevilla, February 2015, Fenix Editora, Sevilla, 2015 (in press).

8. A.B. Pavel: *Membrane Controllers for Cognitive Robots.* Master's thesis, Department of Automatic Control and System Engineering, Politehnica University of Bucharest, Romania, February 2011.

9. A.B. Pavel: *Development of Robot Controllers Using Membrane Computing.* PhD Thesis, Department of Automatic Control and System Engineering, Politehnica University of Bucharest, Romania, July 2015.

10. A.B. Pavel, O. Arsene, C. Buiu: Enzymatic numerical P systems – a new class of membrane computing systems. *The IEEE Fifth Intern. Conf. on Bio-Inspired Computing. Theory and applications. BIC-TA 2010*, Liverpool, Sept. 2010, 1331–1336.

11. A.B. Pavel, C. Buiu: Using enzymatic numerical P systems for modeling mobile robot controllers. *Natural Computing*, 11, 3 (2012), 387–393.

12. A.B. Pavel, I. Dumitrache: Hybrid numerical P systems. *UPB Scientific Bulletin*, Series A, submitted 2014.

13. A.B. Pavel, C.I. Vasile, I. Dumitrache: Robot localization implemented with enzymatic numerical P systems. *Proc. Conf. Living Machines 2012*, LNCS 7375, Springer, Berlin, 2012, 204–215.

14. Gh. Păun: Some open problems about numerical P systems. *Proc. 11th Brainstorming Week on Membrane Computing*, Sevilla, 4-8 February 2013, Fénix Editora, Sevilla, 2013, 235–242.

15. Gh. Păun: Some open problems about catalytic, numerical and spiking neural P systems, *Membrane Computing. Proc. 14th Intern. Conf., CMC 2013, Chişinău, August 2013*, LNCS 8340, Springer, Berlin, 2014, 33–39.

16. Gh. Păun, R. Păun: Membrane computing and economics: Numerical P systems. *Fundamenta Informaticae*, 73 (2006), 213–227.

17. Gh. Păun, R. Păun: Membrane computing as a framework for modeling economic processes. *Seventh Intern. Symp. Symbolic and Numeric Algorithms for Scientfic Computing*, SYNASC 2005, IEEE Press, 2005, 11–18.

18. Gh. Păun, R. Păun: Membrane computing and economics. Section 23.6 in *Handbook of Membrane Computing*, OUP, 2010, 632–644.

19. C.I. Vasile: *Distributed Control for Multi-Robot Systems.* PhD Thesis, Department of Automatic Control and System Engineering, Politehnica University of Bucharest, Romania, July 2015.

20. C.I. Vasile, A.B. Pavel, I. Dumitrache: Universality of enzymatic numerical P systems. *Intern. J. Computer Math.*, 90, 4 (2013), 869–879.

21. C.I. Vasile, A.B. Pavel, I. Dumitrache, J. Kelemen: Implementing obstacle avoidance and follower behaviors on Koala robots using numerical P systems. *Tenth Brainstorming Week on Membrane Computing*, Sevilla, 2012, vol. II, 215–227.

22. C.I. Vasile, A.B. Pavel, I. Dumitrache, Gh. Păun: On the power of enzymatic numerical P systems. *Acta Informatica*, 49, 6 (2012), 395–412.

23. X. Wang, G. Zhang, F. Neri, T. Jiang, J. Zhao, M. Gheorghe, F. Ipate, R. Lefticaru: Design and implementation of membrane controllers for trajectory tracking of nonholonomic wheeled mobile robots. *Integrated Computer-Sided Engineering*, to appear.

6 Concluding Remarks

In many circumstances, computer science got very fruitful inspiration from biology, and this became a systematic research direction in the last decades under the mane of *natural computing*. A comprehensive overview can be found in [19].

Membrane computing is part of this endeavor, well developed at the mathematical level, general, versatile, promising at the level of applications in very (apparently) different areas.

Here, we only mentioned some of the possibilities and the researches reported so far concerning the applications of membrane computing in economics, calling the attention to this research area, with the conviction that it deserves to be explored.

Bibliography

[1] J. Bartosik (2004); Paun's systems in modeling of human resource management. *Second Conference on Tools and Methods of Data Transformation*, WSU, Kielce, Poland..

[2] J. Bartosik (2004); Heaps of pieces and Paun's systems. *Second Conference on Tools and Methods of Data Transformation*, WSU Kielce.

[3] J. Bartosik, W. Korczynski (2002); Systemy membranowe jako modele hierarchicznych struktur zarzadzania, *Mat. Pokonferencyjne Ekonomia, Informatyka, Zarzadzanie. Teoria i Praktyka*, Wydzial Zarzadzania AGH, Tom II, AGH 2002.

[4] G. Ciobanu, Gh. Păun, M.J. Pérez-Jiménez (eds.) (2006); *Applications of Membrane Computing.* Springer, Berlin.

[5] R. Freund, M. Oswald, T. Schirk (2007); How a membrane agent buys goods in a membrane store. *Progress in Natural Science*, 17: 442–448.

[6] P. Frisco, M. Gheorghe, M.J. Pérez-Jiménez (eds.) (2014); *Applications of Membrane Computing in Systems and Synthetic Biology*, Springer, Berlin.

[7] M. Ionescu, Gh. Păun, T. Yokomori (2006); Spiking neural P systems. *Fundamenta Informaticae*, 71(2-3): 279–308.

[8] W. Korczynski (2004); On a model of economic systems. *Second Conference on Tools and Methods of Data Transformation*, WSU Kielce.

[9] W. Korczynski (2005); Păun's systems and accounting. *Pre-Proceedings of Sixth Workshop on Membrane Computing WMC6*, Vienna, Austria, 461–464.

[10] M. Oswald (2007); Independent agents in a globalized world modelled by tissue P systems. *Artificial Life and Robotics*, 11: 171–174.

[11] A. Păun, Gh. Păun (2002); The power of communication: P systems with symport/antiport. *New Generation Computing*, 20(3): 295–306.

[12] Gh. Păun (2000); Computing with membranes. *J. Computer and System Sciences*, 61(1): 108–143, first circulated as *TUCS Report 208*, November 1998 (www.tucs.fi).

[13] Gh. Păun (2002); *Membrane Computing. An Introduction.* Springer, Berlin.

[14] Gh. Păun (2012); Towards "fypercomputations" (in membrane computing). *Languages Alive. Essays Dedicated to Jurgen Dassow on the Occasion of His 65 Birthday* (H. Bordihn, M. Kutrib, B. Truthe, eds.), LNCS 7300, Springer, Berlin, 207–221.

[15] Gh. Păun, R. Păun (2005); Membrane computing as a framework for modeling economic processes. *Proc. SYNASC 05*, IEEE Press, 11–18.

[16] Gh. Păun, R. Păun (2006); A membrane computing approach to economic modeling: The producer-retailer interactions and investments. *Analiză şi Prospectivă Economică*, Part I: 3(1): 30–37, Part II: 4(2): 47–54.

[17] Gh. Păun, R. Păun (2006); Membrane computing models for economics. An invitation-survey. *Studii şi Cercetări de Calcul Economic şi Cibernetică Economică*, 40(1-2): 5–19.

[18] Gh. Păun, G. Rozenberg, A. Salomaa, eds.(2010); *Handbook of Membrane Computing.* Oxford University Press.

[19] G. Rozenberg, T. Bäck, J.N. Kok, eds. (2010); *Handbook of Natural Computing.* 4 vols., Springer, Berlin, 2012.

[20] G. Zhang, M. Gheorghe, M.J. Pérez-Jiménez (2016); *Real-life Applications with Membrane Computing.* Springer, Berlin (in press).

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):113-125, February 2016.

A Preflow Approach to Maximum Flows in Parametric Networks Applied to Assessing the Legal Information

L. Sângeorzan, M. Parpalea, M.M. Parpalea,
A. Repanovici, R. Matefi, I. Nicolae

Livia Sângeorzan*
Transilvania University of Brasov
Faculty of Mathematics and Computer Science
25, Eroilor, Blvd, 500030, Brasov, Romania
*Corresponding author:sangeorzan@unitbv.ro

Mircea Parpalea
National College Andrei Şaguna of Brasov
1, Şaguna, street, 590000, Brasov, Romania
parpalea@gmail.com

Mihaela Marinela Parpalea
Transilvania University of Brasov
Faculty of Letters
25, Eroilor, Blvd, 500030, Brasov, Romania
mihaela.parpalea@unitbv.ro

Angela Repanovici
Transilvania University of Brasov
Faculty of Product Design and Environment
25, Eroilor, Blvd, 500030, Brasov, Romania
repa@unitbv.ro

Roxana Matefi, Ioana Nicolae
Transilvania University of Brasov
Faculty of Law
25, Eroilor, Blvd, 500030, Brasov, Romania
roxana.matefi@unitbv.ro, ioana.nicolae@unitbv.ro

Abstract: The present paper proposes a fractal-like approach to the parametric flow problem, derived from the rules and recursion of generative linguistics. In the same manner in which any sentence can be analysed in terms of "theme" of the sentence (that which is spoken about in the sentence) and "rheme" of the sentence (that which is said about the theme in the sentence), in the proposed parametric preflow-push algorithm, a "partitioning push" (a non-cancelling push of flow in the parametric residual network) might leave the node unbalanced for only a subinterval of the range of parameter values. This will lead to separating the problem for the two disjoints subintervals, which allows the algorithm to continue after the same rules, independently, on each of the partitioned subintervals. The algorithm runs as the template-like structure of a dialogue act which reveals a design where information about the items (part-of-speech) is a two sections vector with one segment for each of the used part of speech categories. The article also proposes a possible application of the algorithm in assessing legal information and in semantic evaluation of legislation.

Keywords: Parametric preflow algorithm, partitioning technique, generative linguistics, legal information.

1 Introduction

Computing maximum flows in various networks is an important problem not only because the results are applied directly to traffic analysis of communication networks, but also because some efficient algorithms are often employed as sub-problems in other general network problems. Consequently, fundamental algorithms for computing maximum network flows were designed and efficient algorithms exist [1] to solve different instances [7] of the problem. A natural generalization of this problem can be obtained by considering that the capacities of some arcs are functions of a real parameter. The obtained parametric maximum flow problem is to compute all maximum flows for every possible value of the parameter.

For the maximum flow problem in parametric networks with zero lower bounds and linear capacity functions of a single parameter, Hamacher and Foulds [3] investigated a "horizontal" approach, based on augmenting directed paths algorithm. In each of its iterations, the algorithm computes an improvement of the flow defined on the whole interval of the parameter. For the same problem, Ruhe [8,9] proposed a "piece-by-piece", "vertical" approach. The main idea of his algorithm is to consider that the parametric maximum flow is known up to a parameter value λ_k and to compute a flow augmentation that assures the optimality of the flow for parameter values $\lambda \geq \lambda_k$ as well as the parameter maximum value λ_{k+1} up to which the computed flow remains a parametric maximum one. Each of the algorithm's iterations represents a non-parametric maximum flow sub-problem which can be solved via one of the classic approaches for the maximum flow problem.

Partitioning technique in networks has been, in the latest years, a more and more active research topic in both engineering and theoretical research. The reason why the problem under consideration is of genuine practical and theoretical interest lies in that graph partitioning applications are described on a wide variety of subjects, such as: data distribution in parallel-computing, VLSI circuit design, image processing, computer vision, route planning, air traffic control, mobile networks, social networks, etc. [2]. Unfortunately, graph partitioning is an NP-hard problem, and therefore all known algorithms for generating partitions merely return approximations to the optimal solution. Partitioning algorithms for the parametric maximum flow problem can be developed starting from any of the three classic approaches for the maximum flow problem: flow augmenting directed paths algorithms, preflow algorithms or min-max algorithm. The approach based on flow augmenting directed paths [4], makes use of the concept of shortest conditional augmenting directed path. In order to avoid working with piecewise linear functions, the approach uses a series of parametric residual networks defined for successive subintervals of the parameter values where the parametric residual capacities of all arcs remain linear functions. The approach based on preflow algorithms will further be presented in detail in this paper while the parametric min-max approach [6] is not itself an algorithm but rather a method of obtaining a parametric minimum flow using any algorithm for obtaining a maximum parametric flow and vice versa.

The idea of the partitioning approach based on preflows, which is used in this paper, derives from the rules and recursion of generative linguistics [10]. The concept which has proved to be the most useful in the description of German word order has become known under the name of Functional Sentence Perspective (FSP). Its main idea is that information is not transmitted in random order but the speaker seeks to give his information to his interlocutor in portions, normally starting from what he assumes is common to both and proceeding to what he regards as important new information [5]. In the same manner in which any sentence can be analysed in terms of "theme" of the sentence (that which is spoken about in the sentence) and "rheme" of the sentence (that which is said about the theme in the sentence), the proposed algorithm for the parametric maximum flow problem uses a fractal-like approach. In the proposed parametric preflow-push algorithm, a 'partitioning push' (a non-cancelling push of flow in the parametric residual network) might leave the node unbalanced for only a subinterval of the range of the values of the parameter. Like in all fractal approaches a partitioning push is followed by separating the problem into disjoints subintervals, allowing the algorithm to continue after the same rules independently on each of the partitioned subintervals. Making use of the partitioning approach, the present paper proposes an original algorithm for computing the maximum flow in parametric networks with linear upper bound (capacity) functions. Although both the algorithm presented in [10] and the one that is proposed in this article deals with the same kind of approach of the parametric flow, they are totally different algorithms and they fundamentally differ because of their purpose: the first computes the minimum flow in networks with constant capacities and linear lower bound functions while the latter computes the maximum flow in networks without lower bounds and with linear capacity functions.

Further on, this paper is organized as follows. Section 2 presents the parametric maximum flow problem and the basic parametric network flow terminology and results which are used in the rest of the paper. Most of the definitions in this section are straightforward modifications of those in [4] and [6], adapted for the maximum flow problem in parametric networks with zero lower bounds. More specialized terminology and further details on the notions, definitions and main results within this section can be found in [1], [4] and [6]. In Section 3 we present our algorithm for solving the parametric maximum flow problem, called "Highest label partitioning push (HLPP) algorithm". In the same section there are also presented the corresponding theorems of correctness and of complexity of the algorithm. Using a multithread based parallel implementation, the complexity of the algorithm is linearly dependent on the number of breakpoints. Section 4 deals with a possible application of the proposed algorithm to assessing the legal information. Finally, Section 4 presents some conclusions regarding to our contribution to the topic presented in the article. In the presentation to follow, some familiarity with flow algorithms is assumed and many details are omitted, since they are straightforward modifications of known results.

2 Flows in parametric networks

The parametric maximum flow problem is an extension of the classical maximum flow problem [1] in which the capacities of certain arcs are functions of a parameter λ. The parametric maximum flow problem is to compute all maximum flows for every possible value of the parameter. For the case of linear capacity functions, the maximum flow value function in a parametric network is a continuous piecewise linear function of the parameter. Each linear segment of the maximum flow value function between the two breakpoints λ_k and λ_{k+1} corresponds to a cut that remains a minimum cut for any $\lambda_k < \lambda \leq \lambda_{k+1}$. The approach presented in this article refers to the maximum flow problem in a network with linear capacity functions of a single parameter.

2.1 Terminology and preliminaries

Let $G = (N, A, u, s, t)$ be a capacitated network with $n = |N|$ and $m = |A|$, $N = \{\dots, i, \dots\}$ being the set of nodes i and $A = \{\dots, a, \dots\}$ being the set of arcs a, so that for every arc in A holds that $a = (i, j)$ with $i, j \in N$. The capacity (*upper bound*) function is a nonnegative function $u(a)$ associated with each arc $a \in A$. The network has two special nodes: a source node s and a sink node t. A *flow* is a function $f : A \to \mathbb{R}^+$ satisfying the next conditions:

$$\sum_{j|(i,j)\in A} f(i,j) - \sum_{j|(j,i)\in A} f(j,i) = \begin{cases} v, & i = s \\ 0, & i \neq s,t \\ -v, & i = t \end{cases} \tag{1}$$

for some $v \geq 0$, where v is referred to as the value of the flow f. Any flow on a directed network satisfying the flow bound constraints:

$$0 \leq f(i,j) \leq u(i,j), \qquad \forall (i,j) \in A \tag{2}$$

for every arc $(i,j) \in A$ is referred to as a *feasible flow.* A *cut* is a partition of the node set N into two subsets S and $T = N - S$, denoted by $[S,T]$. A cut is nontrivial if both S and T are nonempty. An arc $(i,j) \in A$ with $i \in S$ and $j \in T$ is referred to as a *forward arc* of the cut while an arc $(i,j) \in A$ with $i \in T$ and $j \in S$ as a *backward arc* of the cut. Let (S,T) denote the set of forward arcs in the cut and let (T,S) denote the set of backward arcs. A cut $[S,T]$ is an $s-t$ cut if $s \in S$ and $t \in T$. The *maximum flow problem* is to determine a flow $\tilde{f}$ for which v is maximized

2.2 The parametric maximum flow

The parametric flow problem consists in generalising the classic problem of flows in networks by transforming the upper bounds of some arcs $(i,j) \in A$ of the network $G = (N, A, u, s, t)$ in linear functions of a real parameter.

Definition 1. *[4]* A directed network $G = (N, A, u, s, t)$ for which the upper bounds u of some arcs $(i,j) \in A$ are functions of a real parameter λ is referred to as a *parametric network* and is denoted by $\bar{G} = (N, A, \bar{u}, s, t)$.

For a parametric network $\bar{G}$, the *parametric upper bound* (*capacity*) function $\bar{u} : A \times [0, \Lambda] \to \mathbb{R}^+$ associates to each arc $(i,j) \in A$, for each of the parameter values $\lambda \in [0, \Lambda]$, the real number $\bar{u}(i,j;\lambda)$, referred to as the parametric upper bound of arc (i,j):

$$\bar{u}(i,j;\lambda) = u_0(i,j) + \lambda \cdot U(i,j), \qquad (i,j) \in A, \tag{3}$$

where $U : A \to \mathbb{R}$ is a real valued function associating to each arc $(i,j) \in A$ the real number $U(i,j)$ referred to as the *parametric part of the upper bound* of the arc (i,j). The nonnegative value $u_0(i,j)$ is the upper bound of the arc (i,j) for $\lambda = 0$, i.e. : $\bar{u}(i,j;0) = u_0(i,j)$ with $0 \leq u_0(i,j)$. For the problem to be correctly formulated, the upper bound function of every arc $(i,j) \in A$ must respect the condition $0 \leq \bar{u}(i,j;\lambda)$ for the entire interval of the parameter values, i.e. $\forall (i,j) \in A$ and $\forall \lambda \in [0, \Lambda]$. It follows that the parametric part of the upper bounds $U(i,j)$ must satisfy the constraint: $U(i,j) \geq -u_0(i,j)/\Lambda$, $\forall (i,j) \in A$. The *parametric flow value function* $\bar{v} : N \times [0, \Lambda] \to \mathbb{R}$ associates to each of the nodes $i \in N$ a real number $\bar{v}(i;\lambda)$ referred to as the value of node i for each of the parameter λ values.

Definition 2. *[4]* A feasible flow in the parametric network $\bar{G} = (N, A, \bar{u}, s, t)$ is called a *parametric flow*, $\bar{f} : A \times [0, \Lambda] \to \mathbb{R}^+$ satisfying the following constraints:

$$\sum_{j|(i,j) \in A} \bar{f}(i,j;\lambda) - \sum_{j|(j,i) \in A} \bar{f}(j,i;\lambda) = \bar{v}(i;\lambda), \qquad \forall i \in N, \ \forall \lambda \in [0, \Lambda], \tag{4}$$

$$0 \leq \bar{f}(i,j;\lambda) \leq \bar{u}(i,j;\lambda), \qquad \forall (i,j) \in A, \ \forall \lambda \in [0, \Lambda], \tag{5}$$

where $\sum_{i \in N} \bar{v}(i;\lambda) = 0, \qquad \forall \lambda \in [0, \Lambda]$.

The parametric maximum flow (PMF) problem is to compute all maximum flows for every possible value of λ in $[0, \Lambda]$:

$$maximize \quad \bar{v}(\lambda) \quad for\ all \quad \lambda \in [0, \Lambda], \tag{6}$$

$$\sum_{j|(i,j)\in A} \bar{f}(i,j;\lambda) - \sum_{j|(j,i)\in A} \bar{f}(j,i;\lambda) = \begin{cases} \bar{v}(\lambda), & i = s \\ 0, & i \neq s,t \\ -\bar{v}(\lambda), & i = t \end{cases} \tag{7}$$

$$0 \leq \bar{f}(i,j;\lambda) \leq \bar{u}(i,j;\lambda), \qquad \forall (i,j) \in A. \tag{8}$$

This problem looks like a classic maximum flow problem with the decisive difference that the variables $\bar{f}(i,j;\lambda)$ of this problem are piecewise linear functions instead of real numbers and that the upper bounds $\bar{u}(i,j;\lambda)$ are linear functions instead of constants.

Definition 3. *[6] A parametric $s-t$ cut partitioning* denoted by $[S_k; J_k]$, $k = 0,\dots,K$, is defined as a finite set of cuts $[S_k, T_k]$, $k = 0,\dots,K$, together with a partitioning of the interval $[0,\Lambda]$ of the parameter in disjoints subintervals J_k, $k = 0,\dots,K$, so that $J_0 \cup \cdots \cup J_K = [0,\Lambda]$.

Definition 4. *[4]* For the parametric maximum flow problem, the *capacity $\tilde{\bar{c}}[S_k; J_k]$ of a parametric $s-t$ cut partitioning* is a linear function on every subinterval J_k, $k = 0,\dots,K$, defined as:

$$\tilde{\bar{c}}[S_k; J_k] = \sum_{(i,j)\in(S_k,T_k)} \bar{u}(i,j;\lambda), \quad k = 0,\dots,K. \tag{9}$$

Definition 5. *[4]* A parametric $s-t$ cut partitioning $[S_k; J_k]$ with the subintervals J_k assuring that every cut is a minimum cut $[\tilde{S}_k, \tilde{T}_k]$ within the subinterval $[\lambda_k, \lambda_{k+1}]$ is referred to as a *parametric minimum $s-t$ cut* and is denoted by $[\tilde{S}_k; J_k]$, $k = 0,\dots,K$.

Theorem 6. (Parametric Max-Flow Min-Cut Theorem [6]) *If there is a feasible flow in the parametric network $\bar{G}$, the value function $\tilde{\bar{v}}$ of the parametric maximum flow $\tilde{\bar{f}}$ from a source s to a sink t equals the capacity $\tilde{\bar{c}}$ of the parametric minimum $s-t$ cut $[\tilde{S}_k; J_k]$, $k = 0,\dots,K$.*

Let $\bar{f} = (\dots \bar{f}(i,j;\lambda), \dots)_{(i,j)\in A}$ be a vector of feasible flow functions defined on the interval $[0,\Lambda]$. Supposing that an arc $(i,j) \in A$ carries a flow $\bar{f}(i,j;\lambda)$, the existing flow can be increased either by sending the additional flow (*pushing*) $\bar{u}(i,j;\lambda) - \bar{f}(i,j;\lambda)$ from node i to node j over the arc (i,j) or by cancelling the flow $\bar{f}(j,i;\lambda)$ from node j to node i over the arc (j,i) which is equivalent to *pulling* the flow from node i to node j over the arc (j,i). These flows are computed as differences between piecewise linear functions of λ.

Definition 7. *[4]* For the parametric maximum flow problem, the *parametric residual capacity* $\tilde{\bar{r}}(i,j;\lambda)$ of any of the arcs $(i,j) \in A$, with respect to a given parametric flow $\bar{f}$, represents the maximum additional flow that can be sent from node i to node j over the arcs (i,j) and (j,i) and is given by:

$$\tilde{\bar{r}}(i,j;\lambda) = \bar{u}(i,j;\lambda) - \bar{f}(i,j;\lambda) + \bar{f}(j,i;\lambda). \tag{10}$$

Definition 8. *[4]* The subintervals $\tilde{I}(i,j) \subseteq [0,\Lambda]$ where an augmentation of the flow $\bar{f}(i,j;\lambda)$ is possible along the arc (i,j) are defined as follows:

$$\tilde{I}(i,j) = \{\lambda | \tilde{\bar{r}}(i,j;\lambda) > 0\}, \quad (i,j) \in A. \tag{11}$$

Definition 9. *[4]* Given a feasible flow $\bar{f}$ in the parametric network $\bar{G}$, the network denoted by $\tilde{\bar{G}}(\bar{f}) = (N, \tilde{A}(\bar{f}))$, with $\tilde{A}(\bar{f}) = \{(i,j) | (i,j) \in A,\ \tilde{I}(i,j) \neq \emptyset\}$ being the set consisting only of arcs with positive parametric residual capacities, is referred to as the *parametric residual network* with respect to the given flow $\bar{f}$ for the parametric maximum flow problem.

If an arc $(i,j) \in A$ does not belong to $\tilde{\bar{G}}(\bar{f})$ then $\tilde{I}(i,j) := \emptyset$ is set.

Definition 10. The parametric excess of a node $i \in N$ is defined as:

$$\tilde{e}(i;\lambda) = \sum_{j|(j,i)\in A} \bar{f}(j,i;\lambda) - \sum_{j|(i,j)\in A} \bar{f}(i,j;\lambda). \tag{12}$$

Definition 11. *[4]* The subintervals $\tilde{I}(i) \subseteq [0,\Lambda]$ where the excess of node i is positive are defined as follows:

$$\tilde{I}(i) = \{\lambda | \tilde{e}(i;\lambda) > 0\}, \quad i \in N - \{s,t\}. \tag{13}$$

In the residual network $\tilde{\bar{G}}(\bar{f})$ the *distance function* $\tilde{d} : N \rightarrow \aleph$ is a function from the set of nodes to the nonnegative integers. A distance function is said to be *valid* if it satisfies the following conditions: $\tilde{d}(t) = 0$ and $\tilde{d}(i) \leq \tilde{d}(j) + 1$, $\forall (i,j) \in \tilde{A}$.

Definition 12. *[4]* An arc $(i,j) \in \tilde{A}$ in the parametric residual network $\tilde{\bar{G}}(\bar{f})$ is referred to as *conditionally admissible* if both $\tilde{d}(i) = \tilde{d}(j) + 1$ and $\tilde{I}(i,j) \cap \tilde{I}(i) \neq \emptyset$; otherwise it is *conditionally inadmissible.*

3 Partitioning push algorithm

3.1 Highest-label partitioning-push algorithm

The *Highest-label partitioning-push* (HLPP) algorithm maintains a set L of active nodes organised as a priority queue. In the initialisation step of the algorithm, all the nodes $i \in N$ with $(s,i) \in \tilde{A}$ will gain a positive excess, becoming thus active nodes, by setting the flow to the upper bound value $\bar{f}(s,i;\lambda) := \bar{u}(s,i;\lambda)$ for every arc (s,i). Consequently, they are added to the priority queue L and then removed one by one, in the descending order of their priorities $\tilde{d}(i)$. For an active node $i \in \tilde{\bar{G}}(\bar{f})$, if there exists an conditionally admissible arc (i,j), the flow will be pushed over this arc and node j will be added to the priority queue L with the priority $\tilde{d}(j)$; otherwise node i will be relabelled so that at least one conditionally admissible arc to be created and node i is added to L with its new priority $\tilde{d}(i)$. The algorithm terminates when the queue of active nodes is empty. A push of flow from node i to node j is referred to as a *cancelling push* if it deletes the arc (i,j) from the residual network; otherwise it is a *non-cancelling push.*

Algorithm 1 Highest-label partitioning-push (HLPP) algorithm

```
procedure HLPP
    $J \leftarrow [0,\Lambda]$   $L \leftarrow \emptyset$   $\bar{f} \leftarrow 0$   compute $\tilde{d}(\cdot)$ in $\tilde{\bar{G}}(\bar{f})$
    for all $(s,i) \in \tilde{A}$ do
        $\bar{f}(s,i;\lambda) \leftarrow \bar{u}(s,i;\lambda)$   $\tilde{e}(i;\lambda) \leftarrow \bar{u}(s,i;\lambda)$
        if $\tilde{e}(i;\lambda) > 0$ and $i \neq t$ then
            add $i$ with priority $\tilde{d}(i)$ to $L$
        end if
    end for
    $\tilde{d}(s) \leftarrow n$
    PP$(L\ J)$
end procedure
```

For any node $i \in \tilde{\bar{G}}(\bar{f})$, the expressions: *active node* and *balanced node* holds only for subintervals of the parameter values. While both the parametric residual capacity $\tilde{\bar{r}}(i,j;\lambda)$ of any

Algorithm 2 Partitioning push (PP) procedure

```
procedure PP($L\ J_p$)
    if $L \neq \emptyset$ and $J_p \neq \emptyset$ then
        remove the first node $i$ from $L$
        if $\nexists$ an admissible arc $(i,j)$ then
            $\tilde{d}(i) \leftarrow min\{\tilde{d}(j) | (i,j) \in \tilde{A}\} + 1$
            add $i$ with priority $\tilde{d}(i)$ to $L$
            PP($L\ J_p$)
        else
            select an admissible arc $(i,j)$
            push $\tilde{g}(i,j;\lambda) \leftarrow min\{\tilde{e}(i;\lambda), \tilde{\bar{r}}(i,j;\lambda)\}$ over $(i,j)$
            if $j \notin L$ and $j \neq s$ and $j \neq t$ then add $j$ to $L$
            end if
            $J_{p_1} \leftarrow \{\lambda | \tilde{e}(i;\lambda) \leq \tilde{\bar{r}}(i,j;\lambda)\}$
            $J_{p_2} \leftarrow J_p - J_{p_1}$
            $L_{p_1} \leftarrow L \quad L_{p_2} \leftarrow L$
            add $i$ with priority $\tilde{d}(i)$ to $L_{p_2}$
            do in parallel
                PP($L_{p_1} J_{p_1}$)
                PP($L_{p_2} J_{p_2}$)
            end do
        end if
    end if
end procedure
```

arc $(i,j) \in \tilde{A}$ and the parametric excess $\tilde{e}(i;\lambda)$ of any node $i \in N - \{s,t\}$ are linear functions of the parameter values, cancelling or non-cancelling pushes are defined only for certain subintervals of the parameter values.

A non-cancelling push of flow from a node $i \in N - \{s,t\}$ along an arc $(i,j) \in \tilde{A}$ in a subinterval $J_p = (\lambda_p, \lambda_{p+1}] \subseteq [0, \Lambda]$ which leaves the node i unbalanced is referred to as a *partitioning push.* Whenever the algorithm performs a partitioning push in $\tilde{\bar{G}}_p(\bar{f})$, i.e. the network $\tilde{\bar{G}}(\bar{f})$ defined for the subinterval J_p, a new partitioning of the interval J_p in the two subintervals J_{p1} and J_{p2}, with $J_{p1} \cup J_{p2} = J_p$ and $J_{p1} \cap J_{p2} = \emptyset$ will take place. Let J_{p1} be the subinterval within which the partitioning push balances the node i, i.e. $J_{p1} = \{\lambda | \tilde{e}(i;\lambda) \leq \tilde{\bar{r}}(i,j;\lambda)\}$. If $J_{p2} \neq \emptyset$ then, as on every subinterval J_p both $\tilde{\bar{r}}(i,j;\lambda)$ and $\tilde{e}(i;\lambda)$ are linear functions of λ, the partitioning push generates two parametric residual networks: $\tilde{\bar{G}}_{p1}(\bar{f})$ for $\lambda \in J_{p1}$ and $\tilde{\bar{G}}_{p2}(\bar{f})$ for $\lambda \in J_{p2}$, so that node i is balanced in $\tilde{\bar{G}}_{p1}(\bar{f})$ and active in $\tilde{\bar{G}}_{p2}(\bar{f})$ while arc (i,j) will not belong to $\tilde{\bar{G}}_{p2}(\bar{f})$, since $\tilde{\bar{r}}(i,j;\lambda) = 0$ after the partitioning push. The algorithm will then continue separately in each of the parametric residual networks and for each of the two subintervals. Under these observations, the push/relabel procedure from the non-parametric *Highest label* preflow algorithm is replaced with a recursive call of a *partitioning push*(L,J) procedure.

Theorem 13. (Theorem of correctness) *Highest-label partitioning-push algorithm computes correctly a maximum flow in the parametric network* $\bar{G} = (N, A, \bar{u}, s, t)$.

Proof: The proof of the theorem follows from the correctness of the general HL preflow algorithm for each of the subintervals of the parameter values. When the algorithm terminates, let S_p be the set of all nodes which are reachable from the source node within the subinterval J_p. For the resulting cuts $[S_p, T_p]$ and the intervals $J_p = (\lambda_p, \lambda_{p+1}]$, $p = 1, \ldots, K$, the following observations

hold:
(i) If $i \in S_p$, $j \in T_p$ and $(i,j) \in A$ then $J_p \cap \tilde{I}(i,j) = \emptyset$ for otherwise node j could be reached from s in $\tilde{\bar{G}}(\bar{f})$. Hence, by the definition of $\tilde{I}(i,j)$, $\tilde{\bar{f}}(i,j;\lambda) = \bar{u}(i,j;\lambda)$, $\forall\lambda \in J_p$;
(ii) If $i \in T_p$, $j \in S_p$ and $(i,j) \in A$ then $J_p \cap \tilde{I}(j,i) = \emptyset$ for otherwise node i could be reached from s in $\tilde{\bar{G}}(\bar{f})$. Hence, $\tilde{\bar{f}}(i,j;\lambda) = 0$, $\forall\lambda \in J_p$.
Summarizing, $\sum_{(i,j)\in(S_p,T_p)} \tilde{\bar{f}}(i,j;\lambda) = \sum_{(i,j)\in(S_p,T_p)} \bar{u}(i,j;\lambda)$ and $\sum_{(i,j)\in(T_p,S_p)} \tilde{\bar{f}}(i,j;\lambda) = 0$, thus the obtained flow is a maximum parametric flow which equals the capacity of the minimum $s-t$ parametric cut. □

3.2 Complexity issues

A breakpoint is a place where the slope of the piecewise linear maximum flow value function is changing. In the worst case the number of breakpoints may be exponential in the size of the problem. The example originates from the pathological graph of Zadeh [8]. The Highest-label partitioning-push algorithm overcomes this inconvenient by using the multi-thread parallel implementation of a non-parametric algorithm [12]. The main idea of this implementation is to assign a processor to each newly generated subinterval Jp which will carry out the problem forward from the current configuration of the problem. For each of the newly generated subintervals, a copy of the current distance labels values is generated so that they can be independently modified in the further parallel evolution of the algorithm.

Theorem 14. (Theorem of complexity) *The parallel implementation of the Highest-label partitioning-push algorithm solves the parametric maximum flow problem in $O(n^2m^{1/2} + Kn)$ time.*

Proof: The complexity of the non-parametric HL preflow algorithm [1] is $O(n^2m^{1/2})$. The HL partitioning-push algorithm generates new copies of distance label values (one for each of the two new threads which will further run in parallel) every time a breakpoint occurs, i.e. copying distance labels takes $O(Kn)$ time where K is the number of breakpoints. Thus, the total complexity of the algorithm is $O(n^2m^{1/2} + Kn)$. □

4 Application to assessing the legal information

To write an appropriate, objective and specific rule for every imaginable situation is an impossible thing to do. "Given the numberless potential variations, foreseeable and unforeseeable, in motives and circumstances, there can, probably, be no end to the possible specific scenarios - and thus no limit on the number of rules that would result from trying to write an appropriate one for each possible, distinct fact situation." [11] The solution of many complex decision problems, such as assessing legal information, involves combinatorial optimization, i.e. obtaining the optimal solution among a finite set of alternatives. Such optimization problems are notoriously difficult to solve. One of the primary reasons is that in most applications the number of alternatives is extremely large and only a fraction of them can be considered within a reasonable amount of time. As a result, heuristic algorithms, such as evolutionary algorithms are often applied in combinatorial optimization but their major problem consists in their high complexity.
In our opinion, the proposed algorithm can successfully be used in assessing legal information, in their semantic evaluation or in generating files or legal information recording models and subsequently transmitting them to the members of the information society.

4.1 Concepts association network

Regarding the law as a mathematical phenomenon, the legislation represents a logical structure. Any law can be considered as a collection of fractal scenarios, made up of legal stipulations and exceptions. In order to enhance assessing features, large collections of legal information must be organized in hierarchical structures based on key concepts. Generally, there are two approaches to hierarchical structures generation: the query-independent approach, offering a consistent view of the whole corpus, and the query-dependent approach which allows concepts to be organized differently depending on the query. The approach presented in our paper is a query-dependent one, based on statistical co-occurrence in determining the relationship between concepts. The algorithm builds a hierarchical structure where, for a certain topic, each of the main concepts is related to distinct sub-concepts with different degrees of association values, described by the parameter values.
The first stage in generating a concept hierarchy consists in extracting a set of main concepts, related to the topic, and computing the degree of co-occurrences in relation with other sub-concepts. Once the set of main concepts is extracted and their degrees of co-occurrences are calculated, the concepts association network, covering all concepts, is constructed. The network is built as a directed graph with weighted arcs, where each of the concepts represents a node and the source node represents the topic of the query-dependent structure. The directed graph contains an arc between two concepts only if those concepts co-occur in at least one legal stipulation.

Definition 15. The *degree of co-occurrence* is computed as the number of legal articles (stipulation) that contain (refer to) both concepts.

Definition 16. The *strength* of a concept, representing that concept's importance, is the sum of all its co-occurrences.

The capacities of the arcs (s, i) are set to $u(s, i) := strength(i)$ while all the arcs (j, t) have no upper value limit.

Definition 17. The parametric upper bound $\bar{u}(i, j; \lambda)$ of an arc (i, j) is computed as:

$$\bar{u}(i, j; \lambda) = \frac{strength(i)}{externaldegree(i)} + \lambda \cdot co - occurrence(i, j). \tag{14}$$

Finally, based on the computed parametric maximum flow in the concepts association network, the sets of concepts defined by the parametric cut partitioning group the concepts in classes which are ordered in hierarchies. As long as the sink node is reached in the concepts association network, any of the directed paths starting from the source node represents a legal demarch. In the case that for various reasons the sink node can not be reached, the meaning of this fact is that some legal provisions are contradictory, inconsistent or ambiguous.

4.2 Theoretical legal aspects

Let us analyze, for example, the general provision of the Romanian Constitution, namely Article 21, first alignment "any person can appeal to the court for protecting its legitimate interests and liberties" and the second alignment "no law can restrict the exercise of these rights". Among the main concepts with which the lawmaker operates, we can name: the free access to justice, rights, liberties, legitimate interests. These concepts are in close connection to other sub concepts which are not currently found in laws. Thus, according to Article 192 of the Civil Procedure Code, "in order to protect one's legitimate interests and rights, any person can

appeal to justice by approaching the competent court of law".
However, there are some exceptions to the general rule stated above, namely Article 193, the first alignment of the Civil Procedure Code, according to which "approaching the court is an action which can occur only after a preliminary procedure is completed if the law is clear on that matter. The proof of completing this procedure will be attached to the summon", as well as the second alignment of the same law, according to which "failure to complete the preliminary procedure can only be invoked by the defendant in his response". To complete these provisions, we have the first alignment of Article 7 of Law no 554/2004 regarding the administrative procedures which state the following "before addressing the competent legal court, the person who claims that it has suffered an injustice through an unilateral administrative act must first ask the public authority, within 30 days, to revoke that certain document or some parts of it (...)".

4.3 Legal case study

First Court solution

To demonstrate the theoretical aspects discussed above, we will use the following example:
By their petition addressed to the Galati County Appeal Court, the plaintiffs R.I., B.S., G.M., S.C., G.V., S.G., R.V., S.C., A.C.E., G.C., S.A., B.I., E.A., G.G., G.G versus the Romanian Government, The Galati County Pension Institution and the Romanian Council for Fighting Discrimination have asked the annulment of the Government's Decision no 737/2010 regarding the method used for recalculation the amount of retirement money for the categories stated in Article 1, letters c to h of Law no 119/2010 regarding the establishing of some measures for maintaining the amount of retirement money, published in the Official Bulletin no 528 of June 29th, 2010, which the plaintiff was receiving at the date the document whose annulment is asked came into force. By the same petition, they asked, according to the provisions of Article 15 of Law no 554/2001 the suspension of the Government's Decision no 737/2010 until this matter is resolved. By its response, the Romanian Government claimed the petition was inadmissible as it did not complete the preliminary procedure, according to Article 7 of Law no 554/2004. It further asked for the petition to be dismissed as it is not insubstantial. The Romanian Ministry for Work, Family and Social Protection intervened on behalf of the defendant - the Romanian Government. By its decision no 52 of February 22nd, 2011, the Galati County Appeal Court, dismissed the plaintiffs petition as inadmissible and allowed the Ministry to intervene on behalf of the Romanian Government.
In order to reach this decision, the court stated that the provisions of Article 7, the first alignment claim that the legal proceedings can only occur after the administrative authorities are given the opportunity to revoke their document or amend it. In the matter at hand, although it was claimed that the Government's Decision no 737/2001 regarding the method used for recalculating the amount of retirement money for the categories stated in Article 1, letters c to h of Law no 119/2010 regarding the establishing of some measures for maintaining the amount of retirement money, should be annulled, there was no proof that a preliminary proceeding existed, as this was a legal condition for the admissibility of the petition, according to Article 109, second alignment of the Civil Procedure Code.

Appeal Court solution

The plaintiffs appealed this decision, by criticizing it for being illegal and unfounded and claiming that, in the matter at hand, the preliminary procedure can be completed at any time and this condition was met. The High Court of Justice, by analyzing the works and documents of this case, ruled that the first court was correct to dismiss the plaintiff's petition as inadmissible, as it does not respect the provisions of the first alignment of Article 7 of Law no 554/2004. According to these provisions "before addressing the court, the person who claims one of its rights was disrespected must first ask the public authority who created the document, within 30 days of that date, to revoked that document. The petition can also be addressed to the superior hierarchical organ, if such an organ exists". Respecting the terms and conditions stated by law for the preliminary procedure is a special demand of the law and the disrespecting of this demand causes the plaintiffs inability to appeal the courts. In the matter at hand, the plaintiffs asked for the annulment of Government's Decision no 737/2010 without proving they have completed the preliminary proceedings. This is similar to not respecting the legal obligation enforced by the first alignment of Article 7 of Law no 554/2004. Indeed, according to the provisions of Article 7, the first alignment of Law no 554/2004, "in case of an administrative act, the preliminary procedure can be completed at any time", as the Government's Decision no 737/2010 is considered to be an administrative act with power of the law.

However, these legal provisions must be interpreted in close connection with those of Article 11 of the Administrative Proceedings Law which describes the term to formulate such a demand, as well as with the provisions of Article 7, the first alignment of the same law. The phrase "at any time" allows for the possibility of formulating such a petition and for completing the preliminary proceedings without respecting the term stated in Article 11 of Law no 554/2010, except for "ordinances of their dispositions which are considered to be unconstitutional, as well as administrative documents with power of the law which are considered to be illegal", as this course of action is regulated by the fourth alignment of this Article.

The preliminary procedure regulated by Article 7, alignments (1) and (11) of Law no 554/2004, is a condition for the admissibility of the petition, according to Article 109, second alignment of the Civil Procedure Code, as it is previous to appealing the court. Considering all aspects mentioned above and seeing that there are no reasons to annul the first court's decision, the High Court will dismiss the appeal as unfounded [13].

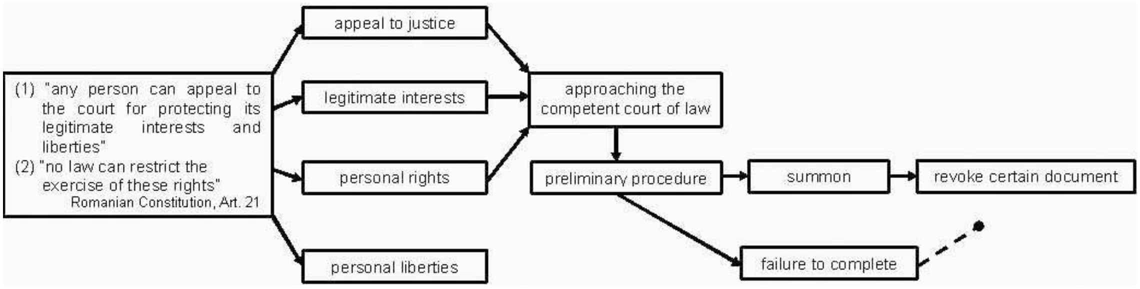

(1) "any person can appeal to the court for protecting its legitimate interests and liberties"
(2) "no law can restrict the exercise of these rights"
Romanian Constitution, Art. 21
appeal to justice
legitimate interests
personal rights
personal liberties
approaching the competent court of law
preliminary procedure
summon
revoke certain document
failure to complete

Figure 1: Concepts association network for the presented legal example.

As can be easily seen in the concepts association network presented above in Fig.1, by following the directed path which does not include the preliminary procedure attached to a summon, the sink node can never be reached, revealing the fact that some legal provisions are contradictory, inconsistent or ambiguous.

Further developing the above presented model, based on an appropriate corpus (database) of legal stipulations which is generated by our algorithm in the way it has previously been presented above already, a computer application can be developed so that to any online transmitted legal question, a solicitor could receive the adequate legal information.

5 Conclusions

The maximum flow problem in parametric networks turns out to be an important scenario in practice since the complexity of its solving algorithm was reduced to a linear dependency of the number of breakpoints. The present article presents the state-of-the-art of the approaches for solving the parametric maximum flow problem and presents an original parametric preflow algorithm, based on network partitioning technique. After presenting the basic parametric network flow terminology adapted for the parametric network with linear capacity functions and zero lower bounds, the proposed *highest label partitioning push (HLPP) algorithm* is described in details, being accompanied by the corresponding theorems of correctness and of complexity of the algorithm. In one of its final sections, the article also proposes a way of implementation of our algorithm in the legislation domain. The Definitions (12-14) and interpretations contained in this section are also original contributions of the authors.
Moreover, the given example shows the way the proposed algorithm progressively generates hierarchical structures of legal articles (stipulations), gathered according to their relevance in explaining a general legal topic (or legal problem). Using suitable file structures for legal information recording, these sets of legal stipulations can be transmitted to any user (or solicitor).

Bibliography

[1] Ahuja, R., Magnanti, T. and Orlin, J.(1993); *Network Flows. Theory, algorithms and applications*, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1993.

[2] Bichot, C-E., Siarry, P. (2011), *Graph Partitioning: Optimisation and Applications*, ISTE Wiley.

[3] Hamacher, H.W. and Foulds, L.R. (1989); Algorithms for flows with parametric capacities, *ZOR - Methods and Models of Operations Research*, 33:21–37.

[4] Parpalea, M., Ciurea, E.(2013); Partitioning Algorithm for the Parametric Maximum Flow, *Applied Mathematics (Special Issue on Computer Mathematics)*, 4(10A):3–10.

[5] Parpalea, M.M. (2009);German Word Order, *Bulletin of the Transilvania University of Brasov*, Series IV, 2(51): 175–182.

[6] Parpalea, M. (2010); Min-Max algorithm for the parametric flow problem, *Bulletin of the Transilvania University of Brasov*, Series III: Mathematics, Informatics, Physics, 3(52):191–198.

[7] Parvan, M., Ghionea, F., Flaut, C. (2008); A mathematical model of the relevant request determination in the transport problem, Mathematics and Computer in Business and Economics, *Proc.of the 9th WSEAS Intl. Conf. on Mathematics and Computer in Business and Economics (MCBE'08)*, Bucureşti, 24-26 iunie 2008, 105–111.

[8] Ruhe, G. (1988); Complexity Results for Multicriterial and Parametric Network Flows Using a Pathological Graph of Zadeh, *Zeitschrift fur Oper. Res.*, 32: 9–27.

[9] Ruhe, G. (1985); Characterization of all optimal solutions and parametric maximal flows in networks, *Optimization*, 16(1): 51–61.

[10] Sângeorzan, L., Parpalea, M., et al. (2010); Partitioning Preflow-pull Algorithm for the Parametric Net-work Flow Problem - a Linguistic Rule-based Constraints Optimisation

Approach, Recent Advances in Neural Networks, Fuzzy Systems and Evolutionary Computing, *Proc.of the 11th WSEAS International Conference on Neural Networks*, Iasi, Romania, 111–116.

[11] Stumpff, A.M. (2013); The Law is a Fractal: The Attempt to Anticipate Everything, *Loyola University Chicago Law Journal*, 44.

[12] Yong, C., Chang-le, Lu (2006); Using Multi-Thread Technology Realize Most Short-Path Parallel Algorithm, *World Academy of Science, Engineering and Technology*, 15: 11–13.

[13] *High Court's Decision no 656/2012*, given in a public hearing on February 9th, 2012, in the file no 1259/44/2010, http://www.iccj.ro/cautare.php?id=68146, accessed September 11th, 2014, at 13,40.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):126-141, February 2016.

Computing Sorted Subsets for Data Processing in Communicating Software/Hardware Control Systems

V. Sklyarov, I. Skliarova, A. Rjabov, A. Sudnitson

V. Sklyarov*, I. Skliarova
University of Aveiro, Dept. of Electronics, Telecommunications and Informatics/IEETA
Campus Universitário de Santiago, 3810-193, Aveiro, Portugal
*Corresponding author: skl@ua.pt

A. Rjabov, A. Sudnitson
Tallinn University of Technology, Dept. of Computer Engineering
Akadeemia tee 15A, 12618, Tallinn, Estonia

Abstract: Computing and filtering sorted subsets are frequently required in statistical data manipulation and control applications. The main objective is to extract subsets from large data sets in accordance with some criteria, for example, with the maximum and/or the minimum values in the entire set or within the predefined constraints. The paper suggests a new computation method enabling the indicated above problem to be solved in all programmable systems-on-chip from the Xilinx Zynq family that combine a dual-core Cortex-A9 processing unit and programmable logic linked by high-performance interfaces. The method involves highly parallel sorting networks and run-time filtering. The computations are done in communicating software, running in the processing unit, and hardware, implemented in the programmable logic. Practical applications of the proposed technique are also shown. The results of implementation and experiments clearly demonstrate significant speed-up of the developed software/hardware system comparing to alternative software implementations.

Keywords: computing sorted subsets, communicating hardware/software systems, filtering, sorting networks, control applications.

1 Introduction

Many electronic, environmental, medical, and biological control applications need to process data streams produced by sensors and measure external parameters within given upper and lower bounds (thresholds) [1]. Let us consider some examples. Applying the technique [2] in real-time applications requires knowledge acquisition from controlled systems (e.g. plant). For example, signals from sensors may be filtered and analyzed to prevent error conditions (see [2] for additional details). To provide more exact and reliable conclusion, combination of different values need to be extracted, ordered, and analyzed. Similar tasks appear in monitoring thermal radiation from volcanic products [3], filtering and integration of information from a variety of different sources in medical applications [4] and in other practical applications described in [5]. Since many control systems are real-time, performance is important and hardware accelerators may provide significant assistance for software. A similar data processing is applicable to data mining algorithms, such as [6].

Let us consider control systems that collect, filter and analyze data produced by some measurements. We will describe below such computations that permit:

- the maximum and/or minimum sorted subsets to be extracted (the maximum/minimum sorted subset of size $\mathrm{L_{max}}/\mathrm{L_{min}}$ contains $\mathrm{L_{max}}/\mathrm{L_{min}}$ data items with maximum/minimum values from a given set);
- the maximum and/or minimum sorted subsets to be found within the given upper $\mathrm{B_u}$ and lower $\mathrm{B_l}$ bounds.

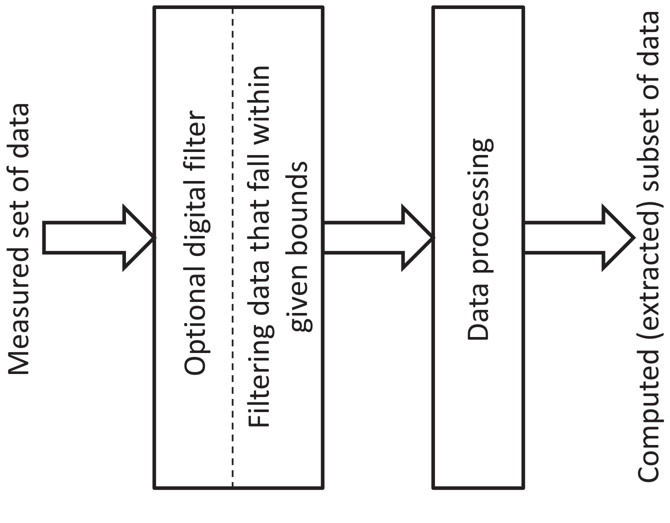

Measured set of data
Optional digital filter
Filtering data that fall within given bounds
Data processing
Computed (extracted) subset of data

Figure 1: General architecture of data processing

The problem can be solved as it is shown in Fig. 1.

There are two blocks in Fig. 1. Measured data items are handled in such a way that the maximum and/or minimum subsets with L_{max} and/or L_{min} items are extracted by the data processing block. Input data may optionally be filtered allowing only items (such as D) that fall within pre-given constraints (e.g. $B_l \leq D \leq B_u$ or $B_l < D < B_u$) to be processed.

The paper suggests a method and high-performance implementation of architecture in Fig. 1 in all programmable systems-on-chip (APSoC) from the Xilinx Zynq-7000 family [7] that are recently developed field-configurable devices integrating the most advanced programmable logic (PL) and a widely used processing system (PS) based on the dual-core ARM® Cortex™ MPCore™. The available interfaces between the PS and PL are supported by ready-to-use intellectual property (IP) cores. These, combined with numerous architectural and technological advances, have enabled APSoCs to open a new era in the development of highly optimized computational systems for a vast variety of practical applications, including high-performance computing, data, signal and image processing, control, and many others. The main target of APSoCs is integration in the developed systems of software and hardware components assuming that such integration enables characteristics (most often performance) of the system to be improved. The complexity of hardware only solutions is frequently limited by the available resources in the PL. Software/hardware solutions can be very complex and they are appropriate for control applications, such as that are described, for example, in [2,4]. The most close related work can be found in [5,8] where the importance of the considered problem is underlined, but the methods that allow the problem to be solved are different and the proposed below methods permit better results to be achieved.

The remainder of the paper is organized in eight sections. Section 2 presents the proposed software/hardware architecture. Section 3 describes a novel method allowing the maximum and minimum sorted subsets for a given set of data items to be computed. Section 4 suggests a run-time filtering method. Section 5 is dedicated to on-chip communication mechanisms linking software and hardware components. Section 6 shows how large subsets (for which hardware resources are not sufficient) can be computed and discusses additional capabilities such as extracting only the maximum or only the minimum subsets. Section 7 demonstrates potential practical application (from the areas of control and data mining). Implementations in Zynq microchips and the results of thorough evaluation and comparison of software only and software/hardware solutions with explicit indication of the achieved acceleration are discussed in section 8. Section 9 concludes the paper.

2 Software/Hardware Architecture

Fig. 2 presents the proposed software/hardware architecture.

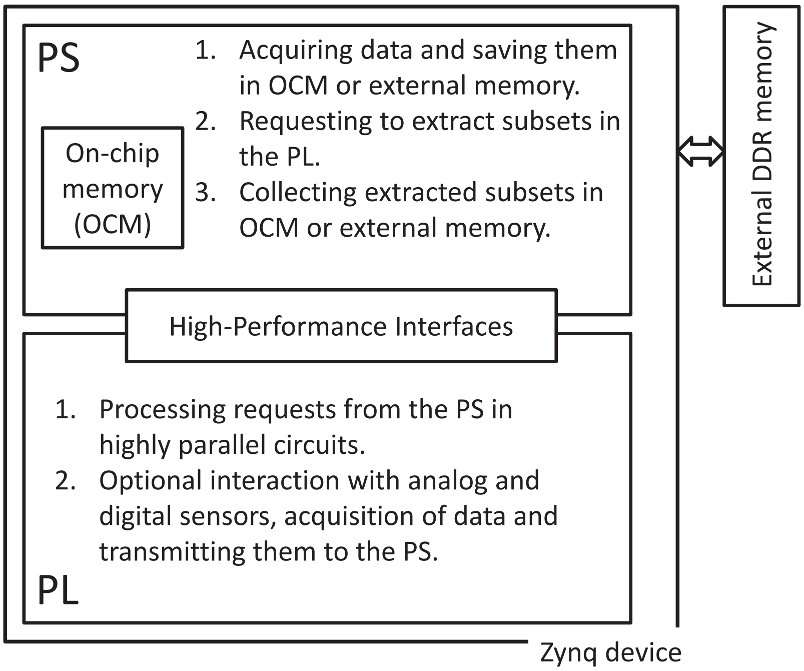

PS
1. Acquiring data and saving them in OCM or external memory.
2. Requesting to extract subsets in the PL.
3. Collecting extracted subsets in OCM or external memory.
On-chip memory (OCM)
High-Performance Interfaces
1. Processing requests from the PS in highly parallel circuits.
2. Optional interaction with analog and digital sensors, acquisition of data and transmitting them to the PS.
PL
Zynq device
External DDR memory

Figure 2: The proposed software/hardware architecture

The PS collects data, that may be acquired from different sources (such as from a host PC or from sensors connected to a Zynq device), and stores them in on-chip or external memory. The PL processes requests from the PS, that is, reads data from memories, and rapidly extracts the maximum and/or minimum subsets. Both parts, that are the PS and PL, may function in parallel and any request can be seen as a macroinstruction executed in the PL concurrently with other potential instructions in the PS.

It is shown in [9] that for transferring a small number of data items between the PS and the PL on-chip general-purpose ports (GPP) can be used more efficiently than other available interfaces. Thus, requests from the PS to the PL are formed through GPP where the PS is the master and the PL is a slave. It is also shown in [7, 9] that large volumes of data can be more efficiently transferred from/to memories to/from the PL through high performance (HP) interfaces: High-Performance Advanced eXtensible Interface (AXI HP) and AXI Accelerator Coherency Port (AXI ACP). In all our designs memories are slaves and either the PL or the processor in the PS is the master. To increase performance, data from memories may be requested to be cacheable.

3 Computing Sorted Subsets

Let set S containing N M-bit data items be given. The maximum subset contains L_{max} largest items in S and the minimum subset contains L_{min} smallest items in S ($L_{max} \leq N$ and $L_{min} \leq N$). We mainly consider such tasks for which $L_{max} << N$ and $L_{min} << N$ which are more common for practical applications. Since N may have very large value (millions of items) it cannot be processed completely in hardware due to the unavailability of sufficient resources. It is shown in [10, 11] that even for relatively complex Field-Programmable Gate Arrays (FPGAs) the size

N is limited. For example, for even-odd merge and bitonic merge networks [12] N cannot exceed a few hundreds of 32-bit items even for very advanced FPGAs (such as the largest devices from the Xilinx Virtex-7 family). In Zynq devices implementing circuits from [12] the maximum value of N does not exceed 128 32-bit items. Iterative even-odd transition networks from [11] permit significantly larger number of items (exceeding thousands of 32-bit items) to be processed and they will be used for computing sorted subsets in hardware. However, in practical cases the given sets anyhow cannot be entirely processed and computing the maximum and/or minimum sorted subsets needs to be done sequentially, nevertheless handling many items in parallel. Fig. 3 depicts the proposed architecture that enables the considered problem to be solved.

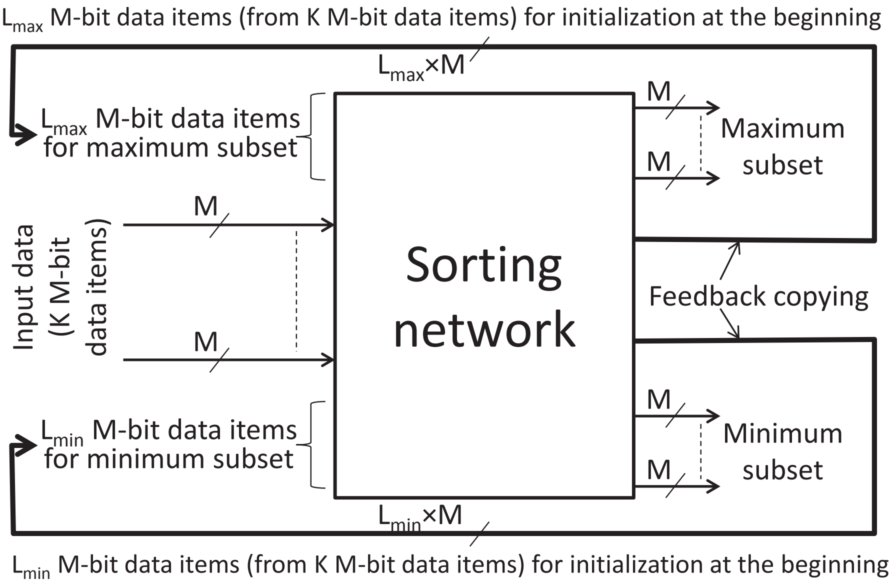

L$_{max}$ M-bit data items (from K M-bit data items) for initialization at the beginning
L$_{max}$×M
L$_{max}$ M-bit data items for maximum subset
M
M
Maximum subset
Input data (K M-bit data items)
M
M
Sorting network
Feedback copying
L$_{min}$ M-bit data items for minimum subset
M
M
Minimum subset
L$_{min}$×M
L$_{min}$ M-bit data items (from K M-bit data items) for initialization at the beginning

Figure 3: Computing the maximum and the minimum sorted subsets

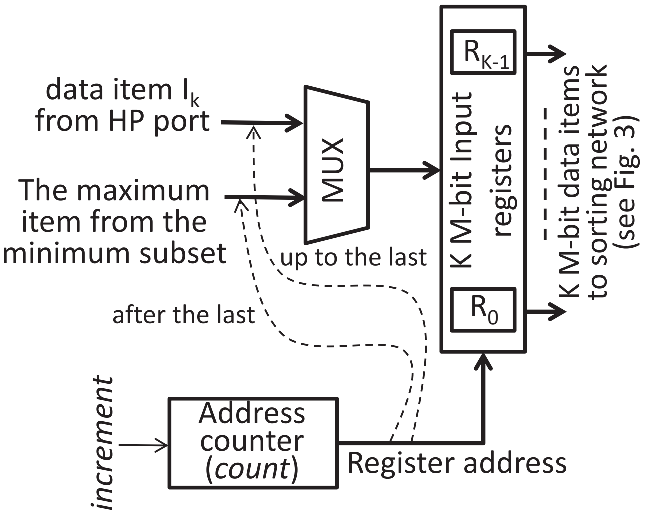

data item I$_k$ from HP port
The maximum item from the minimum subset
MUX
up to the last
after the last
K M-bit Input registers
R$_{K-1}$
R$_0$
K M-bit data items to sorting network (see Fig. 3)
increment
Address counter (count)
Register address

Figure 4: Processing the last (possibly incomplete) subset

Let us divide the given set S into Q = ⌈N/K⌉ subsets, all of which contain exactly K M-bit items except the last one, which may have less than K M-bit items. Computing subsets is done incrementally in Q steps (we assume below that K $\leq$ N).

At the first step, the first K M-bit data items are sorted in the network [11] which processes L_{max}+K+L_{min} data items but comparators linking the upper part (handling L_{max} M-bit data items) and the lower part (handling L_{min} M-bit data items) are deactivated (i.e. the links with the upper and bottom parts are broken). So, sorting is done only in the middle part handling K M-bit items. As soon as the sorting is completed, the maximum subset is copied to the upper part of the network and the minimum subset is copied to the lower part of the network (see Fig. 3).

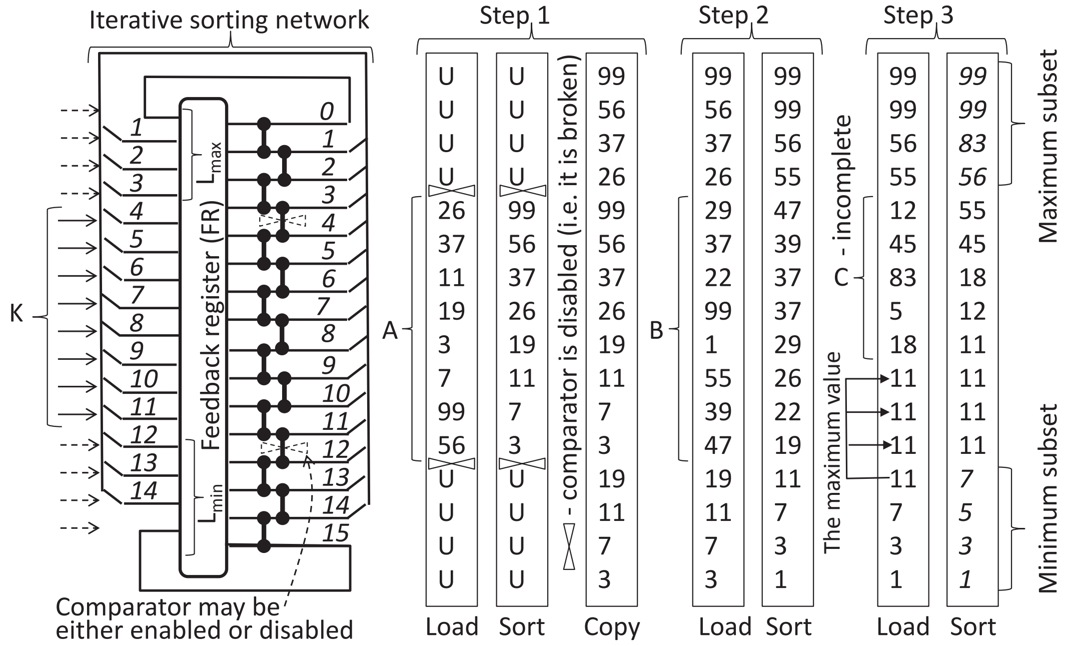
From the second step, all the comparators are properly linked, i.e. the network from [11] handles L_{max}+K+L_{min} items, but the feedback copying (see the first step and Fig. 3) is disabled. Now for each new K M-bit items the maximum and the minimum sorted subsets are appropriately corrected, i.e. new items may be appended.

At the last step, the number of incoming items may be less than K. Fig. 4 explains how the maximum and minimum subsets are corrected for the last possibly incomplete subset of items. There is an additional MUX in Fig. 4, which supplies data items from a HP port (linking the PL with memory) until the received item is not the last. As soon as the last item is read from memory, the next items (until K) are taken as the maximum value from the minimum subset (see the lower subset in Fig. 3). Clearly, such an item cannot be moved again to the minimum subset and the last sorting step is executed similarly to the previous steps.

Let us look at the example shown in Fig. 5 for which: N = 21, K = 8, L_{max} = L_{min} = 4, and S = 26,37,11,19,3,7,99,56,29,37,22,99,1,55,39,47,12,45,83,5,18. The set S is divided into the following three subsets: A = 26,37,11,19,3,7,99,56, B = 29,37,22,99,1,55,39,47, and C = 12,45,83,5,18.

Note that the last subset C contains only 5 elements and is incomplete. Symbol U in Fig. 5 indicates undefined value. The iterative sorting network is exactly the same as in [11]. Any comparator is shown in Knuth notation [13] and it converts two-item inputs in two-item outputs in such a way that the upper value is greater than or equal to the lower value. The maximum number of iterations for sorting is K/2 [14] and this number is almost always smaller because the method [11] terminates subsequent iterations as soon as all items are sorted. There are 3 steps in Fig. 5. At the first step, K (K=8) items are sorted and copied to the maximum and minimum subsets.

Two comparators are disabled in accordance with the explanations given above (breaking links of the middle section in the sorted network with the upper and the lower sections). At the second step, all the network comparators are enabled and L_{max}+K+L_{min} items are sorted by the iterative network with feedback register (FR). All necessary details can be found in [11]. It is easy to show that the maximum number of iterations is $\lceil(\max(L_{max},L_{min})+K)/2\rceil$ and much like the previous case this number is almost always smaller [11]. At the last (third) step, the incomplete subset C is extended to K items by copying the maximum value (11) from the minimum subset 11,7,3,1 to the positions of missing data (see Fig. 5). After sorting L_{max}+K+L_{min} items at the step 3 the final result is produced.

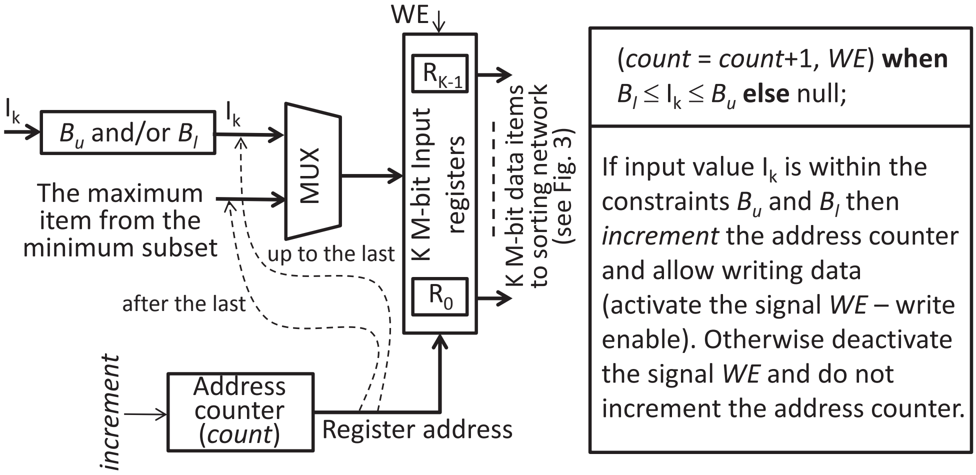


Step 1			Step 2		Step 3	
Load	Sort	Copy	Load	Sort	Load	Sort
U	U	99	99	99	99	99
U	U	56	56	99	99	99
U	U	37	37	56	56	83
U	U	26	26	55	55	56
26	99	99	29	47	12	55
37	56	56	37	39	45	45
11	37	37	22	37	83	18
19	26	26	99	37	5	12
3	19	19	1	29	18	11
7	11	11	55	26	11	11
99	7	7	39	22	11	11
56	3	3	47	19	11	11
U	U	19	19	11	11	7
U	U	11	11	7	7	5
U	U	7	7	3	3	3
U	U	3	3	1	1	1

Figure 5: An example (computing subsets)

4 Filtering

Let B_u and B_l be predefined upper (B_u) and lower (B_l) bounds for the given set S. We would like to use the circuit in Fig. 3 only for such data items D that fall within the bounds B_u and B_l, i.e. $B_l \leq D \leq B_u$ (or, possibly, $B_l < D < B_u$). Fig. 6 depicts the proposed architecture that enables data items to be filtered at run-time (i.e. during the data exchange between the PS and PL). There is an additional block on the upper input of the MUX (see also Fig. 4), which takes a data item I_k from a HP port and executes the operation indicated on the right-hand part of Fig. 6. If the counter is incremented, then a new register is chosen to store I_k. Otherwise, the signal WE (write enable) is passive and a new item with a value that is out of the bounds B_u and B_l is not recorded in the registers.



(*count* = *count*+1, *WE*) **when** $B_l \leq I_k \leq B_u$ **else** null;

If input value I_k is within the constraints B_u and B_l then *increment* the address counter and allow writing data (activate the signal *WE* – write enable). Otherwise deactivate the signal *WE* and do not increment the address counter.

Figure 6: Digital filter

Let us look at the same example in Fig. 5 for which we choose $\mathrm{B_u} = 90$ and $\mathrm{B_l} = 10$ (see Fig. 7). At the first step incoming data items have preliminary been filtered, the values 99, 7, and 3 have been removed (because they are either greater than $\mathrm{B_u} = 90$ or less than $\mathrm{B_l} = 10$), and the subset A with 8 items is built from 11 first elements of the set S. At the second (last) step, the values 99, 1, and 5 have been removed, and the subset B = 55,39,47,12,45,83,18 is built from the remaining allowed elements of the set S. Since there are 7 items in B and K = 8, this subset is incomplete.

As can be seen from Fig. 7, two steps are sufficient to extract the maximum and the minimum subsets from the filtered set S. Similarly, filtering and computing sorted subsets can be done for very large data sets.

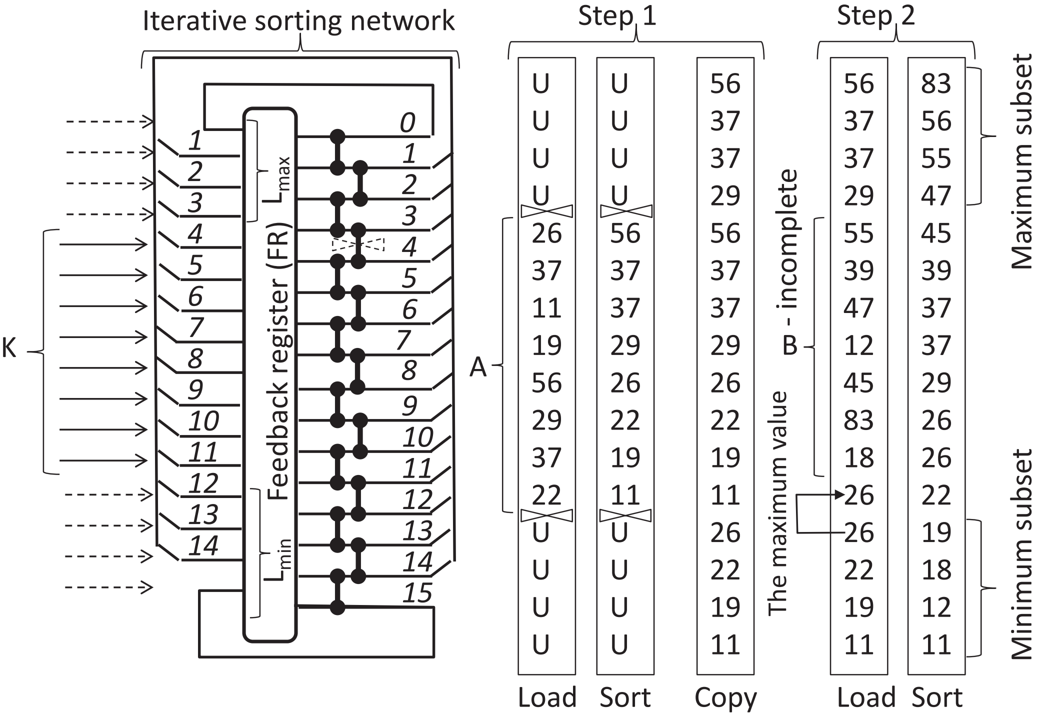


Figure 7: An example (filtering and computing subsets)

Clearly, the described above operations can be done in software. For example, C function qsort permits large data sets to be sorted. After that extracting the maximum and minimum subsets may easily be done. Filtering can be provided much like it is shown in Fig. 6 eliminating items that do not fall within the predefined constraints. However, for many practical applications performance of the described above operations is important. To evaluate software/hardware solutions three different components need to be taken into account (see Fig. 8): 1) software part; 2) hardware part; and 3) the circuits that provide for data exchange between software and hardware. Numerous experiments were done in [15] to compare such solutions with software only systems. One example in [15] enables sorting blocks of data composed of 320 32-bit items in the PL that are further merged in the PS (see Fig. 8). From 512,288 to 4,194,304 of 32-bit data items were randomly generated in the PS (i.e. the size of data varies from 2MB to 16MB) and then sorted in software with the aid of the function qsort and in the software/hardware system (see Fig. 8). The actual performance improvement was by a factor of about 2.5. It was shown in [15] that hardware circuits in the PL are significantly faster than software in the PS. In this paper we evaluate and compare software/hardware and software only solutions taking into account all the involved communication overheads that were measured in [15]. We will mainly use AXI ACP [7] which provides one of the fastest interfaces for exchange of large data sets between the PS and PL [7,9,15]. The number of data items transferred from the PS/memory to the PL is the same as in [15]. However, the number of data items transferred from the PL to the PS/memory is significantly smaller enabling much better acceleration to be achieved.

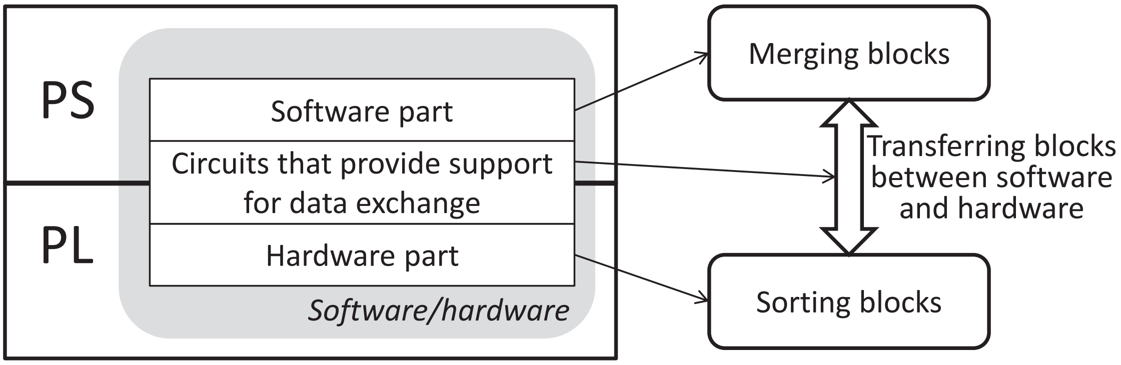

PS
PL
Software part
Circuits that provide support for data exchange
Hardware part
Software/hardware
Merging blocks
Transferring blocks between software and hardware
Sorting blocks

Figure 8: An example of a software/hardware system

5 Communication of Software and Hardware

Fig. 9 shows how communication is organized between software and hardware. It is done similarly to [8,15], but the proposed in this paper processing is different. The developed hardware in the PL is divided in two parts: application-specific (that is filtering and computing subsets) and communication-specific processing. The latter is studied in [15] and provides support for data exchange with storage of the PL that is either block RAM or registers built from flip-flops of configurable logic blocks.

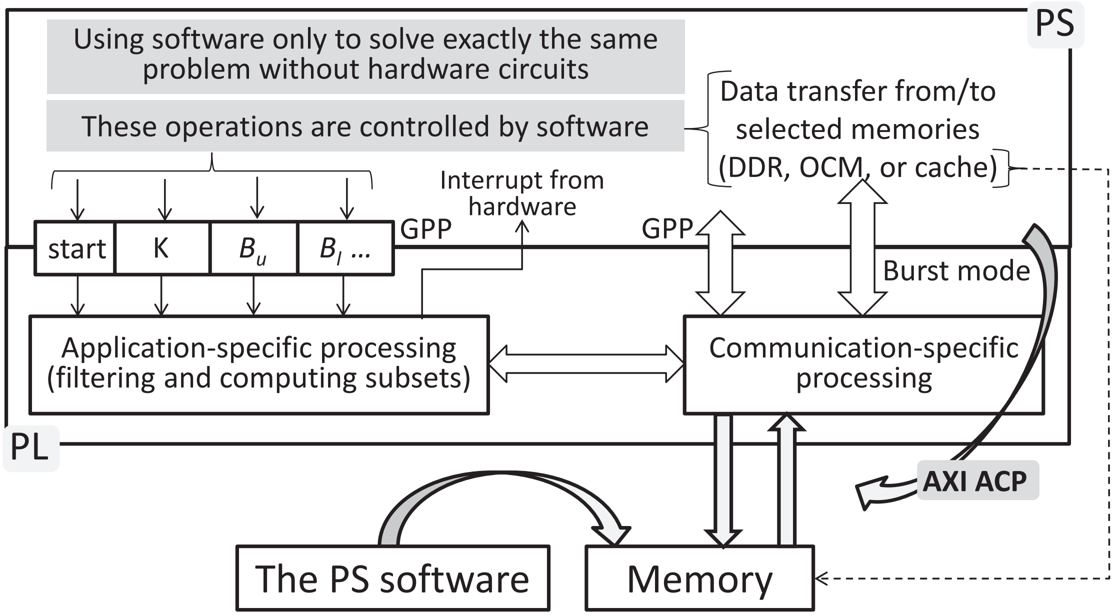

PS
Using software only to solve exactly the same problem without hardware circuits
These operations are controlled by software
Data transfer from/to selected memories (DDR, OCM, or cache)
Interrupt from hardware
GPP
GPP
start
K
B_u
B_l ...
Burst mode
Application-specific processing (filtering and computing subsets)
Communication-specific processing
PL
AXI ACP
The PS software
Memory

Figure 9: Communication between software and hardware components

Data are transmitted in blocks of 32/64-bit items (i.e. either M=32 or M=64) and the fastest burst mode is applied. Input data items I_k (k=0,1,...,K-1) are processed by the described above circuits (see Fig. 3, 4, 6). Note that GPP do not allow burst mode to be applied but are very appropriate for transferring small number of signals that may be used for control in the PL and for some additional details. In our system they are:

1) start requiring data processing in the PL to be initiated;

2) K (see Fig. 3);

3) B_u and B_l (see Fig. 6);

4) additional signals, namely source address, destination address and sizes of data to be transferred from the PS to the PL and vice versa.

Fig. 10 demonstrates a component diagram for reading the initial large volume data from the PS and for transferring the results (i.e. the computed maximum and minimum subsets) from the PL to the PS. We found that in such interactions between the PS and PL the best way is to use HP ports to read data from the PS (memories) and transfer them to the PL and to write data from the PL to the PS (memories). Since memory controllers belong to the PS we can talk about data transfers between the PS and PL. Exchange of data in both directions is done in burst mode supported by a burst reader and a burst writer described in [8,15]. Both processing (T_h) and communication (T_c) times are measured and taken into account.

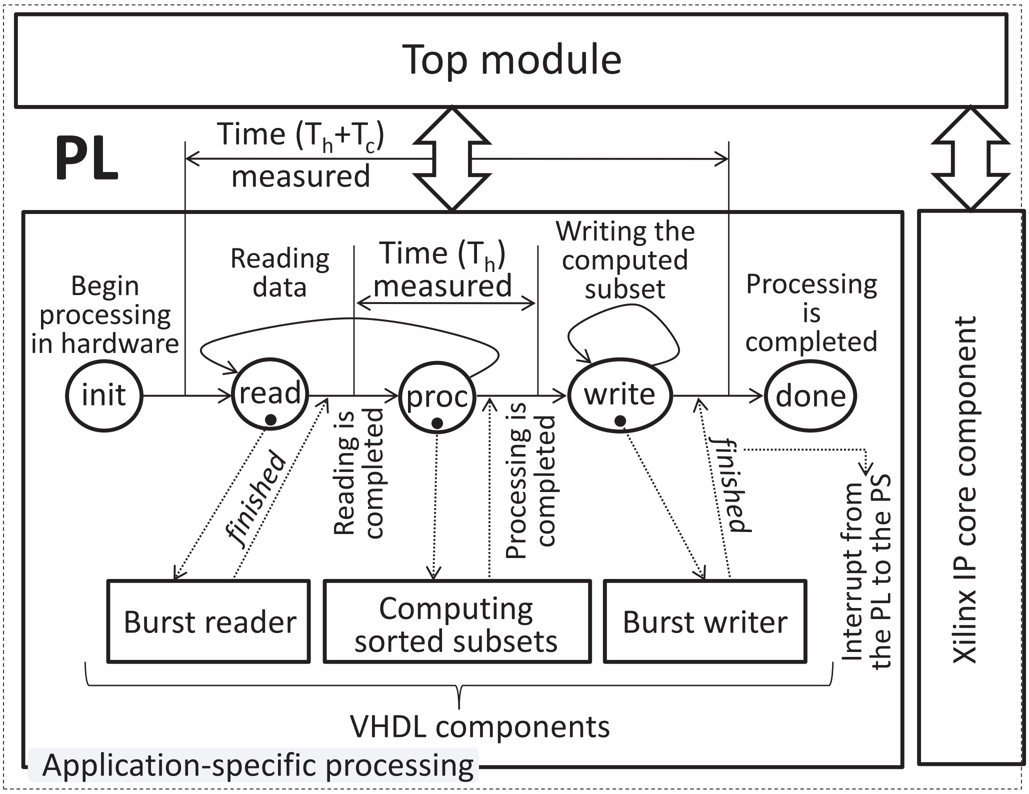


Figure 10: Operations in hardware components

6 Computing Large Subsets and Additional Capabilities

For some practical applications the maximum and/or minimum subsets may be large and the available hardware resources become insufficient to implement the circuits in Fig. 3.

The arising problem can be solved using the following technique. Let l_{max} and l_{min} be constraints for the upper and bottom parts of the sorting network in Fig. 3, i.e. circuits with larger values (than l_{max} and l_{min}) cannot be implemented due to the lack of hardware resources or for some other reasons. Let the parameters for the maximum and minimum subsets be greater than l_{max} and l_{min}, i.e. $L_{max} > l_{max}$ and $L_{min} > l_{min}$. In such case the maximum and minimum subsets can be computed incrementally [8] as follows:

1. At the first iteration the maximum subset containing l_{max} items and the minimum subset containing l_{min} items are computed. The subsets are transferred to the PS (to memories). The PS removes the minimum value from the maximum subset and the maximum value from the minimum subset. Such correction avoids loss of repeated items at subsequent steps. Indeed, the minimum value from the maximum subset (the maximum value from the minimum subset) can appear for subsets to be subsequently constructed in point 3 below and they will be lost because of filtering (see point 3).
2. The minimum value from the corrected in the PS maximum subset is assigned to B_u. The maximum value from the corrected in the PS minimum subset is assigned to B_l. The values B_u and B_l are supplied to the PL through GPP.
3. The same data items (from memory), as in point 1 above, are preliminary filtered (see Fig. 6) in such a way that only items that are less than B_u and greater than B_l are allowed to be processed, i.e. computing sorted subsets can be done only for the filtered data items. Thus, the second part of the maximum and minimum subsets will be computed and appended (in the PS) to the previously computed subsets (such as subsets from point 1). Note, that the method for processing incomplete subsets (see Fig. 4) may need to be applied for the last iteration.
4. The points 2 and 3 above are repeated until the maximum subset with L_{max} items and the minimum subset with L_{min} items are computed.

Note, that if the number of repeated items is greater than or equal to l_{max}/l_{min}, then the method above may generate infinite loops [8]. This situation can easily be recognized. Indeed, if after corrections in point 1 above any new subset becomes empty then an infinite loop will be created. In such case we can use another method based on software/hardware sorters from [9]. In section 8 we will present the results of experiments for such sorters.

For some practical cases only the maximum or the minimum subsets need to be extracted. This task can be solved easier than in Fig. 3 with the aid of the circuit shown in Fig. 11 (for computing only the maximum subset).

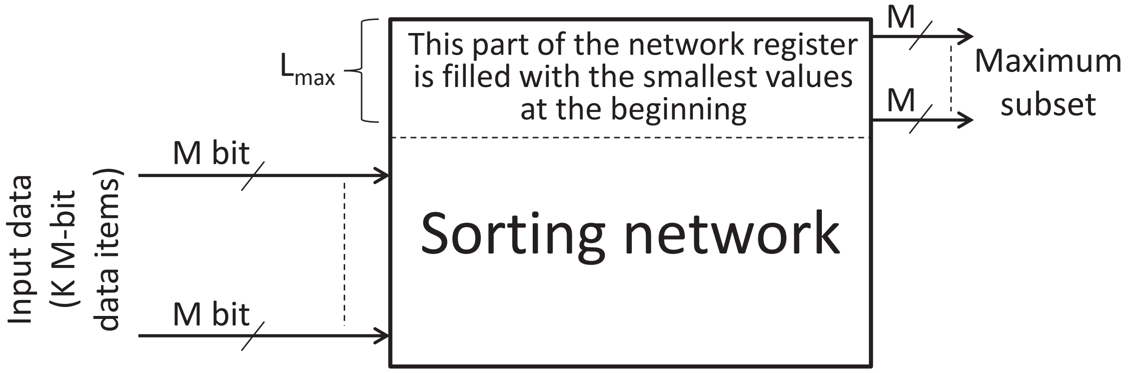

L_max
This part of the network register is filled with the smallest values at the beginning
M
M
Maximum subset
M bit
M bit
Input data (K M-bit data items)
Sorting network

Figure 11: Computing the maximum subset for a given set

At initialization stage L_{max} M-bit words of the FR (see Fig. 5) are filled in with the smallest possible value (such as zero or the minimal value for M-bit data items). After that the processing is executed as before (see section 3) and finally the maximum subset will be computed. For computing the minimum subsets the bottom part of Fig. 3 is filled in with the largest possible value (such as the maximum value for M-bit data items).

7 Practical Applications

Let us consider practical applications from the scope of control. We have already mentioned in section 1 that applying the technique [2] in real-time systems requires knowledge acquisition from controlled devices. The data may be compared with the previously collected data that are kept in databases for similar control scenarios. The results of comparison can be analyzed and used to modify the algorithms allowing control operations to be optimized, undesirable (or error prone) situations to be avoided, etc. Let us look at Fig. 12 where software collects important data from a controlled system, such as changes in temperature, deviation of positions, offsets, etc. The collected data are optionally filtered and their subsets (maximum, minimum, or both) are computed (see the bottom part of Fig. 12). Data from previous scenarios for analogous conditions are extracted from the database and they are also optionally filtered and similar subsets are computed (see the upper part of Fig. 12).

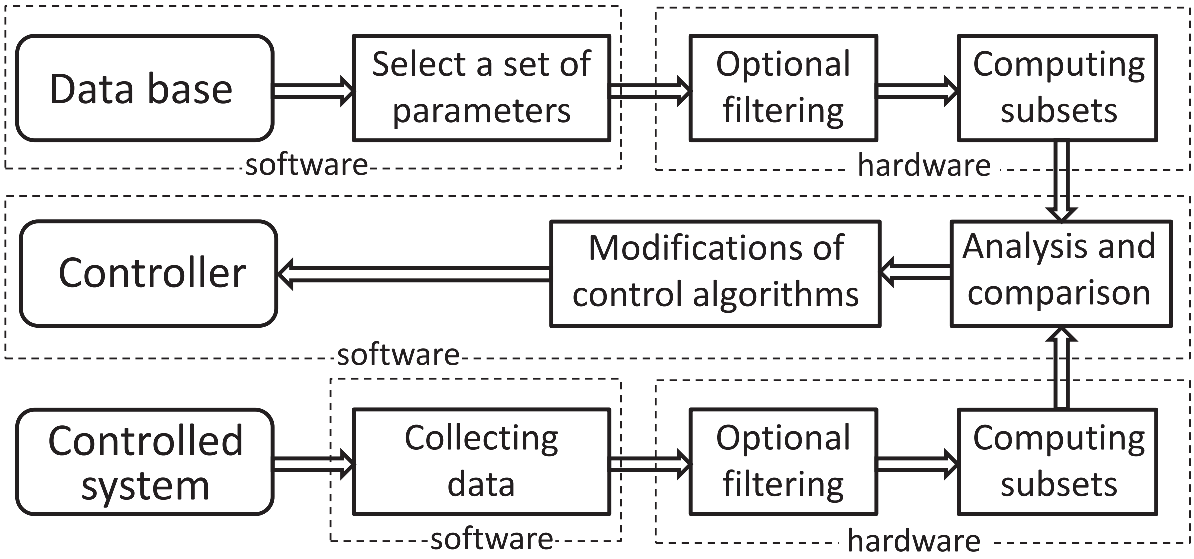


Figure 12: An example of control application

Data from the controlled system (see the bottom part of Fig. 12) and from the database (see the upper part of Fig. 12) are analyzed. For example, average maximum values are checked. The results of analysis may be used to modify control algorithms much like it is done in [9, 16]. For example, modules of controllers from [9] can be replaced to optimize execution of relevant operations.

Another group of potential applications is from the scope of statistical data manipulation such as data mining. To describe one of the problems from this area informally let us consider an example [6] with analogy to a shopping card. A basket is the set of items purchased at one time. A frequent item is an item that often occurs in a database. A frequent set of items often occur together in the same basket. A researcher can request a particular support value and find the items which occur together in a basket either a maximum or a minimum number of times within the database [6]. Similar problems appear to determine frequent inquiries at the Internet, customer transactions, credit card purchases, etc. producing very large volumes of data in the span of a day [6]. Computing sets of the most frequent or the less frequent items in large data sets permits the relevant data mining algorithms to be simplified and accelerated. Sorting of subsets is involved in many known algorithms from this area e.g. [17–19] and the results of the paper may provide a valuable assistance.

8 Implementation, Experiments, and Comparisons

Much like [8] we have used a multi-level computing system [9]. Initial data are either generated randomly in software of the PS with the aid of C language rand or prepared in the host PC. In the last case data may be generated by some functions or copied from available benchmarks. Computing subsets in software/hardware systems is mainly done in Zynq APSoC xc7z020 housed on ZedBoard [20] with the aid of the described above software/hardware architectures (see Fig. 2-4, 6, 9, 10). Computing subsets in software only sorters is completely done in software of the PS calling C language qsort function which sorts data and after that the maximum and/or the minimum subsets are extracted. The results are verified in software running either in the PS or in the host PC. Functions for verification of the results are given in [9]. Verification time is not taken into account in the measurements below.

Synthesis and implementation of hardware modules were done in Xilinx Vivado 2015.2. Standalone software applications were created in C language and uploaded to the PS memory from the Xilinx Software Development Kit (SDK) using methods described in [9]. Interactions with APSoC are done through the SDK console window.

For all the experiments 64-bit AXI ACP port was used for transferring blocks between the PL and memories. The size of each block for burst mode is chosen to be 128 of 64-bit items. Two memories were tested: the OCM (for smaller number of data items) and external (on-board) DDR. The OCM is faster because it provides for 64-bit data transfers [7] but the size of this memory is limited to 256 KB. The available on ZedBoard 4 Gb DDR supports 32-bit data transfers.

The measurements were based on time units (returned by the function XTime_GetTime [21]) for $\mathrm{L_{max}} = \mathrm{L_{min}} = 128$, M=32, and K = 256 (see Fig. 3). The following operations have been executed: a) copying data to the selected memory in the PS; b) providing the necessary initialization for the function XTime_GetTime (i.e. the consumed time will be measured from this point); c) making the request, i.e. setting (through GPP) source address, destination address, the size of data to be copied, and start processing in the PL (optionally some other data, such as $\mathrm{B_u}$ and $\mathrm{B_l}$ for filtering, may be provided); d) copying data from the PS to PL and executing all the required operations in the PL; e) copying the computed subsets from the PL to PS; f) generating a hardware interrupt that is handled in the PS as a completion of the request (thus, the consumed time is measured at this point in the PS). Each unit returned by the function XTime_GetTime corresponds to 2 clock cycles of the PS [21]. The PS clock frequency is 666 MHz. Thus, any unit corresponds to approximately 3 ns. The PL clock frequency was set to 100 MHz.

Fig. 13 shows the time consumed for computing the maximum and minimum subsets for data sets with different sizes in KB (from 2 to 128). Since M=32 the number of processed words (N) is equal to the indicated size divided by 4.

Fig. 14 shows the acceleration of the software/hardware system comparing to the software only system. Note that Fig. 13, 14 give diagrams for the OCM. If DDR memory is used then communication overheads are slightly increased but acceleration in the software/hardware system comparing to the software only system is again significant.

Let us compare the results with [5, 8]. The number of data items in the proposed solutions is larger than in [5] and can easily be additionally increased. For similar data sets the achieved acceleration is better than in [8] thanks to additional optimization of the proposed circuits.

We also implemented and tested the proposed circuits in a more advanced prototyping board ZC706 [22] with Zynq microchip xc7z045. Data were taken from DDR memory and the maximum and minimum subsets were extracted with K data items where K varied from 256 to 1,024 (as before M = 32, N is equal to 256 KB). The consumed time varies from 1,850 μs for $\mathrm{L}_{\mathrm{min}} = \mathrm{L}_{\mathrm{max}} = 256$ to 7,200 μs for $\mathrm{L}_{\mathrm{min}} = \mathrm{L}_{\mathrm{max}} = 1{,}024$. Thus, the proposed solutions can be used for solving significantly more complicated problems that cannot be solved, in particular, with the aid of the methods [5]. If only the maximum or only the minimum subsets have to be computed the acceleration is slightly increased (although it is almost the same) and the occupied hardware resources are reduced.

The proposed filtering (see Fig. 6) does not consume any additional time because it is combined with data transfers. So, we can say that the time is included in communication overheads and the latter were taken into account in all measurements. It should be noted that filtering is not described in [5, 8].

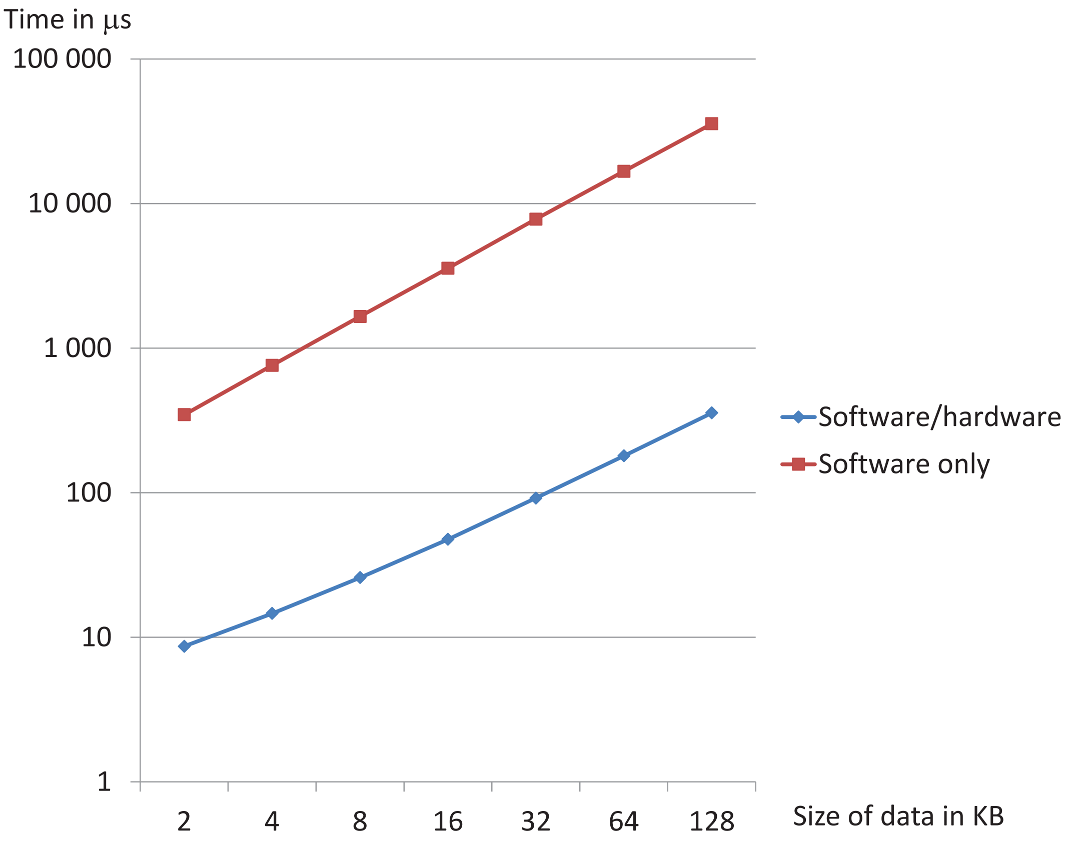


Figure 13: Computing time in software only and software/hardware systems

If the size of the requested subsets is increased in such a way that all data need to be read from memory several times then the results are the same as in [8] (see comments in [8] for additional details).

We found that parallel circuits that enable the maximum and minimum subsets to be ex-

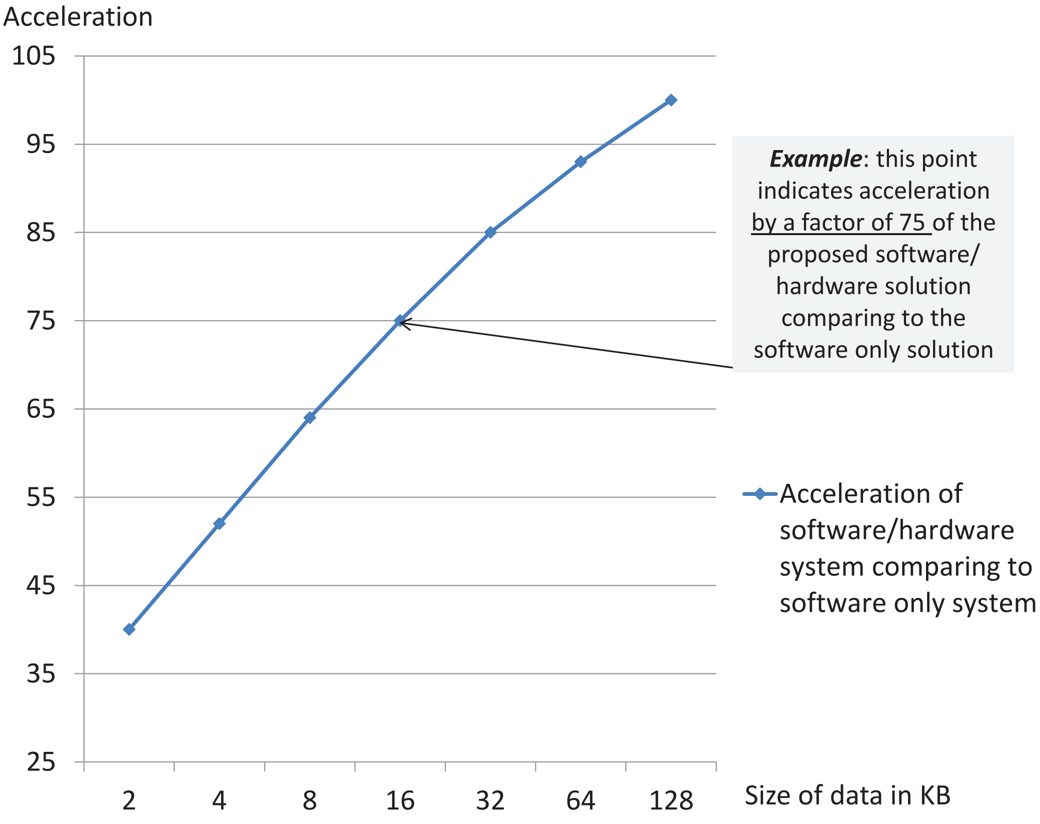


Figure 14: Acceleration of software/hardware system comparing to software only system

tracted in the ZedBoard [20] can be built up to K = 256. In this case additional hardware resources that enable data exchange between the PS (memories) and PL are available. Similar circuits for the ZC706 can be built up to K = 1,024. Note that if a block of data needs to be sorted in hardware then the number of processed data may be greater because in our case two blocks (each of which possesses K items) have to be handled in parallel and in case of data sorting [9] it is sufficient to handle just one block of data. Additional optimizations such as partial merging in hardware circuits permit the size K to be additionally increased. However, the processing time will also be increased.

9 Conclusion

The paper suggests methods for computing the maximum and minimum subsets that are extracted from large data sets in communicating software/hardware systems, namely in devices from the Xilinx Zynq family, which combine a high-performance processing system with advanced programmable logic. The extracted subsets may be filtered and this feature is useful for control applications. The proposed solutions are highly parallel permitting capabilities of programmable logic to be used very efficiently. All the proposed methods were implemented in commercial microchips, tested, evaluated, and compared with alternatives. The results of experiments have shown significant speed-up of the proposed software/hardware systems comparing to software only systems and to competitive hardware/software implementations. In particular, the size of subsets was increased and additional tasks important for control applications were discussed and solved. Practical applications of the proposed technique for control applications and statistical data manipulation were also given.

Acknowledgment

This research was supported by EU through European Regional Development Funds, the institutional research funding IUT 19-1 of the Estonian Ministry of Education and Research, ESF grant 9251, and Portuguese National Funds through FCT - Foundation for Science and Technology, in the context of the project PEst-OE/EEI/UI0127/2014.

Bibliography

[1] Sklyarov, V.; Skliarova, I. (2013); Digital Hamming Weight and Distance Analyzers for Binary Vectors and Matrices, *Int. Journal of Innovative Computing, Information and Control*, 9(12): 4825-4849.

[2] Zmaranda, D.; Silaghi, H.; Gabor, G.; Vancea, C. (2012); Issues on applying knowledge-based techniques in real-time control systems, *International Journal of Computers Communications & Control*, 8(1): 166-175.

[3] Field, L.; Barnie, T.; Blundy, J.; Brooker, R. A.; Keir, D.; Lewi, E.; Saunders, K. (2012); Integrated field, satellite and petrological observations of the November 2010 eruption of Erta Ale, *Bulletin of Volcanology*, 74(10): 2251-2271.

[4] Zhang, W.; Thurow, K.; Stoll, R. (2014); A knowledge-based telemonitoring platform for application in remote healthcare, *International Journal of Computers Communications & Control*, 9(5): 644-654.

[5] Farmahini-Farahani, A.; Duwe, H. J.; Schulte, M. J.; Compton, K. (2013); Modular design of high-throughput, low-latency sorting units, *Computers, IEEE Transactions on*, 62(7): 1389-1402.

[6] Baker, Z. K.; Prasanna, V. K. (2006); An architecture for efficient hardware data mining using reconfigurable computing systems. In Field-Programmable Custom Computing Machines, *Proc. 14th Annual IEEE Symp. on Field-Programmable Custom Computing Machines - FCCM*, Napa, USA, 67-75.

[7] Xilinx, Inc.; Zynq-7000 All Programmable SoC Technical Reference Manual, available at http://www.xilinx.com/support/documentation/user_guides/ug585-Zynq-7000-TRM.pdf.

[8] Sklyarov, V.; Skliarova, I.; Rjabov, A.; Sudnitson, A. (2015); Zynq-based System for Extracting Sorted Subsets from Large Data Sets, *Journal of Microelectronics, Electronic Components and Materials*, 45(2): 142-152.

[9] Sklyarov, V.; Skliarova, I.; Silva, J.; Rjabov, A.; Sudnitson, A.; Cardoso, C. (2014); *Hardware/software co-design for programmable systems-on-chip*, TUT Press.

[10] Mueller, R.; Teubner, J.; Alonso, G.; (2012); Sorting networks on FPGAs, *The VLDB Journal—The International Journal on Very Large Data Bases*, 21(1), 1-23.

[11] Sklyarov, V.; Skliarova, I. (2014); High-performance implementation of regular and easily scalable sorting networks on an FPGA, *Microprocessors and Microsystems*, 38(5): 470-484.

[12] Baddar, S. W. A. H.; Batcher, K. E. (2012); *Designing sorting networks: A new paradigm*, Springer Science & Business Media.

[13] Knuth, D. E. (2011); *The art of computer programming: sorting and searching (Vol. 3)*, Addison-Wesley.

[14] Kipfer, P.; & Westermann, R. (2005); Improved GPU sorting, *GPU gems 2: programming techniques for high-performance graphics and general-purpose computation*, edited by M. Pharr, 733-746, available at http://http.developer.nvidia.com/GPUGems2/gpugems2_chapter46.html.

[15] Silva, J.; Sklyarov, V.; Skliarova, I. (2015). Comparison of On-chip Communications in Zynq-7000 All Programmable Systems-on-Chip, *Embedded Systems Letters, IEEE*, 7(1): 31-34.

[16] Sklyarov, V.; Skliarova, I.; Barkalov, A.; Titarenko, L. (2014); *Synthesis and Optimization of FPGA-based Systems*, Springer.

[17] Sun, S. (2011); *Analysis and acceleration of data mining algorithms on high performance reconfigurable computing platforms*, Ph.D. thesis, Iowa State University, available at: http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=1421&context=etd.

[18] Wu, X.; Kumar, V.; Quinlan, J. R. et al. (2008); Top 10 algorithms in data mining, *Knowledge and Information Systems*, 14(1): 1-37.

[19] Firdhous, M. (2012); Automating legal research through data mining, *Journal of Advanced Computer Science and Applications*, 1(6): 1-8.

[20] Avnet, Inc. (2014); *ZedBoard (ZynqTM Evaluation and Development) Hardware User's Guide, Version 2.2*, available at: http://www.zedboard.org/sites/default/files/documentations/ZedBoard_HW_UG_v2_2.pdf.

[21] Xilinx, Inc.; *OS and Libraries Document Collection UG647*, available at: http://www.xilinx.com/support/documentation/sw_manuals/xilinx2014_2/oslib_rm.pdf.

[22] Xilinx, Inc.; *ZC706, All Programmable SoC Evaluation Kit, UG961*, Available at: http://www.xilinx.com/support/documentation/boards_and_kits/zc706/2014_3/ug961-zc706-GSG.pdf.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(1):142-156, February 2016.

A Context-Aware mHealth System for Online Physiological Monitoring in Remote Healthcare

W. Zhang, K. Thurow, R. Stoll

Weiping Zhang*, **Kerstin Thurow**
University of Rostock
Celisca, Center for Life Science Automation,
Rostock 18119, Germany
weiping.zhang@uni-rostock.de,kerstn.thurow@uni-rostock.de
*Corresponding author: weiping.zhang@uni-rostock.de

Regina Stoll
University of Rostock
Institute for Preventive Medicine,
University Medical Center,Rostock 18055, Germany
regina.stoll@uni-rostock.de

Abstract: Physiological or biological stress is an organism's response to a stressor such as an environmental condition or a stimulus. The identification of physiological stress while performing the activities of daily living is an important field of health research in preventive medicine. Activities initiate a dynamic physiological response that can be used as an indicator of the overall health status. This is especially relevant to high risk groups; the assessment of the physical state of patients with cardiovascular diseases in daily activities is still very difficult. This paper presents a context-aware telemonitoring platform, IPM-mHealth, that receives vital parameters from multiple sensors for online, real-time analysis. IPM-mHealth provides the technical basis for effectively evaluating patients' physiological conditions, whether inpatient or at home, through the relevance between physical function and daily activities. The two core modules in the platform include: 1) online activity recognition algorithms based on 3-axis acceleration sensors and 2) a knowledge-based, conditional-reasoning decision module which uses context information to improve the accuracy of determining the occurrence of a potentially dangerous abnormal heart rate. Finally, we present relevant experiments to collect cardiac information and upper-body acceleration data from the human subjects. The test results show that this platform has enormous potential for use in long-term health observation, and can help us define an optimal patient activity profile through the automatic activity analysis.

Keywords: decision support systems, telemonitoring, Context-Aware application

1 Introduction

For patients with cardiovascular disease, physiological stress is an important index to measure their overall health status [1]. However, the daily physiological stress ration of a patient cannot be carried out through simple laboratory simulation due to the various factors that influence daily life; this makes the standardization work for measuring physiological stress difficult. Currently a large amount of research work in the field of preventive medicine is developed around physiological monitoring. With the development of communication and sensor technologies, the mode of (a) real-time physiological parameter acquisition and (b) automatic online analysis provides new technical solutions for effectively resolving long-term monitoring of a patient's daily activity. In previous work [2] [23] we reported about the potential relationship between cardiac information and physiological stress. A novel fuzzy modeling based HRV analysis method for stress assessment was proposed in [23]. The method of [23] extracts the features of HRV in time-frequency domain and fuzzy techniques are exploited to render robustness in HRV analysis against uncertainties arising from individual variations. This paper presents a wearable, remote-monitoring system based on patient context information. The monitor's purpose is to evaluate the physiological parameter combined with the human body status, and detect an anomalous event through the context information.

Patient context-aware information is an important concept in the mobile health environment. According to the descriptions of Dey, Abowd, et al., in their thesis's general definition of context [3], patient context can be defined as the information for a patient's medical situation, which can be roughly divided into the following categories: (a) patient's vital parameters, (b) medical symptoms (vomiting ...), (c) risk factors (cholesterol level ...), (d) activities (standing, walking ...), and (e) surrounding environment (room temperature ...)

Context-aware computing is an important research field in ubiquitous computing [4]. "Context-aware system" refers to a system which can perceive a change in the user's environment and make a corresponding adjustment. Under a mobile health environment, the context-aware system connects various medical and perception equipment via the network; access to various resources and information is more convenient for medical staff, thus improving their efficiency. The fusion of environment/position information, patient's health status, and physiological parameters can provide rich context information for doctors' decisions and provide an enhanced environment for more-informed medical service.

2 Related Work

In related work various types of relevant data have been collected individually or in some combination for a variety of healthcare purposes [5]- [6]. Physiological profile data has been used exclusively [7] or has been used in conjunction with activity profile data [8] and environmental information [9]. Among these papers, special note should be taken of the system described in [10]. This work presented an approach where vital parameter changes of patients are detected in the biosignal recording system in real-time, however, the users are prompted to provide additional input about their daily living activities. Therefore, user intervention is required in that system to provide additional context information [11]. In [12] systems are presented that analyze physiological function in conjunction with automated observation, wellbeing, and health status. In [13] a system is described that analyzes the relationship between an activity profile and physiological information.

Automatic recognition and quantification of human activities using wearable sensors during activities of daily living has been increasingly used in many research works [24]- [27].

3 System structure

This paper proposes a multi-sensor system which uses an inference decision support system with a basis of rules to improve the accuracy of discovering potentially dangerous heart rate variability. As shown in Figure 1, two different sensors at local site of patient are used for acquiring patient context information and physiological parameters. The sensors in the first category include environmental sensors used for monitoring interaction between a patient and the surrounding environment; meanwhile, a wearable acceleration sensor acquires the patient's activity status. The information acquired in this group of sensors can accurately reflect the time, location, and context of the patient's activity performing in the environment. The sensors in

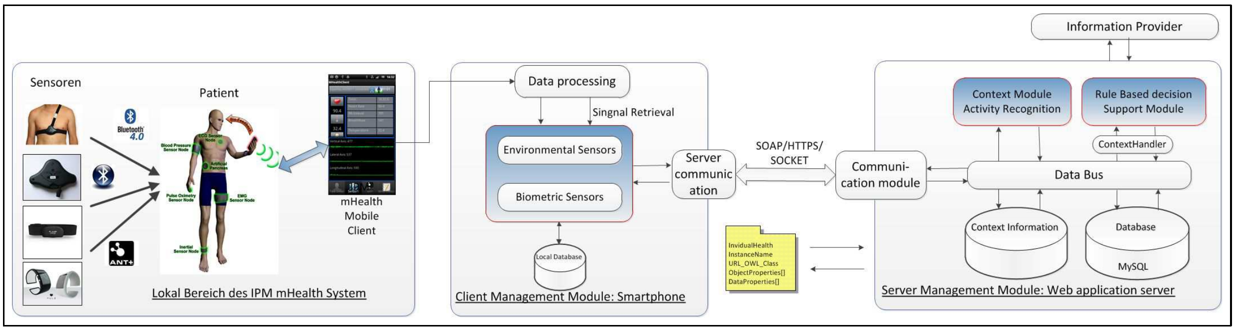


Figure 1: Institute for Preventive Medicine mobile Health (IPM-mHealth) System Structure

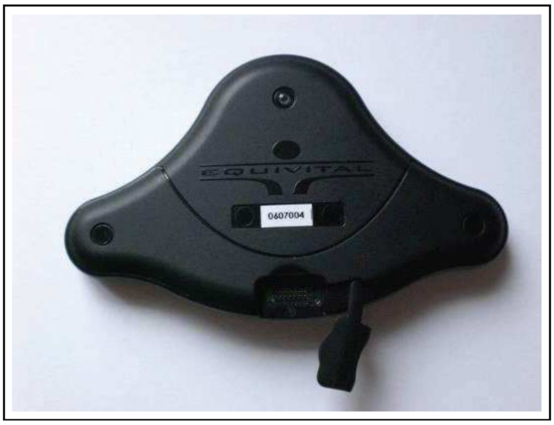


Figure 2: Hidalgo Equivital Muti parameter Sensor

the second category include vital signs sensors, which is a typical wearable sensor for measuring the patient's physiological parameters. Real-time accurate physiological parameter acquisition is the basis for data analysis; then we can combine the context information of physical activity to carry out more effective physiological status analysis. This paper has the focus on the two core elements of the framework: (1) automated activity recognition, and (2) knowledge-based decision module.

- Activity recognition can mainly be divided in two ways: visual-based and sensor-based [14] - [15]. The activity recognition method based on a 3-axis acceleration sensor belongs to the latter; this is the newly emerged branch of human body activity identification research. Compared with traditional activity recognition based on visual sense, it provides advantages such as the acquisition of movement data in a simpler and more human manner, etc. The algorithm designed in this paper focuses on the acceleration signal worn on a patient's body as a recognition basis.

- The Decision Module represents the intelligent core part of the system. For context processing, modeling should be carried out first for a patient's context information. (For example: the activity and position in an abnormal heartbeat situation can be included into one context). The experiments uses Web Ontology Language (OWL) to construct the model and process it using Sample Semantic Web Rule Language [16], which can also provide the higher level information about the overall health status of patients through its conditions based engine.

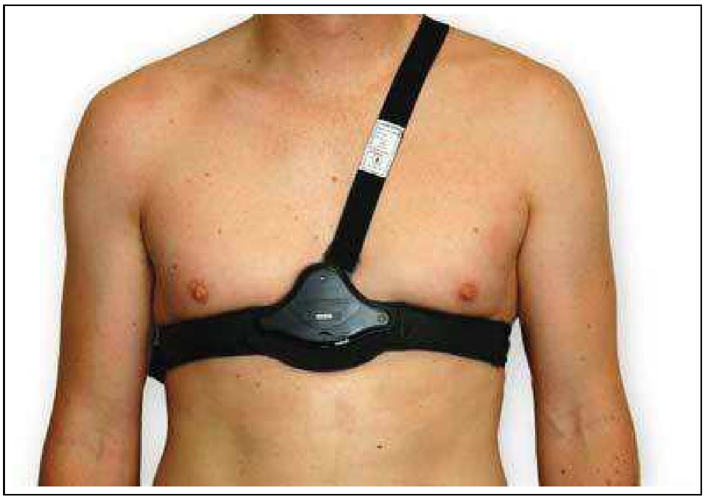


Figure 3: Equivital sensor chest belt

The system is tested by using the hardware devices: Equivital wearable-mutiparameter-sensor solution developed by Hidalgo [22]. One Equivital device was placed on the upper body of the participant (as shown in Figure 2 and 3). The sensor module with a special chest belt is an appropriate solution for recording the required measurands. It offers the possibility to acquire cardiac function (via three integrated fabric-based silver-coated electrodes), pulmonary function (via an integrated resistance strain gauge),activity function activity(via an integrated accelerometer ADXL330) and skin temperature (via an integrated thermistor).

The Equivital sensor solution, fully charged, offers 24 hours of operation. Due to the low overall weight of 175g (electronic module and the sensor 75g; belt 100g) and the dust/water resistance protection rating of IP 67 (0.4 m = 30 min.), the IPM-mHealth system allows subjects to move freely in their familiar work surroundings as an examiner continuously monitors their condition from any location.

4 System Design

4.1 Automatic Activity Recognition

Automated activity analysis is the essential component of the IPM-mHealth system which provides new opportunities for quality long-term studies of various vital functions. Activity identification can be defined as a classification problem in the machine learning field. The basic flow of activity identification is as follows. First, 3-axisacceleration sensor data should be acquired and sensor features (mean value, peak value, standard deviation, spectrum energy, etc.) should be extracted within the specific time window. Then, classification training for sensor data should be carried out under a different activity status. The activity recognition flow is shown in Figure 4:

1. First the accelerometer data will be collected and smoothed.

2. Specific sensor values will be extracted within a defined time window (average, maximum, standard deviation, etc.).

3. Then the WEKA tool will be used to classify the measured data relevant to the various states.

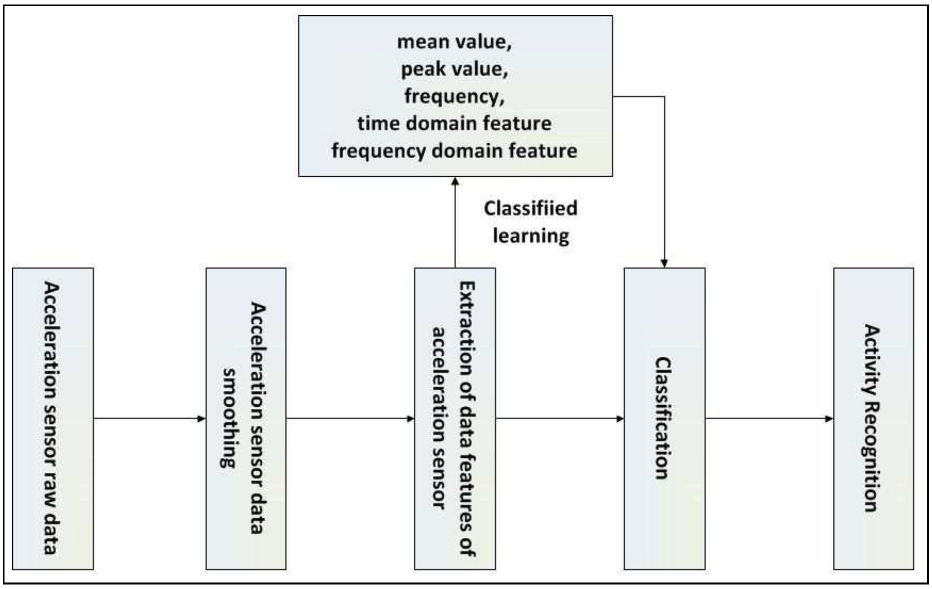


Figure 4: Flowchart for automatic activity recognition

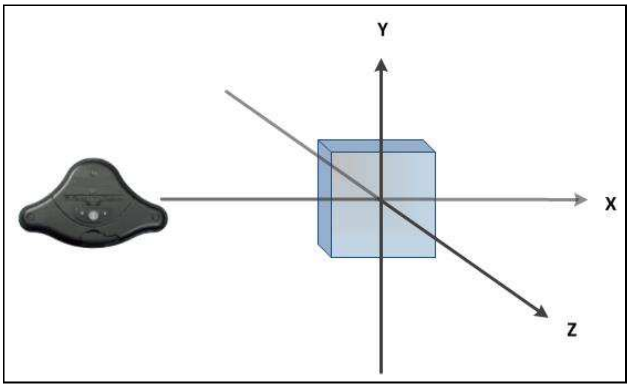


Figure 5: Direction of the Equivital 3-Axis Accelerometer

4. Finally, we use the 1-k-Nearest Neighbour (kNN) classifier to classify the activity status.

(1) Acceleration data acquisition

In this paper, we capture data for ten (10) volunteers performing six (6) different movements, including standing, walking, running, jumping, walking upstairs, and walking downstairs; for each action, the sensor data was collected five (5) times. A total of 300 samples of acceleration data are used as the original acceleration signal. The signals are collected by the Equivital sensor described above. The sensor module itself has an integrated three-axis accelerometer, ADXL330, and allows detection of orientation and body activity. The device's integrated 3-axis accelerometer was used at a sample rate of 100 Hz for each axis, providing synchronous acquisition of motion data.

The coordinate-system of the 3-axis acceleration sensors is defined relative to the front of the devicein its default orientation. As Figure 5 shows, the X-axis is horizontal and points to the right, the Y axis is vertical and points up and the Z axis points towards the outside of the front face of the device. In this system, coordinates behind the device have negative Z values.

All of these data (raw data, calculated data, and message data) can be transferred continuously and wireless from the sensor module, via a Bluetooth class interface, to the IPM-mHealth

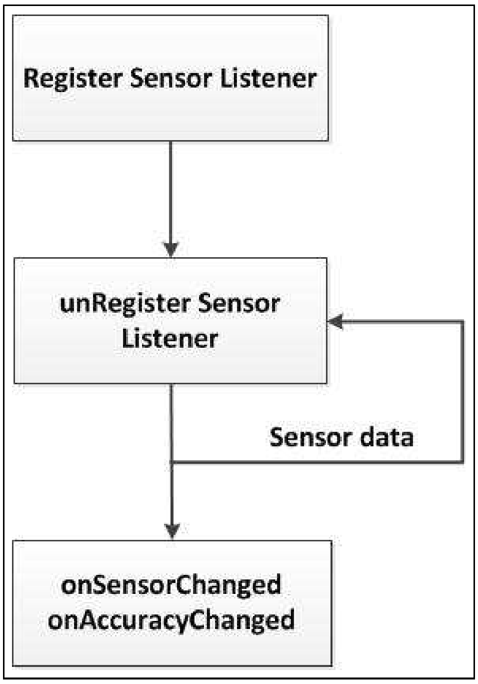


Figure 6: Data transmission process

client in realtime. By calling the Sensor Communication Module offered in the IPM-mHealth client, the raw data of Equivital acceleration sensors can be transferred and stored.

As shown in Figure 1, the IPM-mHealth client, running on the smart phones, is responsible for the collection of sensor node data and for sending the data to a remote data processing center. The data processing center, based on a Hadoop server cluster, is responsible for data collection, analysis, processing, and data mining. The communication between the IPM-mHealth client and server is implemented through HTTP/HTTPS, SOAP, and SOCKET. HTTP is used in synchronous XML workflow description; SOAP is used to transmit ontology instance information; the SOCKET connection is used in real-time high-demand, for example, transmitting ECG (1024 bytes/sec) and Acceleration (300 bytes/sec) data.

The acceleration data transmission process is shown in Figure 6.The Accelerometer EventListener Class is established to acquire the data; this class inherits the class Sensor EventListener of the Android API. It provides the following features: Registration of sensor services, Recall of sensor services, Acquisition of the sensor data, and Listening for real changes in the data.

After the sensor data is detected, the sensor data of the three directions can be obtained by calling the methods: onSensorChanged-DataX, onSensorChanged-DataY and onSensorChanged-DataZ. Since the direction has little influence to the following motion identification algorithm. Here we carried out a handling, i.e.: the vector sum of the acceleration data in each change should be calculated through the following equation:

$$\mathbf{D_i}=\sqrt{(\mathbf{A}cc\_\mathbf{X})^{\mathbf{2}}+(\mathbf{A}cc\_\mathbf{Y})^{\mathbf{2}}+(\mathbf{A}cc\_\mathbf{Z})^{\mathbf{2}}}$$

The data after each change of acceleration sensor status and the currently relative time (unit: millisecond) are respectively saved in two arrays, time-series[] and data-series[]. The original data time sequence after acquisition is shown in the following Figure, the horizontal axis is the time axis, i.e., time-series[], and the vertical axis is acceleration sensor data Di, i.e., data-series[].The motion order in the below figure is: (1) walk, then (2) a series of three times of jumps, (3) run,

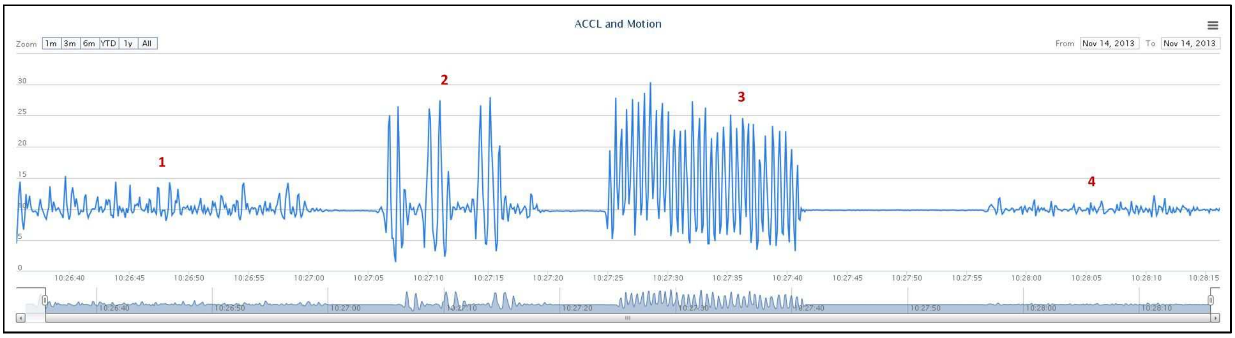


Figure 7: Time sequence diagram of acceleration raw data

and (4) slow walk. It can be seen from the original data that different motions differ in the data features of mean value, peak value, frequency, etc.

(2) Acceleration sensor data smoothing

The data acquisition interface of the Equivital sensor adopts a passive invoking method, where the data will be sent out only when the acceleration status changes. Therefore, the data acquired in the previous step are not evenly distributed at each time point. In order to equalize the time interval between each data point we used the following smoothing algorithm to calculate the mean data value at the time point with the an interval of 40 milliseconds based on the acquired original data time-series[] and data-series[].

The latest two time points in the time-series[] which is nearest to the time point T0 to be calculated are T1 and T2. Wherein: T1 <T0, T2 >T0 and the corresponding data on data-series[] are obtained to be D1 and D2. Assume that the acceleration speed of a mobile phone between T1 and T2 is linear, then the data value at T0 is D0, which can be calculated through the following weighted average method:

$$\mathbf{D_0}=\frac{\mathbf{D_1}*(\mathbf{T_2}-\mathbf{T_0})+\mathbf{D_2}*(\mathbf{T_0}-\mathbf{T_1})}{\mathbf{T_2}-\mathbf{T_1}}$$

New time-series[] and data-series[] can be calculated and set up through the above algorithm.

(3) Extraction of data features of acceleration sensor

In order to accurately capture and identify motions, the data set window can be set to 512*40, i.e., 20,480 milliseconds. Therefore, the data-series[] array can save 512 samples. When the array is full, re-save should be carried out from data-series[0] after emptying it. After filling in the array, data extraction with a time domain feature and frequency domain feature should be carried out for another time.

(a) Time domain feature

The time domain feature is the feature value of the data within a certain time window. Due to the smoothing of data carried out in the previous step, the acceleration sensor data is with the interval of 40 milliseconds Therefore, the time domain feature of the acceleration sensor, within 20,480 milliseconds (about 20 second), can be calculated directly via the mean value, maximum value, and minimum value of all the data within the data-series[] array. By extracting the various features of the acceleration signal within a single time window, several feature vectors can be constructed to characterize the behavior. In this paper, the following feature vectors are used.

1) Standard Deviation Formula: Standard deviation is defined as

$$\sigma=\sqrt{\frac{\mathbf{1}}{\mathbf{N}}\sum_{\mathbf{i=1}}^{\mathbf{N}}(\mathbf{X_i}-\overline{\mathbf{X}})^{\mathbf{2}}}$$

where N is the number of samples and X is the sample mean. Standard deviation, an often-used statistical characteristic. Standard deviation reflects the degree of dispersion of the acceleration sensor data. Since the acceleration data are unchanged when subjects are in the static state, standard deviation is near zero. When subjects are moving, the acceleration data are constantly changing and the standard deviation is always much greater than zero. Therefore, the standard deviation is an important feature to identify static operation and dynamic action.

2) Skewness: skewness is defined as formula

$$\mathbf{S}K{=}\frac{\mathbf{N}\sum_{\mathbf{i=1}}^{\mathbf{N}}\left(\mathbf{X_i}-\overline{\mathbf{X}}\right)^{\mathbf{3}}}{(\mathbf{N}-\mathbf{1})\,(\mathbf{N}-\mathbf{2})\sigma^{\mathbf{3}}}$$

where N is the number of samples, X is the sample mean (data), and σ is the sample standard deviation.In probability theory and statistics, skewness is a measure of the asymmetry of the probability distribution of the acceleration sensor data about its mean. The Skewness of the X axis can effectively distinguish between actions such as walking downstairs.

3) Kurtosis: kurtosis is defined as formula

$$\mathbf{K}{=}\frac{\sum_{\mathbf{i=1}}^{\mathbf{N}}\left(\mathbf{X_i}-\overline{\mathbf{X}}\right)^{\mathbf{4}}\mathbf{f_i}}{\mathbf{N}\sigma^{\mathbf{4}}}$$

where N is the number of samples, X is the sample mean (data) , σ is the sample standard deviation, and fiis the Sample Interval. Kurtosis is any measure of the "peakedness" of the probability distribution of acceleration sensor data, an important statistical characteristic. Kurtosis of the Y axis can effectively distinguish running and other actions.

(b) Frequency domain feature

The time domain feature alone cannot reflect the obvious characteristics in frequency domain of motion, therefore, characteristics in frequency domain should be obtained via Fast Fourier Transformation. Here, Fast Fourier Transformation should first be carried out on the data-series[] array. Fast Fourier Transformation (FFT) is a kind of fast algorithm of Discrete Fourier Transformation (DFT), which reduces the number of computations needed for N points from 2N2 to 2NlgN and the data can be efficiently changed from time domain into frequency domain. Fast Fourier Transformation should be carried out on data-series[] (FFT operation can be carried out directly if data-series[] array is 512and results in a complex number array in frequency domain space. Each complex number in the array includes real and image data. The corresponding Fourier component of each frequency within the frequency domain can be obtained through the following relationship, i.e., modulus of the complex number.

The energy of characteristics quality in frequency domain, which is required in this paper, is the ratio between the quadratic sum of Fourier components within a window and the window size. I.e., frequency domain energy reflected the data periodicity, as for the motion with high periodicity (such as running), the frequency domain energy is obviously high.

$$\mathbf{E}nergy{=}\int_{\mathbf{k=0}}^{\mathbf{w}}\frac{(\mathbf{M}od[\mathbf{k}])^{\mathbf{2}}}{\mathbf{w}}$$

In order to be convenient for feature extraction, this paper uses a windowing method to split the original acceleration signals. The single acceleration signal, after being windowed, includes 512 samples, which is enough for including a single complete motion of patient. If a shorter, rectangular window is adopted, it is not large enough to include information for identifying different motions. If the length of the rectangular window is too long, serious delay phenomenon will occur to the real-time system.

(4) Classified-learning

After above procedures obtain the feature values, data excavation tools should be used in the following step. This paper uses WEKA [18] to classify the learned test data, so as to lay a foundation for the following Activity.

(5) Classification

In the experiment, we used 45Training samples and 6-11 training samples of each action. Volunteers completed the following actions sequentially:(1) walking, (2) jumping, (3) standing, (4) walking up stairs, (5) walking down stairs, and (6) running. Every action lasted 30 to 50 seconds to make sure that, for every action, three (3) to five (5) groups of eigenvalues could be collected.

Finally, we use the 1-k-Nearest Neighbour (kNN) classifier to classify the obtained feature values based on the learning basis in the previous step. Its confusion matrix test results are shown as Table 1. We can see from the confusion matrix that the classification accuracy of 1-k-Nearest Neighbour (kNN) is high, and the overall accuracy is 95.56%, of which an example could be one jump motion wrongly included into running, and an example of one standing motion included into walking, which shows that these two groups of motions may produce slight confusion to sorting algorithms.

Table 1. Confusion matrix of knn results

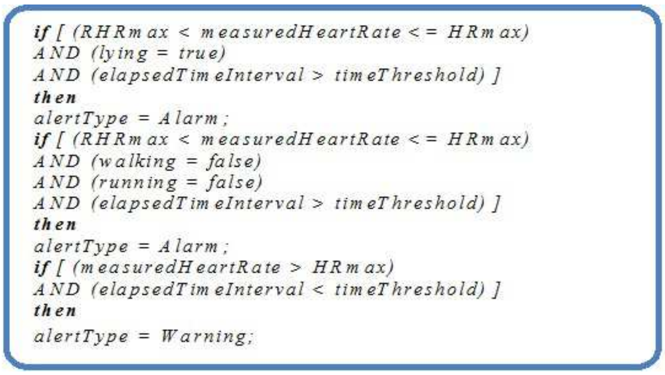
Activity	**Classification**					
	Jump	**Walking**	**Standing**	**Down-stairs**	**Up-stairs**	**Running**
Jump	7	0	0	0	0	1
Walking	0	11	0	0	0	0
Standing	0	1	11	0	0	0
Down stairs	0	0	0	6	0	0
Upstairs	0	0	0	0	4	0
Running	0	0	0	0	0	4

4.2 Knowledge Representation

In order to detect an abnormal situation, the heart rate variability is particularly the most relevant parameter identified by cardiologists [20]. In this experiment, we used Equivital multi-parameter sensors to collect the ECG data. The changes in the beat-to-beat heart rate, the heart rate variability (HRV),calculated from the electrocardiogram (ECG), is a key indicator of an individual's cardiovascular condition. Assessment of HRV has been shown to aid clinical diagnosis and intervention strategies. An overview of some implemented production rules in the system is shown in the following Figure.

During the process of acquiring physiological parameters and activity identification an approach based on Context Information is used in the Physiological Decision Module to detect any abnormal cardiovascular events. It includes the basic medical knowledge required to identify potentially dangerous situations of patients in cooperation with the cardiologists and clinical experts. We first need to construct the Context Model before processing the Context Information [19]. The following forms are included in the context models: (Subject, Predicate, Value).

If the heart rate value is outside the scope regarded as "normal", or the heartbeat is not regular, we can simply identify HRV. Moreover, it can also determine HRV by means of the SDANN, the standard deviation of the average R-R intervals calculated over short periods, usually 5 minutes. However, according to the involved cardiologists, the HRV by itself may not have any meaning for identifying an anomalous situation; instead, it should be correlated with other information pertaining the patient, e.g. the physical activity and his/her posture. The following



```
if [ (RHRmax < measuredHeartRate <= HRmax)
AND (lying = true)
AND (elapsedTimeInterval > timeThreshold) ]
then
alertType = Alarm;
if [ (RHRmax < measuredHeartRate <= HRmax)
AND (walking = false)
AND (running = false)
AND (elapsedTimeInterval > timeThreshold) ]
then
alertType = Alarm;
if [ (measuredHeartRate > HRmax)
AND (elapsedTimeInterval < timeThreshold) ]
then
alertType = Warning;
```

physiological parameters are used by the implemented production rules on this aspect:

Cardiac information

- HRmax: patient's maximum heart rate calculated by Karvonen formula [21];
- measuredHeartRate: patient's heart rate measurement;
- restingHeartRate: patient's resting heart rate;
- timeThreshold: time threshold usually 10 sec;
- RHRmax : RHR+10;
- RHRmin : RHR-10;

Posture Information

- lying: datatype [boolean] property detecting whether the patient is lying or not;
- standingUp: datatype [boolean] property detecting whether the patient is standing up or not;

Physical Activity Information

- running activity: datatype [boolean] property detect whether the patient is running or not;
- walking activity: datatype [boolean] property detect whether the patient is walking or not;

Environmental Information

- room temperature: temperature of room where patient lives;

For example, we have a patient with HeartRate of 70 and ID 001, and physical activity was regarded as running, then the context information would be recorded as follows:

- (Patient001, HeartRate, 70), (Activity, Running, true), We can also define the specific conditions for the emergency task to be triggered: e.g. the Heart Rate of the patient drops below 40, or the blood pressure (systolic) exceeds 170. Reasoning may be carried out according to the rules of the language itself, for example:

- $(?a?p?b),(?prdf : subPropertyOf?q)- > (?a?q?b)$

User defined rules may also be used in the reasoning process, e.g.:

- (?patient, EquitalBloodPressure001, vl), GE(?vl, 170), (?patient, EquitalBloodPressure, vl), LE(?vl, 40) ->(?patient, healthStatus, "danger")
- (?patient, healthStatus, "danger"), (?Patient, healthStatus, "movementfail"), (EmergencyTask, taskState, "CLOSE") ->(EmergencyTask, taskState, "OPEN")

Some example rules are shown in Table 2, which are used by the system to predict an abnormal situation. When the heart rate is abnormally high, the HeartRateHigh rule assigns the alarm level. Abnormal heart rate in each instance of the HeartRate Class is specified in the hasMinRange and hasMaxRange properties. The system sets the Heartrate alarm if a patient's heart rate is greater than the specified normal maximum.

Table 2. Rules for the system's alarm management

Rule	**Description**
HeartRate-High	(?patient rdf:type HeartRate) , (?par1 hasCurrentValue ?v1) , (?par1 hasMaxRange?Max), greaterThan(?v1, ?Max) ->(?taskstate hasAlarmLevel "HES")
HeartRate-Low	(?patient rdf:type HeartRate) , (?par1 hasCurrentValue ?v1) , (?par1 hasMinRange ?Min), greaterThan(?v1, ?Min) ->(?taskstate hasAlarmLevel "LES")

5 Experimental results

In order to identify whether various functions of the IPM-mHealth system are effective or reliable in practical application, we designed an initial experiment. Its purpose was to carry out a test and verify the following functions: quality of physiological signals acquired by IPM-mHealth; comfort of sensor system; online activity recognition and alarm reliability; reliability of short-distance wireless communication; and reliability of remote wireless communication. Five healthy volunteers in the 21 35 age range without cardiovascular disease history participated in this test.

In the first test subjects with Equivital sensors carried out the motions for walking, jumping, running, standing, climbing the stairs, and walking down the stairs, in proper order; another observer recorded the test subjects' motions and corresponding times. In the second test, we simulated the abnormal heartbeat event and tested the reliability of the decision-making system under lying-flat and running conditions with the test subjects.

The relevant test data and statistical results are shown in Table 3 and Table 4. In the stability tests, we used ten (10) minutes to teach each volunteer how to put on and operate the system, then let them go home with the wearable device and keep the system running more than 24 hours. The sensor data was collected in real-time and sent to the central IPM-mHealth server; Table 4 lists the transmission packet loss rate for 24 hours' data. We can see that both, the short-distance wireless transmission and remote wireless transmission, operated normally. The expected real-time physiological data transmission was not lost. Illegal disconnection conditions did not occur in short-distance wireless transmissions during the entire test process. The acquired signal quality was satisfactory. In the automatic activity recognition experiment, volunteers completed totally 300 actions and 10 times for each action. In the abnormal alarm part, good effect was also received in the simulated anomaly detection, of which the conditions of over 150 heartbeats under 10 times of lying flat were correctly identified with alarms, while over 150 heartbeats under running state were included into the normal range. In the activity identification part, 90% motions were correctly identified by the system. The activity identification of standing and jumping had the best accuracy; the activity of climbing and walking down the stairs also had the better accuracy, while running, walking, etc. has misjudgement within 8%.

Table 3. Results of the automatic activity recognition and abnormal situation indicating

Activity	**Correct recognition rates**	**HR max**	**Alert (HR max)**	**Std**
Standing	97.8%	170	Alarm	10
Walking	91.6%	122	None	10
Running	92.6%	178	None	10
Jump	97.6%	150	None	10
climbing the stairs	96.6%	140	Warning	10
Walking down the stairs	95.2%	110	None	10

Table 4. Results of stability test

Parameter	**Acquisition frequency of sensor data**	**Packet-loss-rate**	**Data transmitted in a 24-hour period**	**Std**
Heartrate	0.3Hz	<0.1%	25,920 bytes	24
Respiratory	0.3Hz	<0.1%	25,910 bytes	24
Patient Temperature	0.3Hz	<0.1%	25,890 bytes	24
R-R Interval	1Hz	<0.1%	86,390 bytes	24
Acceleration X	100Hz	<0.1%	2,164,986bytes	24
Acceleration Y	100Hz	<0.1%	2,164,980bytes	24
Acceleration Z	100Hz	<0.1%	2,164,900bytes	24

We also find in the test results that the motions identified within the activity change duration may be misjudged when test subjects conducted multiple motions successively. The causes for this misjudging lie in the selection of the time window when the sensor test data was acquired. The time window selected in this paper is about 20 seconds, the data within 1 second may mix the corresponding acceleration data of two activities. Therefore, the data features and user activity did not mutually correspond, it could lead to misidentification between two activities. This problem can be solved by selecting a shorter time window, but the selection of an overly-short time window may create non-obvious problems for the corresponding data features of various activities.

6 Conclusion

A framework for context-aware physiological analysis with regard to daily activities is proposed in this paper for the detection of abnormal cardiac situations. OWL is used in the context-reasoning module to construct target medical tasks and circumstances. The IPM-mHealth system has been used in the Institute for Preventive Medicine at Rostock University Medical Center

Germany for remote monitoring research and about 60 subjects carried out 8-24 hour remote monitoring tests. In the questionnaire, over 90

However, this paper does not address how to process the context conflicts in the process of constructing ontology and reasoning conditions. The reliability and energy consumption problem of remote medical systems will be the focus in further research. As for the energy consumption problem, some new ultra low power transmission protocols that can greatly improve the sustainability of remote monitoring have appeared, such as ANT (ANT is a proven ultra-low power (ULP) wireless protocol that is responsible for sending information wirelessly from one device to another device, in a robust and flexible manner) and BLE (Bluetooth Low Energy). Other challenging tasks will include the system recovery mechanism and development of an intelligent error discovery to keep the stability of the system over time.

7 Acknowledgement

The authors wish to thank the Federal Ministry of Education and Research (BMBF, Germany) for the financial support (FKZ: 01Z1KN11).

Bibliography

[1] Jovanov, E., Milosevic, M., Milenkovic, A.(2013); A mobile system for assessment of physiological response to posture transitions, *Proc. of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, EMBS, art. no. 6611220, 7205-7208.

[2] Berndt, R.-D., Takenga, M.C., Kuehn, S., Preik, P., Stoll, N., Thurow, K., Kumar, M., Weippert, M., Rieger, A., Stoll, R.(2011); A scalable and secure telematics platform for the hosting of telemedical applications. Case study of a stress and fitness monitoring, *IEEE 13th International Conference on e-Health Networking, Applications and Services, HEALTHCOM*, art. no. 6026726, 118-121.

[3] Dey, A.K., Abowd, G.D., Salber, D.(2001); A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications, *Human-Computer Interaction*, 16 (2-4): 97-166.

[4] Vajirkar, P., Singh, S., Lee, Y.(2003); Context-aware data mining framework for wireless medical application, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2736, 381-391.

[5] Ito, T.(2013); An approach of body movement-based interaction towards remote collaboration, *20th ISPE International Conference on Concurrent Engineering, CE - Proceedings*, 163-172.

[6] Hossain, M.A., Alamri, A., Parra, J.(2013); Context-aware elderly entertainment support system in assisted living environment, *Electronic Proc. of the IEEE International Conference on Multimedia and Expo Workshops*, ICMEW 2013, art. no. 6618365.

[7] Van Cauwenberg, J., Van Holle, V., De Bourdeaudhuij, I., Clarys, P., Nasar, J., Salmon, J., Maes, L., Goubert, L., Van de Weghe, N., Deforche, B.(2014); Physical environmental factors that invite older adults to walk for transportation, *Journal of Environmental Psychology*, 38: 94-103.

[8] Ogawa, Mitsuhiro, Tamura, Toshiyo, Togawa, Tatsuo.(1998); Fully automated biosignal acquisition in daily routine through 1 month, *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, 4: 1947-1950.

[9] Manios, Y., Moschonis, G., Papandreou, C., Politidou, E., Naoumi, A., Peppas, D., Mavrogianni, C., Lionis, C., Chrousos, G.P. (2015); Revised Healthy Lifestyle-Diet Index and associations with obesity and iron deficiency in schoolchildren: The Healthy Growth Study, *Journal of Human Nutrition and Dietetics*, Suppl 2:50-8. doi: 10.1111/jhn.12183. Epub 2013 Dec 5.

[10] Jakkula, V.(2007); Predictive data mining to learn health vitals of a resident in a smart home, *Proceedings - IEEE International Conference on Data Mining*, ICDM, art. no. 4476662, 163-168.

[11] Gjevjon, E.R., Eika, K.H., Romoren, T.I., Landmark, B.F.(2014); Measuring interpersonal continuity in high-frequency home healthcare services, *Journal of Advanced Nursing*, 70(3): 553-563.

[12] Costin, Hariton, Rotariu, Cristian, et al. (2008); Complex Telemonitoring of Patients and Elderly People for Telemedical and Homecare Services, *Proc. of the 1st WSEAS Int. Conf. on Biomedical Electronics and Biomedical Informatics*, Rhodes, GREECE, AUG 20-22, 2008, Edited by: Long, CA; Anninos, P; Pham, T; et al., Book Series: Recent Advances in Biology and Biomedicine, 183-187.

[13] Berkow, M., Monsere, C.M., Koonce, P., Bertini, R.L., Wolfe, M. (2009); Prototype for data fusion using stationary and mobile data, *Transportation Research Record*, 102-112.

[14] Ermes, M., Parkka, J., Mantyjarvi, J., Korhonen, I.(2008); Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions, *IEEE Transactions on Information Technology in Biomedicine*, 12 (1): 20-26.

[15] Heinz, EA., Kunze, KS., Gruber, M., Bannach, D., Lukowicz, P. (2006); Using Wearable Sensors for Real-Time Recognition Tasks in Games of Martial Arts-An Initial Experiment, *Computational Intelligence and Games*, IEEE , 98-102.

[16] Han, S.N., Lee, G.M., Crespi, N. (2014); Semantic context-aware service composition for building automation system, *IEEE Transactions on Industrial Informatics*, art. no. 6478809, 10 (1): 252-261.

[17] Zhang, S., McClean, S., Scotney, B., Galway, L., Nugent, C. (2014); A framework for context-aware online physiological monitoring, *Proceedings - IEEE Symposium on Computer-Based Medical Systems*, art. no. 5999038.

[18] Abe, B.T., Olugbara, O.O., Marwala, T. (2014); Classification of hyperspectral images using machine learning methods, *Lecture Notes in Electrical Engineering*, 247 LNEE, 555-569.

[19] Zhang, W., Thurow, K., Stoll, R.(2013); A SOA and knowledge-based telemonitoring framework: Design, modeling, and deployment, *International Journal of Online Engineering*, 9 (6): 48-57.

[20] Sannino, G., De Pietro, G. (2011); A smart context-aware mobile monitoring system for heart patients, IEEE International Conference on Bioinformatics and Biomedicine Workshops, BIBMW 2011, art. no. 6112448, 655-659.

[21] She, J., Nakamura, H., Makino, K., Ohyama, Y., Hashimoto, H., Wu, M. (2013); Experimental selection and verification of maximum-heart-rate formulas for use with karvonen formula, ICINCO - Proceedings of the 10th International Conference on Informatics in Control, *Automation and Robotics*, 2: 536-541.

[22] Hidalgo Limited, Equivital EQ-01 Vital Signs Monitor Health Care Practioner Guide. Dokumentation, Hidalgo Ltd., 2006.

[23] M. Kumar, M. Weippert, R. Vilbrandt, S. Kreuzfeld, and R. Stoll, (2007); Fuzzy evaluation of heart rate signals for mental stress assessment, *IEEE Transactions on Fuzzy Systems*, 15: 791-808.

[24] K Aminian, B Najjafi, (2002); Body movement monitoring system and method, US.Patient EP 119513910.

[25] Fuentes, D., Gonzalez-Abril, L., Angulo, C., Ortega, J. A. (2012); Online motion recognition using an accelerometer in a mobile device, *Expert systems with applications*, 39(3): 2461-2465.

[26] Godfrey, A., Bourke, AK., Olaighin, GM. (2011); Activity classification using a single chest mounted triaxial accelerometer , *Medical Engineering and Physics*, 33(9): 1127-1135.

[27] Fleury, A., Noury, N., Vacher, M. (2009); A wavelet-based pattern recognition algorithm to classify postural transitions in humans, *17th European Signal Processing Conference*, 2047 - 2051.

Author index