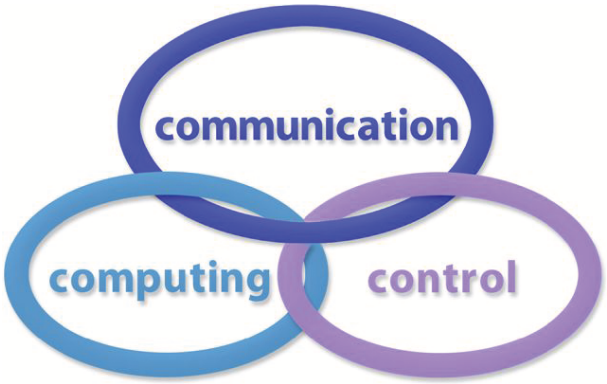
INTERNATIONAL JOURNAL

of

COMPUTERS COMMUNICATIONS & CONTROL

ISSN 1841-9836



A Bimonthly Journal
With Emphasis on the Integration of Three Technologies

Year: 2015 Volume: 10 Issue: 2 (April)

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

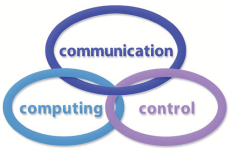
CCC Publications - Agora University Editing House

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:

1. Since 2006, Vol. 1 (S), IJCCC is covered by Thomson Reuters and is indexed in ISI Web of Science/Knowledge: Science Citation Index Expanded.
2. Journal Citation Reports (JCR - Science Edition), IF = 0.694 (JCR2013).
   Subject Category:
   (a) Automation & Control Systems: Q4 (46 of 59);
   (b) Computer Science, Information Systems: Q3 (96 of 135).
3. Since 2008, 3(1), IJCCC is covered in Scopus, SJR2013 = 0.231, H index = 13.
   Subject Category:
   (a) Computational Theory and Mathematics: Q4;
   (b) Computer Networks and Communications: Q3;
   (c) Computer Science Applications: Q3.
4. 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:

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

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/

Gheorghe Păun
Institute of Math. of Romanian Academy
Bucharest, PO Box 1-764, Romania
gpaun@us.es

Mario de J. Pérez Jiménez
Dept. of CS and Artificial Intelligence
University of Seville, Sevilla,
Avda. Reina Mercedes s/n, 41012, Spain
marper@us.es

Dana Petcu
Computer Science Department
Western University of Timisoara
V.Parvan 4, 300223 Timisoara, Romania
petcu@info.uvt.ro

Radu Popescu-Zeletin
Fraunhofer Institute for Open
Communication Systems
Technical University Berlin, Germany
rpz@cs.tu-berlin.de

Imre J. Rudas
Óbuda University
Budapest, Hungary
rudas@bmf.hu

Yong Shi
School of Management
Chinese Academy of Sciences
Beijing 100190, China &
University of Nebraska at Omaha
Omaha, NE 68182, U.S.A.
yshi@gucas.ac.cn, yshi@unomaha.edu

Athanasios D. Styliadis
University of Kavala
Institute of Technology
65404 Kavala, Greece
styliadis@teikav.edu.gr

Gheorghe Tecuci
Learning Agents Center
George Mason University
U.S.A.
University Drive 4440, Fairfax VA
tecuci@gmu.edu

Horia-Nicolai Teodorescu
Faculty of Electronics and
Telecommunications
Technical University "Gh. Asachi" Iasi
Iasi, Bd. Carol I 11, 700506, Romania
hteodor@etc.tuiasi.ro

Dan Tufiş
Research Institute for Artificial Intelligence
of the Romanian Academy
Bucharest, "13 Septembrie" 13, 050711,
Romania
tufis@racai.ro

Lotfi A. Zadeh
Director,
Berkeley Initiative in Soft Computing (BISC)
Computer Science Division
University of California Berkeley,
Berkeley, CA 94720-1776
U.S.A.
zadeh@eecs.berkeley.edu

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

International Journal of Computers Communications & Control
ISSN 1841-9836, Volume 10, Issue 2, April, 2015.

Contents

Bearing-Opportunistic Network Coding
K. Alic, E. Pertovt, A. Svigelj **154**

Control System Architecture for a Cement Mill Based on Fuzzy Logic
C.R. Costea, H.M. Silaghi, D. Zmaranda, M.A. Silaghi **165**

Web Service Composition Framework using Petrinet and Web Service Data Cache in MANET
M. Deivamani, S.R. Murugaiyan, V. Ravisankar, P. Victer Paul, R. Baskaran, P. Dhavachelvan **174**

Zero-watermarking Algorithm for Medical Volume Data Based on Difference Hashing
B.R. Han, J.B. Li, Y.J. Li **188**

HAPA: Harvester and Pedagogical Agents in E-learning Environments
M. Ivanović, D. Mitrović, Z. Budimac, L. Jerinić, C. Bădică **200**

Modelling and Analysis of Mobile Computing Systems: An Extended Petri Nets Formalism
L. Kahloul, A. Chaoui, K. Djouani **211**

Logging for Cloud Computing Forensic Systems
A. Pătraşcu, V.V. Patriciu **222**

Implementing BPMN 2.0 Scenarios for AAL@Home Solution
L. Rusu, B. Cramariuc, D. Benta, M. Mailat **230**

Multicriteria Supplier Classification for DSS: Comparative Analysis of Two Methods
J.M. Sepulveda, I.S. Derpich **238**

A Comprehensive Trust Model Based on Multi-factors for WSNs
N. Wang, Y. Chen **248**

QEAM: An Approximate Algorithm Using P Systems with Active Membranes
G. Zhang, J. Cheng, M. Gheorghe, F. Ipate, X. Wang **263**

Human-Manipulator Interface Using Hybrid Sensors via CMAC for Dual Robots
P. Zhang, G. Du, B. Liang, X. Wang **280**

Author index **291**

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):154-164, April, 2015.

Bearing-Opportunistic Network Coding

K. Alic, E. Pertovt, A. Svigelj

Kemal Alic*, Erik Pertovt, Ales Svigelj
Department of Communication Systems, Jozef Stefan Institute
Jozef Stefan International Postgraduate School
Jamova cesta 39, 1000 Ljubljana, Slovenia
kemal.alic@ijs.si, erik.pertovt@ijs.si, ales.svigelj@ijs.si
*Corresponding author: kemal.alic@ijs.si

Abstract: This paper presents a novel, practical, routing-independent network-coding algorithm: BON-Bearing opportunistic network coding. Simplicity is its main benefit as it introduces little overhead to the network since nodes do not need to keep track of received traffic for their neighbouring nodes. Algorithm makes coding decisions based solely on the information about the packet previous and next hop node position. Algorithm functions between the MAC and link layers, with small modifications made only to the MAC layer. Using different topologies and different traffic loads and distributions in the simulation model we evaluated algorithm performance and compared it to a well-known COPE algorithm.
Keywords: Algorithms, network layer network coding, network throughput, performance evaluation.

1 Introduction

Network coding (NC) introduced by Ahlswede [1] brings a promising approach to improving network throughput and performance. The classical communication networks paradigm of nodes only forwarding packets is metamorphosed by a simple and important premise that nodes also process the incoming packets [2].

NC was originally proposed in order to achieve multicast data delivery at the maximum data transfer rate in single-source multicast networks [3]. We soon began to see NC ideas being put to use in problems other than multicast networks. Increasing throughput in satellite networks [4, 5] and P2P networks [6, 7], improving delivery reliability over the lossy links either in wireless networks over TCP [8] or in Delay Tolerant Networks such as deep space links [9] all bring promising results. Depending on the application of NC the implementation affects different OSI layers. In multicast scenarios NC is typically implemented in the application layer while two stage NC for increased spectrum efficiency is deployed in the physical layer [10].

We are interested in opportunistic NC for static wireless mesh networks such as metropolitan WiFi networks [11] for unicast traffic. Benefits of opportunistic NC can be best explained with a help of a simple example depicted in in Figure 1. Consider the following situation: Node 1 (N1) has a packet P1 for Node 2 (N2) and Node 2 (N2) has a Packet P2 for N1. The nodes can exchange their packets through intermediate node (N3). N1 and N2 send packets P1 and P2 to N2. Without using NC the N3 first sends out P1 and later on P2. By using the NC procedure N3 performs a linear operation over the two packets (e.g. XOR) and sends out a coded packet P1$\bigoplus$P2. N1 and N2 XOR the received packet with the sent packet (P1 and P2) and obtain the packets headed to them. With NC, the number of transmissions nodes need to perform in order to deliver both packets to their destinations has been reduced from 4 to 3.

The question which packets the coding node can encode together in general network deployment in order that the receiving nodes will be able to decode the coded packet, which is the key issue in this paper. Optimal integration of NC into the existing architecture of a network is not straightforward and its implementation influences all functional components in current

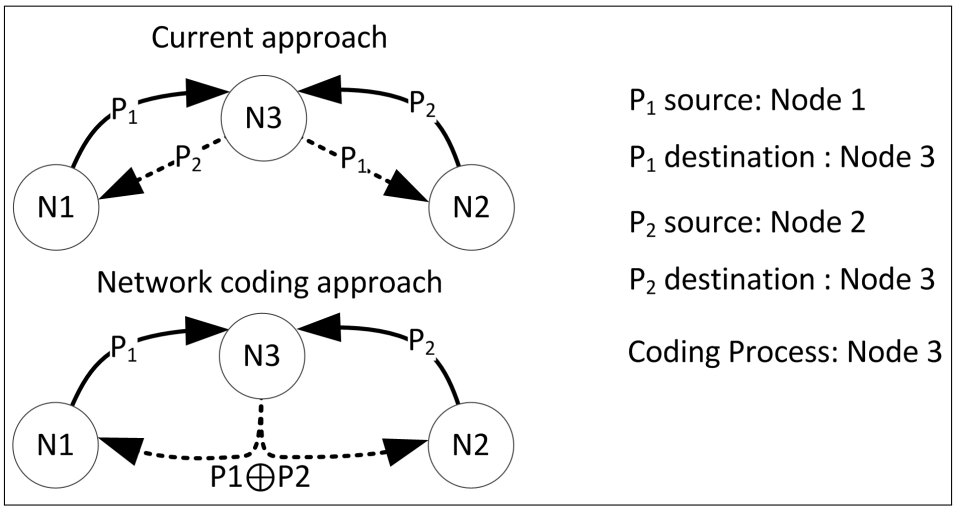


Figure 1: Simple example on how opportunistic NC increases throughput.

network protocol stack e.g. scheduling, routing and congestion control [12]. Though, interaction with all these layers increases the implementation difficulty and possibly also the computational complexity. Henceforth, a trade-off needs to be made between coding and complexity. Hence, the objective of this paper is to propose a novel inter-session NC schema that is easily implementable in the real world, can be used in the current OSI structure and improves the network performance.
Thus, in this article we propose a novel Bearing-Opportunistic Network coding (BON) algorithm, which main advantages are the following: (i) it can be seen as an individual OSI layer between the MAC and the network layers; (ii) it introduces little overhead to the network; (iii) it works for all traffic types; (iv) it is distributed; (v) it does not need to record information about the received packets on individual neighbouring nodes and it is (vi) routing independent.

The rest of this paper is organized as follows. In Section 2 related work is described. In Section 3 the BON coding process is explained and its implementation into OSI model is explained in Section 4. Performance evaluation results are given in Section 4. Section 5 concludes the article and gives insight into further work.

2 Related work

Practical NC has mainly been influenced by an excellent and intuitive algorithm COPE (Coding Opportunistically) [13] which principles have been adapted and extended also for covering other ideas in NC: for example in [14] noCoCo algorithm specializes in bidirectional traffic flows. It is trying to maximize the number of coding opportunities for the two opposite direction routes. CLONE [15] generalized COPE to address multiple unicast sessions. The system takes into account lossy links and highlights specific situations where COPE provides no coding gain. COPE coding decisions are classified and improved thus making better coding decisions by K. Chi et.all [16]. In [17] the MORE and COPE principles have been joint in search of benefits of the two at the same time. Making routing aware of COPE coding opportunities has been investigated in [18] and [19], while [20] investigated new metric schemas for coding aware wireless routing. In DCAR architecture [21] coding is based on COPE, though extended and supported by route discovery.

COPE procedure introduced NC between the MAC and network layers with small modifications made only to the MAC layer. COPE takes advantage of the broadcast nature the wireless

channel to perform opportunistic listening and encoded packet broadcasting so that the number of packet transmissions can be reduced.

In COPE, every node keeps information for each of its neighbours about packets already received. Information can be gathered through ACK packets and through update packets. The later introduces additional overhead as update packets are broadcasted by individual nodes to all their neighbours. Owing to these facts, nodes require additional hardware functionalities and the network requires handling of additional packets which on one hand results in lower network capacity, and is on the other hand undesirable in energy constrained networks such as Wireless Sensor Networks (WSN). A fair share of coding decisions in COPE is based on the delivery probabilities between the nodes. These are calculated in the routing process in the routing protocols based on the ETX metrices. Hence for COPE implementation access to the routing layer is needed, which is not always possible, especially when combining equipment from different vendors.

3 Network Coding Process

Essentially the Coding Process is about making decisions on which packets the coding node can encode together in order that the receiving nodes will be able to decode the coded packet. The more packets we code together or more loos the conditions in coding process are the higher the possibility of receiving nodes not being able to decode packets. Though, if coding conditions are hard to meet, than there are few coding opportunities and low bandwidth saving benefits.

3.1 General coding conditions

BONCoding process is based on the local bearing of packets. We define packet bearing on a coding node as the unit vector showing direction between previous hop and next hop of the packet.

Let us consider the situation depicted in Figure 2. Packet P1 has already been transmitted from node 1 (N1) to node 2 (N2) and is destined to node 3 (N3). Packet Pn has already been transmitted from node 4 (N4) to N2. P1 and Pn are now waiting in the output queue on the coding node N2. When N2 gains channel access, it tries to encode P1 and Pn based on the local bearing process. On the coding node N2 local bearing of the packet P1 depends on the positions of its previous hop N1 and its next hop N3. Since the coding node is familiar with the node positions of the packet previous hop and the packet next hop, the coding node can calculate the direction vector $\vec{a}$ which represents the direction in which the packet P1 is travelling. In Cartesian coordinate system we calculate the angle α for P1 between the x-axis and the direction vector $\vec{a}$ which is between 0 and 2π.

After calculating the local bearing for the first packet P1, the algorithm goes through the remaining packets in the output queue searching for a matching packet to be encoded with P1. For each of the possibly matching packets Pn the angle β between the x-axis and its direction vector $\vec{b}$ are calculated using the same principle as described above for the first native packet P1. Coding conditions are met for packets P1 and Pn, if (1) is true:

$$\pi - \epsilon \leq |\alpha - \beta| \leq \pi + \epsilon \tag{1}$$

where ϵ is the tolerance angle. In a given situation N1 can decoded packet since it is the source of half of the information encoded in the P1$\bigoplus$Pn. The question is whether N3 can decode this packet. The probability that N3 will successfully decode P1$\bigoplus$Pn is the same as probability that N3 has overheard the Pn transmission from N4 to N2. We may assume that nodes on one side of the coding node have the knowledge of the content of packets of the first flow, while nodes

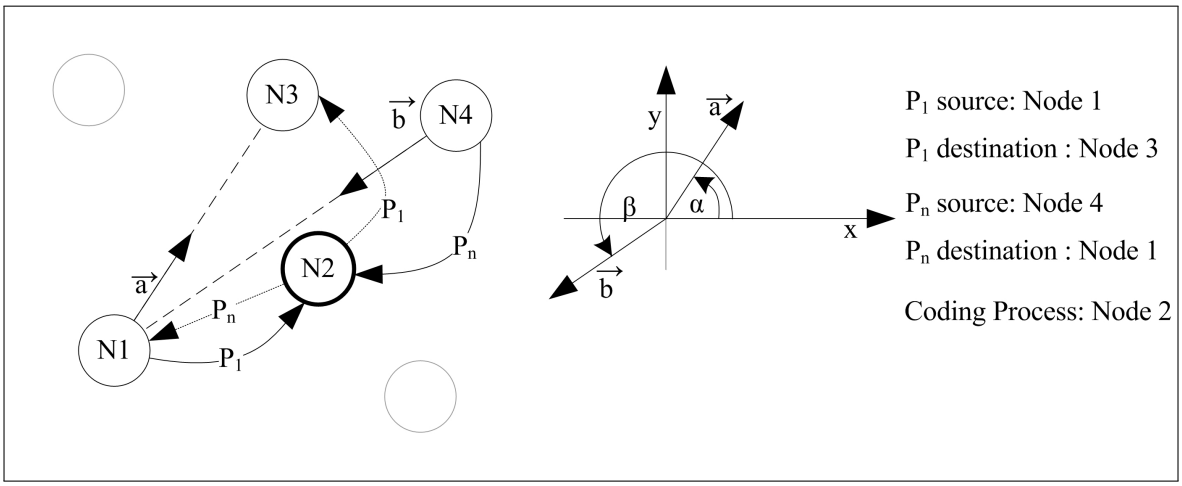


Figure 2: Coding process: the calculation of parameters for encoding evaluation for two packets.

on the other side of the coding node recognise the content of packets from the second flow, thus matching them up for encoding. Nodes that are placed in approximately the same bearing from the coding node, can also hear each other transmissions, thus being able to decode the packet. By increasing ϵ the possibility of encoding multiple packets and that one of the receptionists will not be able to decode the packet is increased. By lowering ϵ towards zero coding opportunities are reduced but the possibility of successful packet decoding on receiving nodes is increased. When two packets belonging to two traffic flows that are headed into opposite directions meet on the relay node, (1) is true even for $\epsilon = 0$ and receiving nodes will always be able to decode the encoded packet.

BON allows coding of multiple packets. Multiple packets are encoded into one packet when the expression (1) is true for all packet pairs to be encoded. If $\epsilon = 0$ two packets can be encoded at the most.

4 Network Coding Procedure

4.1 Packet coding

Each node maintains two packet queues. The highest priority queue is used for signalization packets and the lower priority queue is for packets carrying information through the network. If signalization queue is empty the coding algorithm takes the packet that is at the head of the queue and then searches the rest of the queue looking for possible coding matches based on the process described in Section 3. If a coding opportunity is found, the packets are encoded using XOR operation into a coded packet accompanied with the headers for the decoding process at the receiving nodes. If no coding opportunities are found the packet is transferred to the MAC layer as is and the native packet is transmitted.

4.2 Pseudo-broadcast

In the MAC layer the pseudo-broadcast mechanism, first presented in [13] is used. The link-layer destination field is set to the MAC address of one of the intended recipients. Since all nodes are set in the promiscuous mode, they can overhear packets not addressed to them. When a node receives a packet with a MAC address identical to its own, it sends an ACK message to

the sender. Regardless of the address of the packet next hop the node pushes the packet to the NC module.

4.3 Packet reception

Upon the packet reception in the NC module further actions depend on whether the packet is coded or native (not coded). In the case of the coded packet the process checks the packet pool where all received and overheard packets are stored for decoding purposes to determine whether it has already received N-1 of the packets coded in the coded packet. If not the coded packet cannot be decoded and it is simply dropped. If the node has at least the required N-1 packets, i.e., enough information, it decodes the coded packet using these packets with the XOR operations, thus gaining a set of native packets. Each native packet is treated individually. From here on the process is the same as upon receiving a native packet. The process checks whether the node has already received the packet. If so it drops it. If the packet is new its copy is inserted into the packet pool for decoding purposes. It does so for every received native packet, as all received packets are potentially needed for further decoding purposes. The process checks whether the node is the next hop for the native packet. If so, and if the packet has been a part of the coded packet, ACK message is scheduled and the packet is sent to the upper layers for further processing.

4.4 Signalization

Since static nodes are under investigation, ACK messages are the only signalization packets required in BON. Depending on the application and expected gains, different ways of sending ACK messages can be used. Since we are optimising a network from the throughput perspective we adopt the structure proposed in [13]. ACK messages are periodically broadcasted as cumulative reports. If opportunity arises the ACK reports are attached to the regular outgoing packets, thus reducing the overhead.

5 Performance Evaluation

5.1 Simulation Setup

We evaluated BON using a simulation model built in OPNET Modeler [22] for analysis of network layer NC algorithms that has already been presented in [23].

We conducted a set of simulations using the following settings: 40 static nodes randomly placed on 4 km x 4 km area; 800, 900 and 1000 m of transmission range for three topologies T1, T2 and T3, respectively. The evaluated topologies are presented in Figure 3, where dashed lines indicate wireless links between nodes. All nodes receive and transmit on the same channel and each node has 2 Mbit/s of channel bandwidth.

BON and COPE use up to one retransmission in the wireless module and up to two retransmissions in the NC layer. Both algorithms store packets in the packet pool for 20 seconds after the reception, thus making sure packet decoding did not fail due to delays in the network. Output queues are limited to 50 packets.

BON and COPE send out cumulative ACKs every 0.5s. NC layer packet retransmissions are scheduled 0.8 s after the initial packet transmission. COPE update packets as described in [13] are sent out at least every 0.5 s. When possible, update packets are attached to regular outgoing packets, or cumulative ACKs thus introducing less overhead to the network.

ϵ has been set at 25 for BON, while ϵ for evaluating coding condition in packet retransmission has been set to half of the initial value.

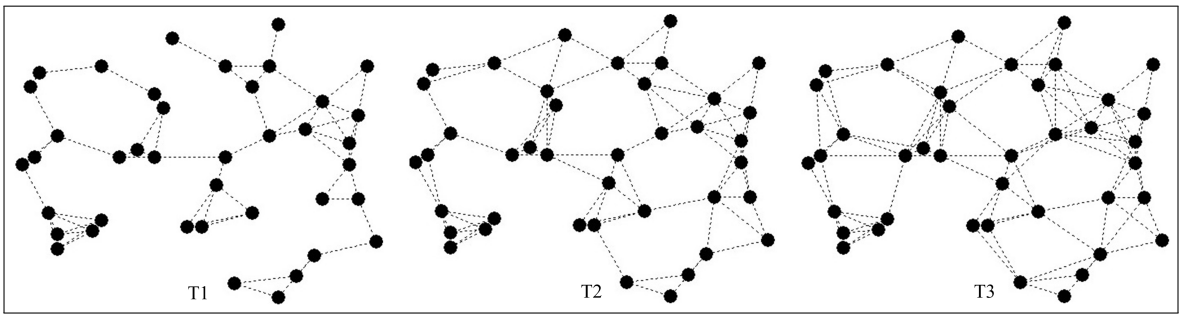


Figure 3: Wireless nodes connectivity for three evaluated topologies (namely T1, T2 and T3).

Simulation runs took 150 s. Results were collected between 60th and 120th second to observe steady-state conditions. Through each simulation run we observed network with one load. For each load we made three simulation runs using three different seeds. For the first 30 s traffic load was set at 10 % of the final load, between 30th and 60th second the load was set to 80 % of the final load (i.e. warm up time). Network was loaded with full load and results were collected between the 60th and 120th second. Routing tables are calculated at the beginning of each simulation run and are updated every 30 s. Dijkstra algorithm is used for routes calculation taking into account ETX metric, which is also required by COPE.

UDP like traffic has been generated. Packet sizes vary between 45 and 1500 bytes. Packet size distribution has two peaks and follows the measured Internet traffic as presented in [24]. Exponential distribution is used for calculating packet inter-arrival times.

Two traffic flow distributions have been used in the process of evaluation. The first, the Uniform distribution scenario, is where all nodes generate traffic flows and destine them to all nodes in the network (uniform distribution is used to generate packet destinations). In the second, the Gateways scenario, nodes communicate with its nearest gateway only. Nodes and gateways generate symmetric traffic flow. Three selected nodes from topologies have been given the base station functionality. Delivery probability (DP) between nodes depends on the amount of traffic between the neighbouring nodes and the distance between nodes (e.g. for T3, load 1.9 Mbit/s, uniform traffic distribution, BON algorithm average DP is 0.88, max DP 0.97, min DP 0.78 and median for DP is 0.87).

We have compared BON algorithm to the well-known COPE and to reference scenario (ref.sc.) where no coding was used.

5.2 Simulation results

For Uniform traffic flow distribution for all topologies we have observed average goodput (Figure 4) highlighting results important from the operator point-of-view while End-to-End (ETE) Delay (Figure 5) shows results important from the end-user perspective. Normalised Number of Wireless Transmissions (NNWT) which is the ratio between the number of packets received from NC layer by the wireless module and number of packets received in the application layer (Figure 6) shows the number transmissions needed in the wireless module for every goodput packet. In ideal link conditions NNWT is a sum of network average diameter and overhead in terms of standalone packets. We use this measure for overhead comparison reasons. Number of packet retransmissions in the NC layer indicates how well NC algorithms coded packets together (Figure 7). Maximal normalised gain which is the highest ratio between the number of goodput packets received using coding algorithm and the number of goodput packets received in ref. sc. (Table 1) shows highest measured gain when using NC. Each dot on the plot represents the average of three simulation runs with different seeds of pseudo random generator.

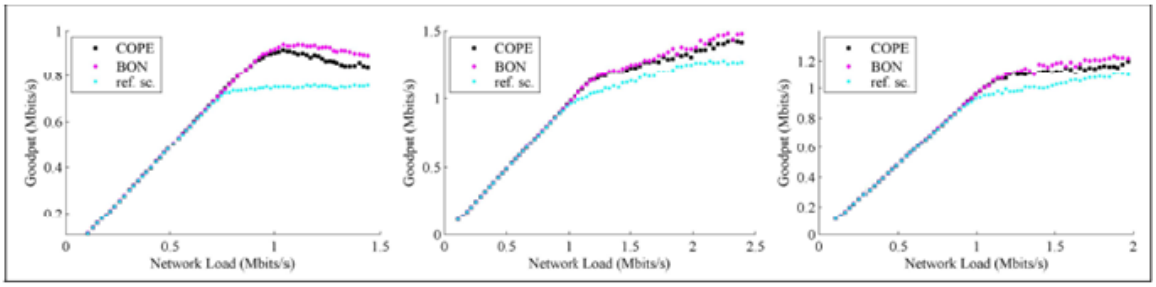


Figure 4: Goodput in dependency of network load for BON, COPE and ref.sc. for topologies T1, T2 and T3 in Uniform traffic distribution.

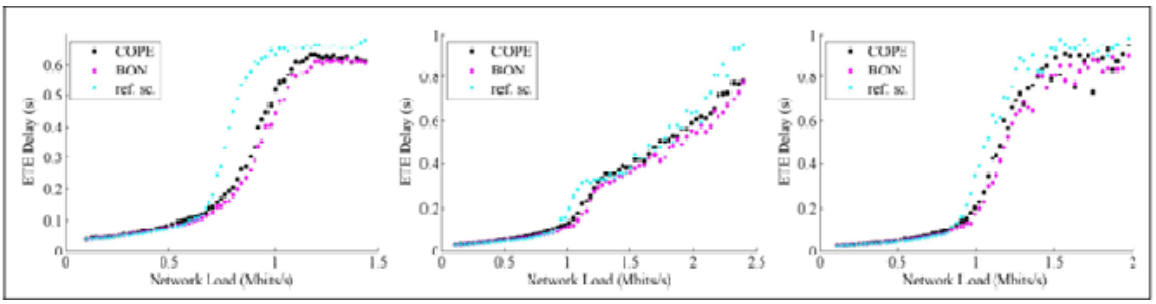


Figure 5: Delay in dependency of network load for BON, COPE and ref.sc. for topologies T1, T2 and T3 in Uniform traffic distribution.

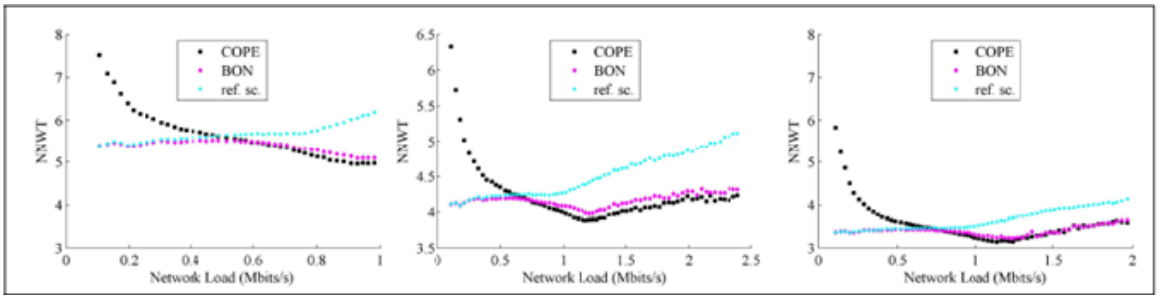


Figure 6: Normalised number of wireless transmissions in dependency of network load for BON, COPE and ref.sc. for topologies T1, T2 and T3 in Uniform traffic distribution.

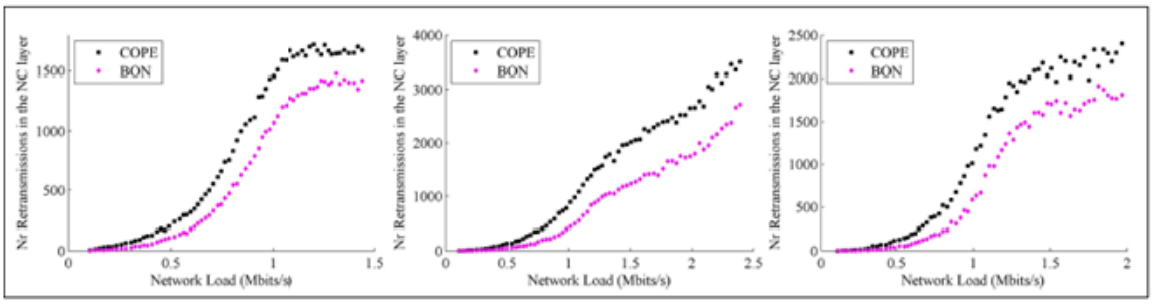


Figure 7: Nr Retransmissions in the NC layer in dependency of network load for BON and COPE for topologies T1, T2 and T3 in Uniform traffic distribution.

Table 1: Maximal normalised gain for BON and COPE for T1, T2 and T3 in Uniform traffic distribution.

	Maximal normalised gain		
	T1	T2	T3
COPE	1.22	1.12	1.14
BON	1.26	1.17	1.17

From the Goodput in dependency of network load plot (Figure 4) for all topologies we can observe that with low traffic loads goodput is approximately the same for both coding algorithms as well as with the reference scenario. However, by increasing the traffic load, we can see that both COPE and BON significantly improve the network performance as compared to the ref.sc Scenario, and that BON can provide the highest goodput. The gain itself also depends on the network load and observed topology. Table 1 shows that the Maximal normalised gain (ratio between number of packets received using coding algorithm and number of packets received in ref. sc) and that BON outperforms COPE for additional 4 to 5 percentage points.

From the End-to-End Delay in dependency of network load plot (Figure 5) we can observe that with low traffic loads the delay is low and approximately the same for BON and the reference scenario, while delay with COPE is slightly higher. By increasing the network load the delay increases for all three scenarios. The delay increases the fastest with ref.sc. COPE follows and BON keeps the lowest delay of the three for almost all loads and all three topologies. Higher delay with COPE in lower loads can be explained with the help of plots in Figure 6 (Normalised number of wireless transmissions). In low traffic loads COPE finds few opportunities to attach update packet to regular outgoing packets thus introducing higher overhead to the network. With the increased load COPE and BON need to transmit approximately the same number of packets in the wireless module in per application layer packet. COPE keeps the ratio slightly lower than BON with high loads. This is due to the fact that BON outperforms COPE in throughput gains and due to limited queue sizes. Packets travelling higher node distances are more likely to be dropped due to full queues. The ratio for both algorithms is much better in the congested network where NC benefits are at its best.

BON better performance in comparison to COPE can be explained also with the number of retransmissions in the NC layer. Retransmissions in the NC layer occur in case of reception node unsuccessful decoding or errors in the transmission (pseudo broadcast demands ACK message only from one of the recipient nodes). Hence, number of retransmitted packets indicates how successful the algorithm was in coding together packets that can be decoded on their next hop: BON needs fewer retransmissions than COPE and we may claim that it makes better coding decisions.

In Figure 8 and Figure 9 the results for the Gateways scenario are presented. Results were collected in the same way as for Uniform traffic case. We present only results for network goodput and End-to-End Delay as this are the most representative results; other results bring up the same conclusions as in the Uniform distribution scenario. In comparison to the Uniform traffic flow distribution coding gain benefits are lower for the flows directed from and to the gateways. This is due to the changed traffic conditions. Packets travel shorter hop distances between their source and destination, providing less coding opportunities and thus fewer possibilities for goodput improvements. Still, the BON algorithm can provide the highest goodput while keeping the delay lowest. Table 2 shows the Maximal normalised gain and that BON can bring additional 2 to 8 percentage points of goodput increase in comparison to COPE.

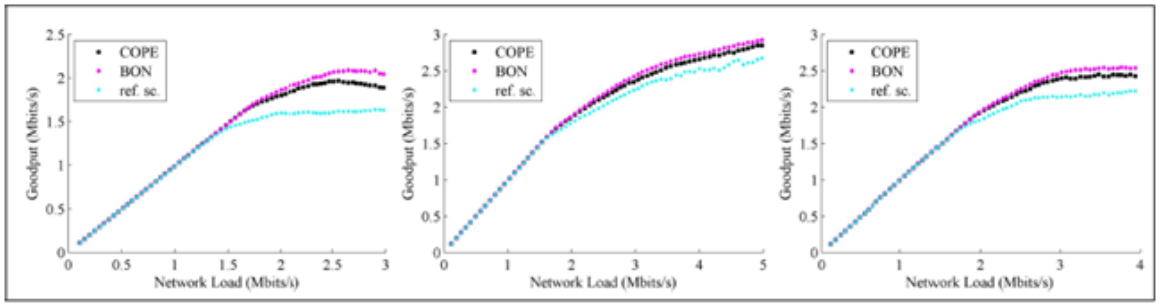


Figure 8: Goodput in dependency of network load for BON, COPE and ref.sc. for topologies T1, T2 and T3 in Gateways traffic distribution.

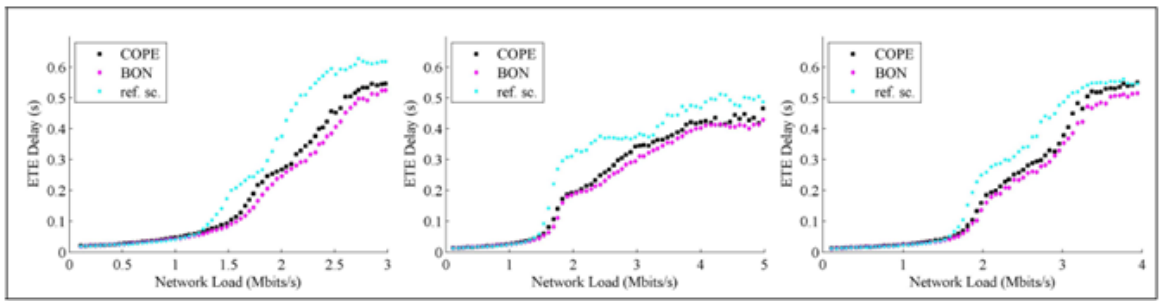


Figure 9: Delay in dependency of network load for BON, COPE and ref.sc. for topologies T1, T2 and T3 in Gateways traffic distribution.

Table 2: Maximal normalised gain for BON and COPE for T1, T2 and T3 in Gateways traffic distribution.

	Maximal normalised gain		
	T1	T2	T3
COPE	1.21	1.09	1.13
BON	1.29	1.11	1.18

6 Conclusions and further work

BON is a novel, general-purpose algorithm which works in the broadcast networks between the network and the MAC layer. It is applicable to the static networks such as WSN networks and metropolitan WiFi networks. Small modifications are made only to the MAC layer while other OSI layers remain unchanged (the same as in COPE). We assume that each node is familiar with positions of all of its neighbours which is easy to obtain as nodes are static.

BON is routing independent and as presented in the results it introduces very little overhead to the network, thus using less radio resources and consequently less energy. For nodes without any coding opportunities or generally in networks with low loads the overhead is the same as in the reference scenario. Since nodes do not need to save the information about packets received on neighbour nodes, the coding process is fairly simple. In addition, we have shown in the results that BON outperforms COPE in terms of increasing the network goodput while keeping the network delay lower for different topologies and traffic distributions.

In-depth analysis shows that COPE finds more coding opportunities, though incorrect coding decisions are more often made than in BON. Furthermore, lower overhead in BON also results

in higher good put.

Main drawback for using BON is its requirement of knowing the positions of the nodes; although the networks where nodes know their position are increasing rapidly e.g. in backhaul networks such as metropolitan WiFi where end-users use different part of spectrum for communication.

In addition, setting the tolerance angle might not appear straightforward. For static networks, such as in our case, this can be done through simulation or emulation, though through experience the approximate value of the optimal Îľ can be set by carefully observing the topology. By adapting the algorithm to automatically set the tolerance angle on individual nodes, planning as a future work, seems an important improvement towards additional practical value of BON.

BON low overhead makes BON suitable for studying energy efficiency in wireless sensor or similar networks, especially in load cases when network is not congested. Besides that, its routing independency shows potential for modifying routing metrics.

Bibliography

[1] R. Ahlswede, N. Cai, S.-y. R. Li, and R. W. Yeung (2000); Network Information Flow, *IEEE Transactions on Information Theory*, 46: 1204-1216.

[2] C. Fragouli and E. Soljanin (2008); *Network Coding Applications*, Hanover, MA, USA Now Publishers.

[3] T. Matsuda, T. Noguchi, T. Takine (2011); Survey of Network Coding and Its Applications,*IEICE Transactions on Communications*, vol. E94-B, 698-717.

[4] C. Hausl, O. Ican, and F. Rossetto (2012); Resource allocation for asymmetric multi-way relay communication over orthogonal channels, *EURASIP Journal on Wireless Communications and Networking*, 1-20.

[5] F. Vieira, S. Shintre, and J. A. Barros (2010); How feasible is network coding in current satellite systems?, *Advanced satellite multimedia systems conference (asma) and the 11th signal processing for space communications workshop (spsc)*, Cagliari, Italy, DOI: 10.1109/ASMS-SPSC.2010.5586880, 31-37.

[6] D. Niu and B. Li (2011); Asymptotic Optimality of Randomized Peer-to-Peer Broadcast with Network Coding, *INFOCOM, Shanghai, China, 2011*, 1-9.

[7] B. Li and D. Niu (2011); Random Network Coding in Peer-to-Peer Networks: From Theory to Practice, *Proceedings of the IEEE*, 99: 513-523.

[8] J. Chen, L. Liu, X. Hu, and W. Tan (2011); Effective Retransmission in Network Coding for TCP, *International Journal of Computers Communications & Control*, 6(1):53-62.

[9] S. Haoliang, L. Lixiang, and H. Xiaohui (2011); A Network Coding based DTN Convergence Layer Reliable Transport Mechanism over InterPlaNetary Networks, *International Journal of Computers Communications & Control*, 6(2):236-245.

[10] K. Yasami, A. Razi, and A. Abedi (2012); Analysis of Channel Estimation Error in Physical Layer Network Coding, *IEEE Communications Letters*, 15: 1907-1910.

[11] S. Chieochan and E. Hossain (2012); Network coding for unicast in a WiFi hotspot: Promises, challenges, and testbed implementation, *Computer Networks*, 56(2): 2963-2980.

[12] L. You, L. Ding, P. Wu, Z. Pan, H. Hu, M. Song, et al. (2011); Cross-layer optimization of wireless multihop networks with one-hop two-way network coding, *Computer Networks*, 55(8): 1747-1769.

[13] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft (2008); XORs in the Air: Practical Wireless Network Coding, *IEEE/ACM Transactions on networking*, 16(3): 497-510.

[14] B. Scheuermann, W. Hu, and J. Crowcroft (2007); Near-optimal coordinated coding in wireless multihop networks, *2007 ACM CoNEXT conference, 2007*, DOI: 10.1145/1364654.1364666.

[15] S. Rayanchu, S. Sen, J. Wu, S. Banerjee, and S. Sengupta (2008); Loss-aware network coding for unicast wireless sessions: design, implementation, and performance evaluation, *ACM SIGMETRICS Performance Evaluation Review - SIGMETRICS '08*, 36(1): 85-96.

[16] K. Chi, X. Jiang, B. Ye, and Y. Li (2011); Flow-oriented network coding architecture for multihop wireless networks, *Computer Networks*, 55(10): 2425-2442.

[17] C. Qin, Y. Xian, C. Gray, N. Santhapuri, and S. Nelakuditi (2008); I2MIX: Integration of Intra-flow and Inter-flow Wireless Network Coding, *5th IEEE Annual Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks Workshops, 2008. SECON Workshops '08*, DOI: 10.1109/SAHCNW.2008.29.

[18] Y. Yan, B. Zhang, J. Zheng, and J. Ma (2010); CORE: a coding-aware opportunistic routing mechanism for wireless mesh networks, *IEEE Wireless Communications*, 17(3): 96-103.

[19] Z. Zhou and L. Zhou (2010); Network Joint Coding-Aware Routing for Wireless Ad Hoc Networks, *2010 IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS), 2010*, DOI: 10.1109/WCINS.2010.5541877, 17-21.

[20] L. Yifei, S. Cheng, X. Qin, and T. Jun (2009); ICM: a novel coding-aware metric for multi-hop wireless routing, *WiCOM'09 Proceedings of the 5th International Conference on Wireless communications, networking and mobile computing* NJ, USA: IEEE Press, DOI: 10.1109/WICOM.2009.5302189, 1-4.

[21] J. Le, J. C. S. Lui, and D.-M. Chiu (2010); DCAR: Distributed Coding-Aware Routing in Wireless Networks, *IEEE Transactions on Mobile Computing*, 9(4): 596-608.

[22] http://www.riverbed.com/products/performance-management-control/opnet.html.

[23] K. Alic, E. Pertovt, and A. Svigelj (2012); Network coding simulation model in OPNET modeler, in *OPNETWORK 2012*, Washington, USA.

[24] A. Svigelj, M. Mohorcic, G. Kandus, A. Kos, M. Pustisek, and J. Bester (2004); Routing in ISL Networks Considering Empirical IP Traffic, *IEEE Journal on Selected Areas in Communications*, 22(2): 261-272.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):165-173, April, 2015.

Control System Architecture for a Cement Mill Based on Fuzzy Logic

C.R. Costea, H.M. Silaghi, D. Zmaranda, M.A. Silaghi

Claudiu Raul Costea, Helga Maria Silaghi,
Doina Zmaranda*, Marius Alexandru Silaghi
University of Oradea
Romania, 410087 Oradea,Universitatii, 1
ccostea@uoradea.ro, hsilaghi@uoradea.ro,
dzmaranda@uoradea.ro, msilaghi@uoradea.ro
*Corresponding author: dzmaranda@uoradea.ro

Abstract: This paper describes a control system architecture for cement milling that uses a control strategy that controls the feed flow based on Fuzzy Logic for adjusting the fresh feed. Control system architecture (CSA) consists of: a fuzzy controller, Programmable Logic Controllers (PLCs) and an OPC (Object Linking Embedded for Process Control) server. The paper presents how a fuzzy controller for a cement mill is designed by defining its structure using Fuzzy Inference System Editor [1]. Also, the paper illustrates the structure of the implemented control system together with the developed PLC program and its simulation. Finally, the dynamic behavior of the cement mill is simulated using a MATLAB-Simulink scheme and some simulation results are presented.
Keywords: Control System Architecture (CSA), fuzzy controller, cement mill, fresh feed control, ball mill, feed change.

1 Introduction

The modern automation equipment is controlled by software running on Programmable Logic Controllers (PLCs). The classical closed loop control presents a long time until stable operation and slow reaction on interruption, but the modern fuzzy control presents a rapid stabilisation and a fast reaction on interruption. Process control is an essential part of the cement milling system. Development of an effective control strategy requires a good knowledge of the dynamics of the milling circuit. An effective process control system consists of: instrumentation, hardware peripherals and control strategy [4] . Ball mills rotate around a horizontal axis, partially filled with the material to be ground plus the grinding medium. In cement industry, stainless steel balls is used very often. An internal cascading effect reduces the material to a fine powder. Industrial ball mills can operate continuously, fed at one end and discharged at the other end [6]. Cement mill has two chambers, separated by a diaphragm. The purpose for using a diaphragm is because it divides mill into chambers or compartments. Also, allows operators to have different ball charges and liners in each and hence a different type of grinding action in each. Diaphragm controls the material flow from one compartment to the other, regulates partially the retention time and the degree of material filling in the grinding media voids. In the cement manufacturing process there are many equipments linked in the closed loop. The mill, that is a part of closed loop, has the longest delay. From the mill outlet, the product is transported by a bucket elevator to the air separator. Air separator is mainly used for raw materials and clinker classifying and setting up close-circuit grinding system with mill. The separator separates fine particles from coarse particles. The fine particles are collected as final product while the coarse particles are sent back for further grinding. In this section (mill and separator) large amounts of mass are being circulating: fresh feed flow, mill product, flow of rejected particles which is re-circulated to the mill inlet, final product (cement). Important values that present more attention are: bucket

elevator power, re-circulated flow (coarse return), clinker level in the first chamber of mill, clinker level in the second chamber of mill. Figure 1 presents the closed loop for grinding circuit with main interest points.

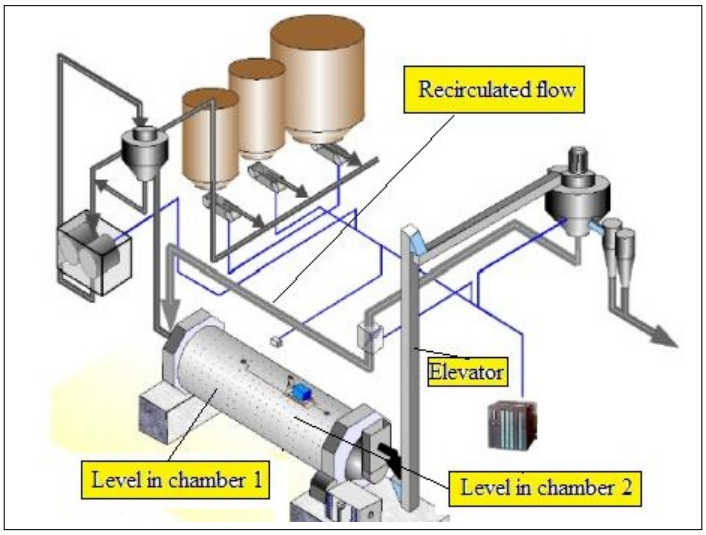


Figure 1: Closed loop for grinding circuit

The direct and precise measurement of levels inside the mill enables the fastest acquisition of any change in the grinding circuit and thus the fastest closed loop control imaginable.

2 The fuzzy system design

Taking into consideration the loop for grinding circuit of the cement mill, inputs and output of proposed fuzzy expert system used for grinding system control are presented in Figure 2 [1].

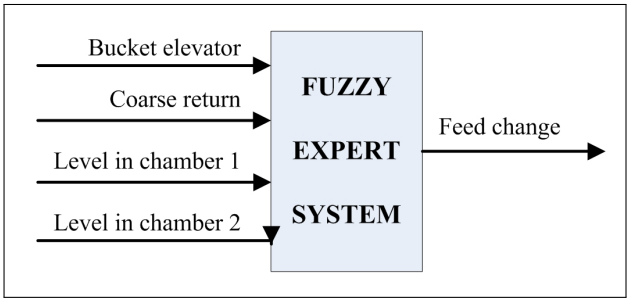


Figure 2: Inputs and output of fuzzy system

The output of the fuzzy controller is presumed to be described through singleton membership functions or by linear or nonlinear functions depending on the output process signal. The fuzzy structure consists on a fuzzy controller and a process [2, 3]. The structure presented in Figure 3 is being constructed by a number of r fuzzy blocks, connected in parallel FB1, FB2, ... FBr. The number r represents the number of the rules that define the fuzzy controller [2].

In Figure 3 the following notations are used: FC represent the fuzzy controller, FB is the fuzzy block, z is the controller input, u represent the command signal generated by the fuzzy controller, p is a vector that describes some possible external disturbances, x represents the state vector [1].

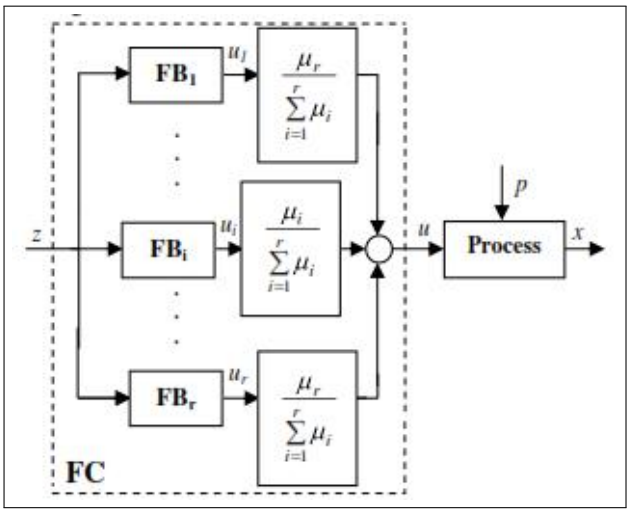


Figure 3: Fuzzy system structure with fuzzy controller decomposed by fuzzy rules

It was defined by the Wong team [7, 8] as a fuzzy subsystem associated to rule i, a system presumed to control the given process only by command ui. The command ui represents the output of fuzzy block associated to fuzzy rule i [2]. It was assumed that the fuzzy controller uses a rule basis consisting on r like rules. Each of these rules generates an output ui.

$$Rule\ i: IF premise\ i\ THEN u = ui, \tag{1}$$

with i = 1,2, ..., r.

Fuzzy systems are created based on three main steps. The first step is to define the input and output variables. The second step is to define the fuzzy subsets of each input and output variable and create membership functions. The third step is to define fuzzy rules that relate each input membership function to each output membership function [5]. Upon the completion of a fuzzy system, the fuzzy process will fuzzify an input, check each rule to find a degree of truth, and then defuzzify the result into an output value.

Using Fuzzy Inference System (FIS) Editor from MATLAB, the fuzzy system structure was defined by four inputs (bucket elevator power, coarse return, clinker level in the first chamber of mill, clinker level in the second chamber of mill) and one output (feed change). The fuzzy system structure, as it is defined by Fuzzy Inference System Editor, is presented in Figure 4.

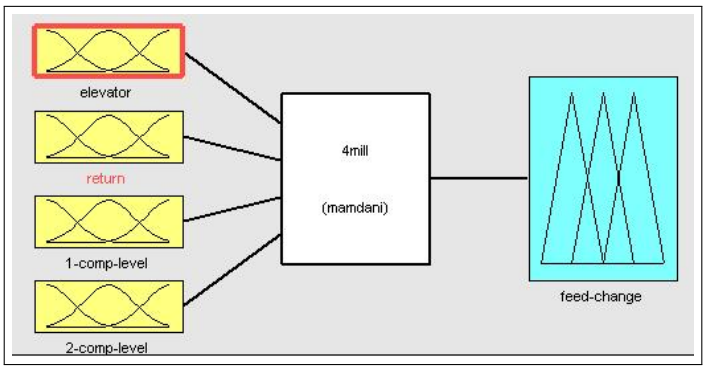


Figure 4: Fuzzy system structure using Fuzzy Editor

Each input linguistic variables has three membership functions: *low*, *ok* and *high*. Output

linguistic variable has two membership functions: *add* and *sub*. The rule basis was implemented using Fuzzy Inference System Editor from MATLAB.

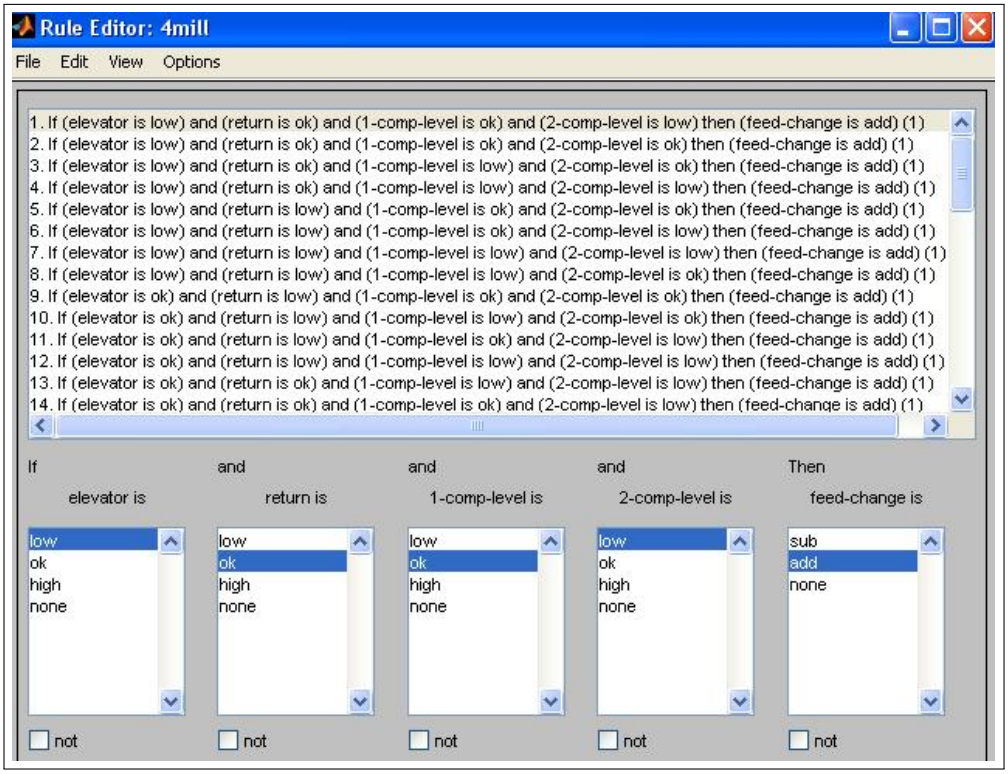


Figure 5: Rule editor

Optimal values for clinker level inside cement mill is 50% (30% are grinding media, for example steel balls and 20% are gas). Therefore, clinker level inside the mill is approximative 70% from mill capacity, excluding grinding media percentage. For coarse return a value of 20% can be considered optimal. This means that the final product is 80% and recirculated flow is 20% from mill product. The bucket elevator power is an important value because is necessary to avoid overfilling the buckets and therefore demaging the buckets [1]. Fuzzy sets have membership functions defined between 0 and 1. In this case, for a recirculated flow value of 20%, the membership function has a value of 0.2; for a clinker level value of 70%, the membership function has a value of 0.7. Figure 6 is a mesh plot of relationship between inputs and output (of fuzzy system). The plot results from a rule base and the surface is more or less bumpy.

The rule viewer depicts the fuzzy inference diagram for a Fuzzy Inference System stored in a file. The rule viewer is used to view the entire implication process from beginning to end. It is possible to move around the line indices that correspond to the inputs and then watch the system readjust and compute the new output. For the analyzed case, the rule viewer is shown in figure 7.

Figure 7 is a graphical construction of the algorithm, generated in the Fuzzy Inference System Editor from MATLAB. In Figure 7, each row refers to one rule. For example, the first row says that if the elevator power is low (row 1, column 1) and the return flow is ok (row 1, column 2) and the clinker level in the first chamber of mill is ok (row 1, column 3) and the clinker level in the second chamber of mill is low (row 1, column 4), then the output should be add (row 1, column 5).

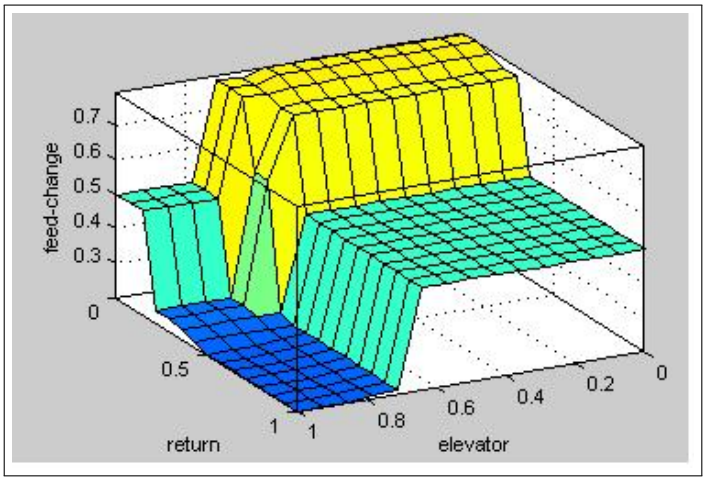


Figure 6: Control surface

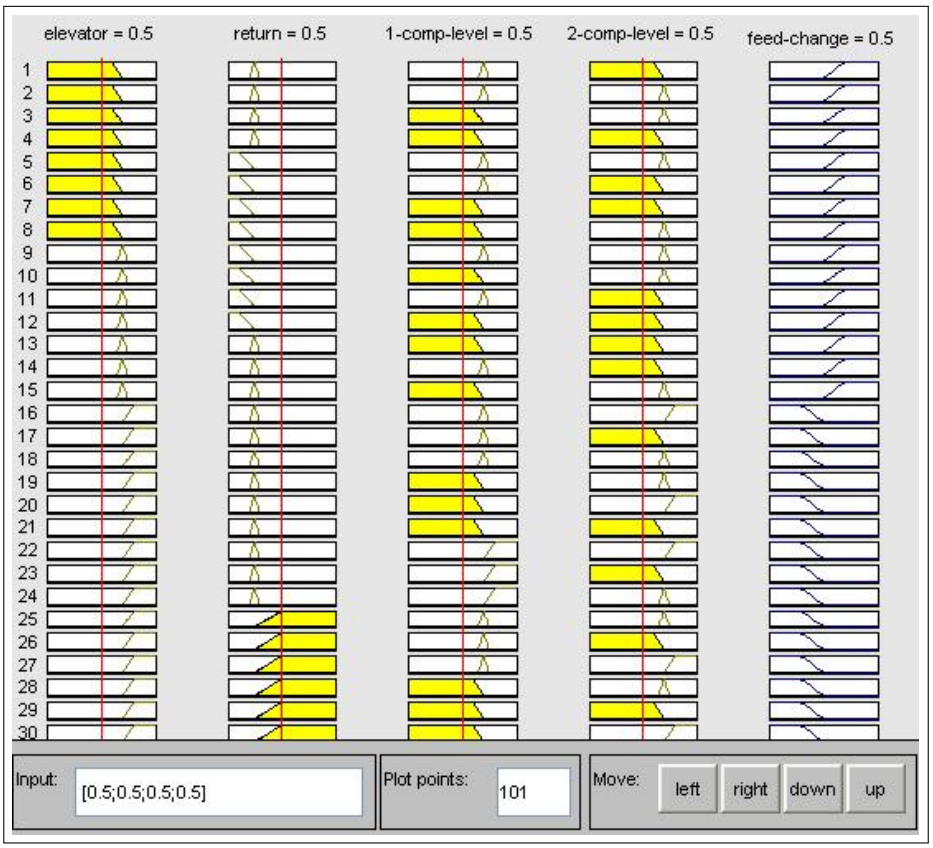


Figure 7: Graphical construction of the control signal in a fuzzy controller

3 The mill control system structure

A control system based on PLCs for clinker grinding circuit is developed. For cement mills, there are a few control loops that are considered; in this case, the control strategy is based on maintaining the total feed constant, by adjusting the fresh feed. The system has several options to enable application deployment:

- could be executed on systems from several suppliers;
- is able to work with other applications made on open systems;
- has a consistent style of interaction with the user.

The smart control system structure is illustrated in Figure 8.

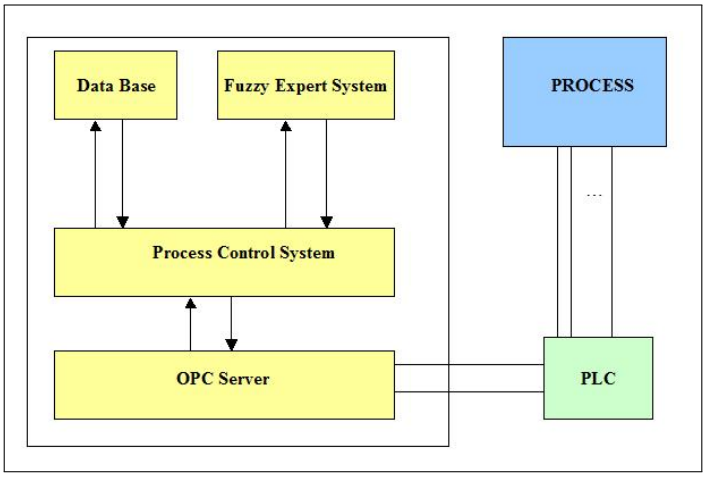


Figure 8: Smart control system structure

The process is controlled using an OPC server and this OPC server is connected to PLCs. In order to enable process monitoring, a graphical user interface was implemented by using WinCC Flexible environment for user interface development and configuration [9]. Thus the user is able to monitor the workload of the mill from the computer. The application is implemented using the SIMATIC STEP7 programming PLC [10] and SIMATIC WinCC Flexible implementation monitoring system [9].

As it is illustrated in Figure 9, the mill is load with throughput, from the feed bin. The ingredients fall down from feed bins to transporters. From the transporters, the ingredients fall down to another transporter. This transporter is feeding the mill. Material fed through the mill is grinding between the balls. From the mill outlet, the product is transported by a bucket elevator to the air separator. The air separator classify fine particles from coarse particles. The fine particles are collected as final product while the coarse particles are sent back for further grinding.

Ball mill control strategy proposal is: *total feed* = *constant*, by adjusting the fresh feed. At start, the ingredients quantities that fall down from feed bins to transporters are standard. The total feed is 100% fresh feed and 0% recirculated flow (coarse return). This fact can be seen in the Figure 9. After the materials are grinding, the particles pass out from mill and are send to the air separator. Here, the grinding particles are sorted in fine particles and coarse. From this moment, the coarse particles returns to the mill input. Because this reason, the fresh feed flow decreases (from 100% to 80%) and recirculated flow increases (from 0% to 20%). This fact can be seen in Figure 10.

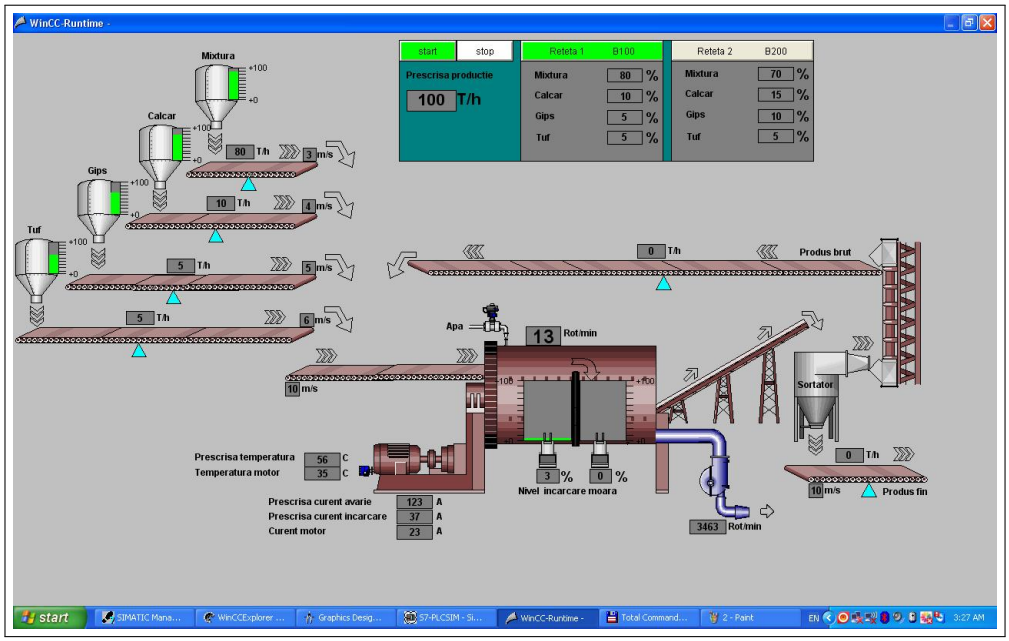


Figure 9: The process as viewed by WinCC Flexible

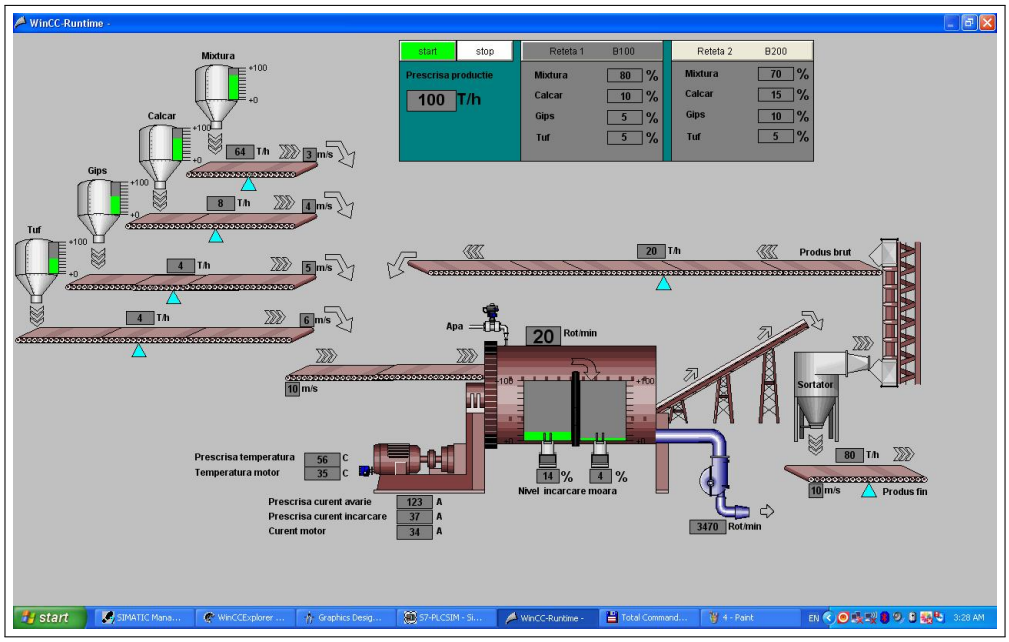


Figure 10: The process with total feed consisting of fresh feed plus recirculated flow

4 Simulation results

The dynamic behavior of a cement mill is simulated using a MATLAB-Simulink scheme. Simulation results that are presented in Figure 11 showing the value of the setpoint and the feed change. If it is used a signal generator to represent the setpoint, then the result are like in the Figure 11, where the setpoint has a square form. Figure 11 show that the feed is changing within range: 40%-60%.

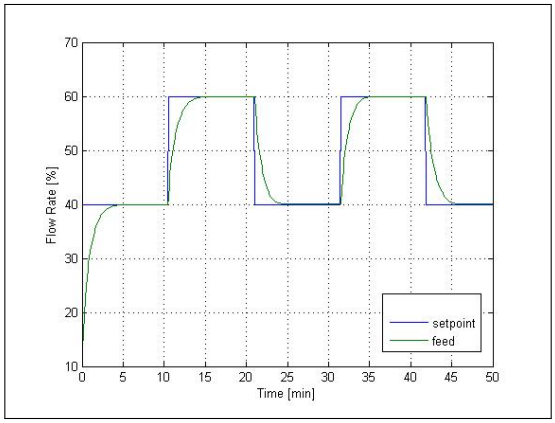


Figure 11: Feed Flow rate having a square form

But if is used a Step block instead signal generator, the results are better. Figure 12 is obtained for a setpoint that has a step signal form. The feed is constant, that mean the results will be better, because the flow rate hasn't oscillations. Figure 12 show that the feed has the optimal value (as is describe by membership function of fuzzy set) which remaind constant. Level signals simulated show a very fast reaction on material flow.

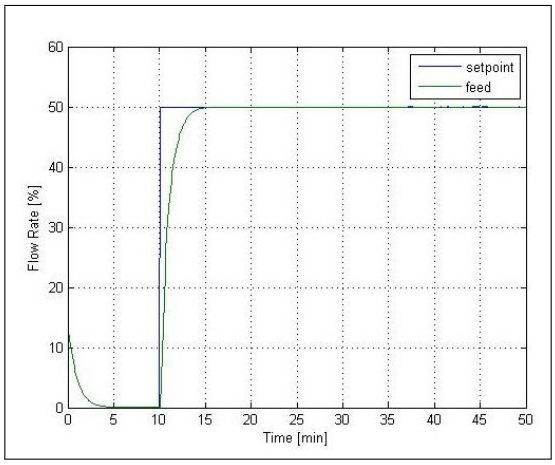


Figure 12: Feed flow rate having a step form

5 Conclusions and future works

A good control of the milling circuit contributes greatly to the stability of the process to increase grinding capacity and thus increase the amount of cement produced. This paper presents a strategy for process control in a cement miling circuit. The strategy proposed is based on

keeping the *total feed constant*, by adjusting the fresh feed. The first part of the paper describes how the structure of the fuzzy controller for a cement mill, an important part of the closed loop of the cement grinding circuit, is designed [1]. The second part presents the design of a control system by using PLC for clinker grinding circuit together with the application of a PLC program and its simulation. Automation problem presented in the paper is complex, but the proposed solution leads to the development of a modern and efficient control system.

Bibliography

[1] Costea C.R, Silaghi H., Zmaranda D., Silaghi A.M.(2014),Control of Feed Flow Rate with Fuzzy Logic for a Cement Mill, *Abstracts of ICCCC Papers, 5th International Conference on Computers, Communications and Control, ICCCC2014*, Baile Felix, Romania, ISSN 1844-4334, 4:39.

[2] Dale S., Silaghi H., Costea C.R. (2013), Procedural and Software Development of a Liapunov-based Stability Analysis Method for Interpolative-type Control Systems, *Proceedings of the 17th International Conference on System Theory, Control and Computing, ICSTCC*, 11-13 October 2013, Sinaia, Romania, ISBN 978-1-4799-2228-4:156-159.

[3] Zhenbin D., Lin T.C., Balas V.E.(2012), A New Approach to Nonlinear Tracking Control Based on Fuzzy Approximation, International Journal of Computers Communications & Control, ISSN 1841-9836, 7(1): 61-72.

[4] Costea C.R., Silaghi H., Rohde U., Silaghi A.M. (2011), Grinding Circuit Control Using Programmable Logic Controllers, Recent Advances in Signal Processing, Computational Geometry and Systems Theory, *Proceedings of the 11th WSEAS International Conference on Systems Theory and Scientific Computation*, August 23-25, 2011, Florence, Italy, 48-52.

[5] Tamir D.E., Kandel A.(2011), Axiomatic Theory of Complex Fuzzy Logic and Complex Fuzzy, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 6(3):562-576.

[6] Costea C.R., Abrudean M., Silaghi H.M., Silaghi M.A.(2010), Control of Flow Rate with Fuzzy Logic for Ball Mill, *Proceedings of 2010 IEEE International Conference on Automation, Quality and Testing, Robotics, THETA 17th edition*, May 28-30, 2010, Cluj-Napoca, 153-156.

[7] Wong L.K., Tang F.H.F., Tam P.K.S.(1998), Combination of sliding-mode controller and PI controller using fuzzy controller, *Proceedings of IEEE International Fuzzy Conference*, Anchorage, AK, 296-301.

[8] Wong L.K., Tang F.H.F., Tam P.K.S.(1997), The design of Stable Fuzzy Logic Controllers with Combination of Conventional Controllers, *Proceedings of ISIE.97*, Guimaraes, Portugal, 3: 993-997.

[9] http://www.automation.siemens.com/mcms/human-machine-interface/en/visualization-software/wincc-flexible/Pages/Default.aspx

[10] http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/Pages/Default.aspx

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):174-187, April, 2015.

Web Service Composition Framework using Petrinet and Web Service Data Cache in MANET

M. Deivamani, S.R. Murugaiyan, V. Ravisankar, P. Victer Paul,
R. Baskaran, P. Dhavachelvan

M. Deivamani*, **R. Baskaran**
Department of Computer Science and Engineering,
Anna University, Chennai, India.
m.deivamani@gmail.com, baaski@annauniv.edu
*Corresponding author:m.deivamani@gmail.com

S.R. Murugaiyan
Department of Computer Science and Engineering,
M.S.University, Tamil Nadu, India.
murugaiyansr@gmail.com

V. Ravisankar
Department of Computer Science,
Bharathiar University, Tamilnadu, India.
v_ravisankar@yahoo.com

P. Victer Paul
Department of Information Technology,
Sri Manakula Vinayagar Engineering College, Puducherry, India.
victerpaul@gmail.com

P. Dhavachelvan
Department of Computer Science,
Pondicherry University, Puducherry, India.
dhavachelvan@gmail.com

Abstract: A Mobile Ad Hoc Network (MANET) is characterized by multi-hop wireless links and frequent node mobility. Every neighboring node in the MANET is likely to have similar task and interests, several nodes might need to access the similar web service at different times. So, by caching the repeatedly accessed web service data within MANET, it is possible to reduce the cost of accessing the same service details from the UDDI and also from the external providers. *Composition of web services leads to a better alternative as, at times a candidate web service may not completely serve the need of the customer.* An effective Data Cache Mechanism (DCM) has been proposed in [6] using the Distributed Spanning Tree (DST) as a communication structure in Mobile network to improve scalability and lessen network overload. As an enhancement, Ant Colony Optimization (ACO) technique has been applied on DST to cope with the fragile nature of the MANET and to improve the network fault tolerance [1]. In these perspectives, an efficient Web Service Cache Mechanism (WSCM) can be modeled to improve the performance of the web service operations in MANET. In this paper, a fine grained theoretical model has been formulated to assess the various performance factors such as Cooperative Cache and Mobility Hand-off. In addition to these, the performance improvement of WSCM using DST and ACO optimized DST techniques in MANET has been proved experimentally using Precision and Data Reliability of the system using appropriate simulation.
Keywords: Data Cache, Web Service, Distributed Spanning Tree, Ant Colony Optimization, MANET, OMNeT++

1 Introduction

A Mobile Ad-Hoc Network (MANET) is an autonomous collection of mobile nodes that communicate over relatively bandwidth constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. The network is decentralized; where all network activity including discovering the topology and delivering messages must be executed by the nodes itself. i.e., routing functionality will be incorporated into mobile nodes [18]. The nodes in MANET would probably work for tasks of similar goal (common interest). So, most of the nodes would try to access the same web service data at different time through their corresponding Access Point (AP). The Access Points may be located at the boundaries of the MANET, where reaching them could be costly in terms of delay, power consumption, and bandwidth utilization. Moreover, the access points would be connected to a highly overloaded resource (e.g., a satellite), or an external network that is susceptible to intrusion plays a vital role in response time, security and availability of the system. For such reasons, it is recommended to cache the frequently accessed service information within the nodes in the MANET and the search application should check for the availability of the required service data within the network before requesting the external service registry [2].

Caching of service refers to the technique of caching the service method invocation information (WSDL) from the registry or service response from the corresponding service providers. The cached responses of service methods can be utilized only if the future requests use similar arguments as that of cached responses. Caching the WSDL information saves significant time and resources because subsequent service requests of similar methods will not be required to download WSDL files in a repetitive fashion. At this juncture, a service item refers to the cached WSDL information description or/and the cached web response of the corresponding service. So, the MANET applications should check for the existence of the desired service item within the network before attempting to request the external service source [2, 6]. This scenario can reduce the overload of the system for accessing external source for same service and also avoid the possible intrusion threats.

A set of proxy nodes are introduced in the MANET to provide information about the mobile nodes in the network and the services invoked by them. The proxy nodes are configured with a domain ontology and petri net modelling. The domain ontology enhance the selection of appropriate web service (by using semantics), from the service registry or peer agent nodes. The petri net modelling aims to provide composite value added service to the service requester. The service data caching within the MANET can be discussed in two methods based on the diversity of the cached service information. In most of the works, the decision to cache a service is done locally in the proxies [3, 14, 16, 17], that is, without taking into account the all the peers within the network. In such case, there may situation happens that multiple copies of the same data can be cached in the proxy nodes. This redundancy of same service data cache could reduce the possibility to cache different data that are also of interest, which affect the overall performance of the cache system. On other hand, nodes are made to decide the caching data co-operatively among the proxies [13, 15, 18], which can improve the cache diversity and also the overall data cache performance of the system. Various opportunities and challenges, like load balancing and mobility, which arise on caching web data within mobile networks, are theoretically discussed in [12].

Lan Wang [3] proposed Clustering in large-scale MANET as a means of achieving scalability through a hierarchical approach in which every node in the cluster is one hop away from every other node, that is, each cluster is a diameter-1 graph. But static cached data item manager may easily become the traffic bottleneck and single point failure of the cluster [4]. Hassan Artail et al. [2] proposed the Minimum Distance Packet Forwarding technique for search applications

within MANET that are based on the concept of selecting the nearest node from the designated nodes. The cache techniques in the studied works endure problems such as large hop count, message density and single point failure because of not following some efficient communication structure within the MANET. To cope with the problem stated, Distributed Spanning tree (DST) has been used as a communication structure in MANET for effective data cache technique proposed in our previous work [5]. DST is a recent and formally proved communication structure in MANET to lessen node isolation problem, to reduce the number of hops required to reach the nodes and makes the network scalable [7, 8, 10].

Another important performance factor in MANET is finding and maintaining routes since node mobility causes topology change which need to be observed for effective communication. In [1], Ant Colony Optimization has been used to deal with the fragile nature of the MANET which dynamically identifies the optimal path between the nodes in the DST on-demand. It is also justified that applying ACO on DST, enhance the effective routing of message (at low cost) in the MANET which in turn reduces the number of message hops required for communication to achieve excellent efficiency in DCM applications. Though, effective WSCM in MANET using DST and ACO techniques has been formally proposed in [1, 6], experimentation analyses of the work has been performed for very few performance factors such as hit ratio and message passes. Thus, in this paper, it is intended to conduct an extensive analysis on several other critical performance factors such as Cooperative Cache model, Mobility Hand-off, Precision and Data Reliability.

2 Background information needed

In this section, the discussion on innovative techniques proposed in [1, 6] which are necessary to understand the performance assessments performed in the following sections of the paper.

2.1 Distributed Spanning Tree (DST)

Distributed Spanning Tree (DST) [10] is the interconnection formation we follow as in [5, 11, 21] which, improve the routing and reduce the number of message passes required for any communication in MANET. DST systematizes MANET into a hierarchy of groups of nodes. The DST is an overlay structure designed to be scalable [11]. It supports the growth from a small number of nodes to a large one. A comprehensive algorithm for formulation of DST in MANET has been proposed and exemplified in [9]. Consequently, the MANET can be logically converted into DSTs and each DST should have its root node, named as the Head Node (HN) and the possible Leaf Nodes (LNs). Every HN will hold the complete details regarding its LNs and vice versa. These HNs are to be generated dynamically and should hold the service cache details, which is to be accessed by their corresponding LNs and indeed by other HNs also. *In addition to the cache details, domain ontology and Petrinet formalism is included in the HNs to deliver prominent service compositions.*

The DST formulated MANET can be represented as G_m in equation(1),

$$G_m = \left\{ \begin{array}{l} DST_1 = (HN_1, LN_{11}, LN_{12}, ...LN_{1(j_1-1)}) \\ DST_2 = (HN_2, LN_{21}, LN_{22}, ...LN_{2(j_2-1)}) \\ . \\ . \\ . \\ DST_i = (HN_i, LN_{i1}, LN_{i2}, ...LN_{i(j_i-1)}) \end{array} \right\} \quad (1)$$

Where,

- DST_v is the Distributed Spanning Tree and 'i' is the total number of DSTs formed in the network and $0 < v \leq i$
- HN_v is the Head Node (HN) and 'i' is the total number of HNs in the peer network equal to the number of DSTs and $0 < v \leq i$
- LN refers to the Leaf Node(s). In LN_{vz}, refers to the corresponding HN_v and $0 < z \leq j_v - 1$, where '$j_v - 1$' is the total number of LN_s in the corresponding DST.

Ant colony optimization for DST

Ant colony optimization (ACO) [19,21] is one of the most recent techniques for approximate optimization. The inspiring source of ACO algorithms are real ant colonies. More specifically, ACO is inspired by the ants' foraging behavior. By applying the ACO over the formulated DST [1], we can obtain the optimal path in terms of reduced number of message passes among the nodes in the network. ACO is also capable to reform a new optimal path in case of any problem with the current optimal path. In this paper the Ant Colony Optimization Algorithm has been modified and proposed for finding an optimal path in DST of the MANET. By optimizing every DST and connection among all the other DSTs through their HNs, it can be argued that the entire network is optimized with ACO technique for improved efficiency.

The Complexity of ACO technique depends on the method it is implemented in the MANET. In DST structure, computational complexity for ACO technique can be calculated as,

$$O(N_{DST} * N_{LN}) + O((N_{DST})^2) \tag{2}$$

Where,

- N_{DST} is the number of DSTs or the number of HNs formed in the network
- N_{LN} is the number of LN under a HN (theoretically taken same number of LNs under every HN)

2.2 Web Service Data Cache Mechanism (WSCM) with DST and ACO techniques

An efficient WSCM system in MANET with DST and ACO techniques has been formulated in [1,6] with necessary algorithms. The projected system has the capability to cope with fragile and dynamic topology changing MANET environment and the system structure can be viewed as a four layered as shown in Figure 1.

MANET Network Layer - is a network level layer consists of wireless and highly dynamic topology network.

DST Formulation Layer - is simple and converts the graph structure MP2P network into a collection DSTs. This DST structure provides the features that are necessary for a dynamic network like reduced size of routing table, minimizes routing overhead, easy network management, reduced message pass, load balancing and fault tolerance. This layer offers the dynamic node insertion into the network and exit from the network in both normal and abnormal manner. This layer also makes the system highly scalable.

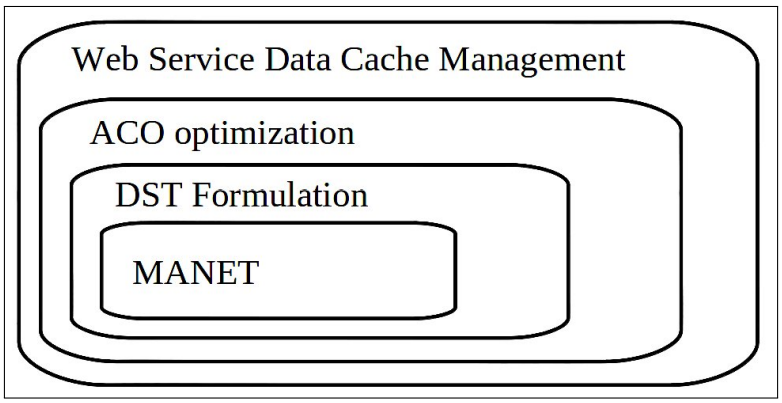


Figure 1: The layered architecture of WSCM in ACO optimized DST MANET

ACO optimization Layer - provides the system to manage with the dynamic nature of the MANET. This layer works in simple, effective and on demand way which makes the system to operate on a fragile system with asymmetric links and constantly changing topology.

Web Service Data Cache Management and Service Composition - is an application level layer with specified protocol for an effective web service information cache management system in MANET. *Using the Petrinet formalism, the web services are composed together to enhance the service quality.* Thus, using this four layer system structure, WSCM application can be efficiently performed in the highly fragile MANET environment using DST and ACO techniques.

3 Experimental analyses

In this section, an extensive analysis of DCM system using DST and ACO techniques in MANET has been performed based on critical performance factors such as Cooperative Cache model, Mobility and Hand-off, Availability, Routing technique, Cache Replacement method, Precision and Data Integrity of the system.

3.1 Simulation Setup

A MANET environment with 30 nodes and 70 service items has been simulated using OMNeT++ tool, which is an object-oriented modular discrete event network simulator. Table 1 show the partial view of HNs and its neighboring nodes, and hop distance between them. Simulation parameters are followed as similar to [1]. In the simulated network, the HNs formed are *node*05, *node*11, *node*17, *node*23 & *node*29 and other nodes act as LN to any one of these HNs. For our simulated, equation (1) can be rewritten as,

$$\left\{\begin{array}{c} DST_1 \\ DST_2 \\ DST_3 \\ DST_4 \\ DST_5 \end{array}\right\} = \left\{\begin{array}{ccccccc} 06 & 03 & 02 & 05 & 17 & \_ & \_ \\ 12 & 01 & 08 & 14 & 20 & 25 & \_ \\ 18 & 04 & 07 & 19 & 10 & 22 & 27 \\ 24 & 14 & 13 & 16 & 23 & 28 & \_ \\ 30 & 29 & 09 & 15 & 21 & 26 & \_ \end{array}\right\} \tag{3}$$

Where,

- The node number are to be preceded by the term *node*, for example the DST_1 should be interpreted are $DST_1 = (node06, node03, ..., node17)$, in which the first node $node06$ is the HN and all other nodes are LNs of DST_1.

To simulate the web service cache mechanism, we created 50 different service items and stored in the *service registry* which are accessed by the *proxy nodes* in the MANET through the access point. When a node requires any service, it will send the request to its corresponding HN (proxy node). *The HNs upon receiving the requests, extracts the keywords and match with the cache entries to identify the service has already executed from that node with in a cut-off time. The cut-off time is a time span a service information in a cache will remain valid, after which the WSDL information will be deleted from the cache (leading to cache miss for the next attempt). This deletion ensures to devour the up to date information in the cache. The HNs advertises its presents in the network for the LNs to identify and request for a service.*

S.No	HN	Nearest HNs and distance in Hop(s)			
		HN 1	Hop Distance from HN	HN 2	Hop Distance from HN
1	node06	node18	2	node30	4
2	node12	node06	4	node17	2
3	node18	node30	3	node12	2
4	node24	node17	4	node18	3
5	node30	node12	2	node06	5

Table 1: HNs formulated and its hop distance from the nearest two HNs in the simulated MANET

During the very first time access of any service, the LN saves the copy of the accessed service item and intimates its corresponding HN to save the type of the service item and the details of LN which holds it. When any node request for the web service of similar nature, the requesting node is served with the service item by the LN which holds the cache, identified through its HN. *A keyword extracted from the request may or may not match with the cache entries. If it matches, the WSDL information is provided to the corresponding requesting LN using standard message passing. If the keyword does not match with the cache table, the domain ontology is used at the first level to find related services (rather than exact keyword matching) from the cache. At the next stage, if related services also give a cache miss, the peer HNs or the external service registry is contacted for availing the service configurations.* Upon reaching all the HNs, it can be found that the service does not exist within the network and obtained from the external UDDI/service providers.

A service request initiated by a LN can be fulfilled by an atomic service or a composite service. At times, atomic services may not be available for a given service request, where several compatible services are identified, composed and executed. To identify the compatible services for composing a value added service, petrinet formalism is used in the MNs. Petri Net is used to mainly identify the reachability of the compatible web services within the domain.

Definition 1: The Best and Worst case analysis for accessing the cached service item based on the message hops can be modeled as follows.

Best Case: The total number of message hops required to access the cached data item is minimum, when service item is being cached in any LN which is under the HN of the requesting node.

This can be expressed as,

$$n(access_{messagehops}) + n(dictionary_{search}) \leq 4N \tag{4}$$

where,

- $n(access_{messagehops})$ is the total number of message hops required to access the cached service item.
- $n(dictionary_{search})$ is the total time to search the semantic of the keyword in the search.
- N is the total number of message hops between HN and its LN (consider equal for every LNs).

Worst Case: The total number of message hops required to access the cached service item is maximum, when the HN of the node that holds the requested service item is at distance 'k' from the HN of requester node, where 'k' is the total number of HNs in the MANET. This can be expressed as,

$$n(access_{messagehops}) + n(dictionary_{search}) \geq (kXM) + 4N \tag{5}$$

where,

- $n(access_{messagehops})$ is the total number of message passes required to access the cached item
- $n(dictionary_{search})$ is the total time to search the semantic of the keyword in the search
- M is the total number of message pass between two HNs (consider equal for between every HNs).
- N is the total number of message pass between HN and its LN (consider equal for every LNs).

An extensive analyses to model the various performance factors such as Cooperative Web Service Cache model, Mobility and Hand-off, Availability, Routing technique, Cache Replacement method, Precision and Consistency for the WSCM in DST and ACO optimized DST MANET have been performed in the following sections.

3.2 Cooperative cache model

To improve the service information accessibility, mobile nodes should cache different service item that of their neighbor nodes [1]. Every LN should cooperatively cache the different service to avoid the replicated caching of same service within the network. Caching same service on different LNs may reduce the access delay but on considering size of the cache memory in LNs it will block caching more frequently accessing services (different).

Definition 2: Let LN_i is the mobile node which cache the service item. To show that each LN caches different service,

$$LN_1(S_{id}) \cap LN_2(S_{id}) \cap LN_3(S_{id}) .... \cap LN_k(S_{id}) = \phi \tag{6}$$

$$LN_m(S_{id1}) \cap LN_m(S_{id2}) \cap LN_m(S_{id3}) .... \cap LN_m(S_{idn}) = \phi \tag{7}$$

where,

- k be the number of LN which cache data item in MANET.
- n be the number of service items cached in m^{th} LN and $0 < m < k$.
- S_{id} is web service index entry in LN_TABLE of a LN.

Eq. 6 refers that no same service is cached by the different LNs and Eq. 7 refers to no same service is within a LN. Thus, there is no repetition in the web service index entry in LN_TABLE of every LN. So, every service is cached only once in the MANET.

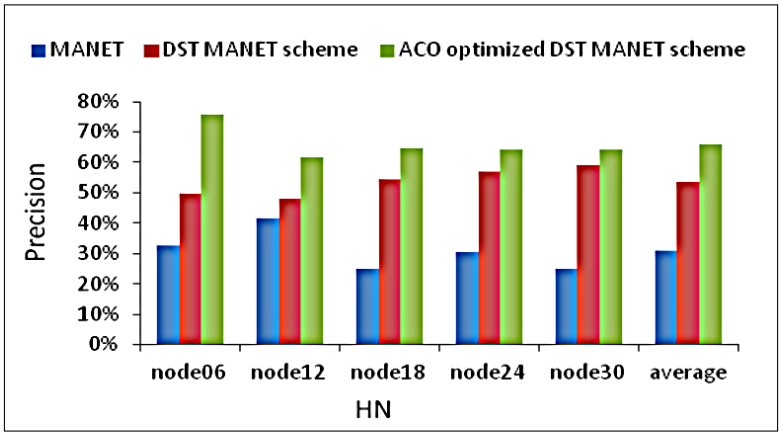


Figure 2: Comparison on Precision performance of three different schemes

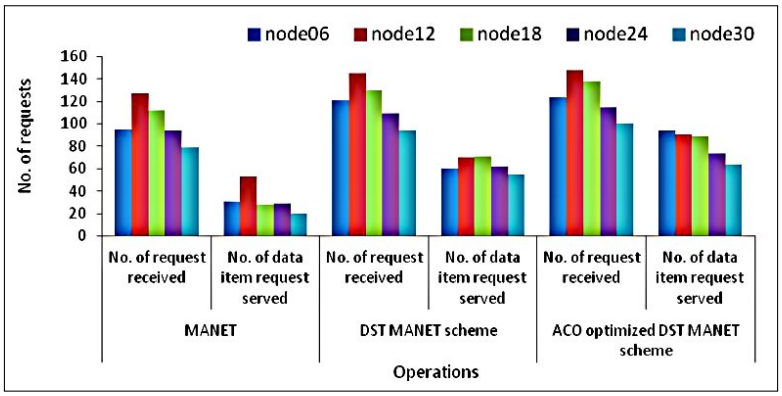


Figure 3: Comparison on service request received and served using three different schemes

3.3 Mobility and hand-off model

Due to the fragile nature, mobile nodes are set to move free in MANET and node tracking task added complex. This tracking task can be accomplished in two scenarios, exit or remove from the MANET [5] and switching among the DSTs within the MANET.

Scenario 1: A node can exit/remove from the network dynamically in the following fashion: The node that wants to get remove from the network should send an inform message to its HN, so that the corresponding HN removes the node details and its cached web service identity from its LNs list, *which obviously removes from the spanning tree.*

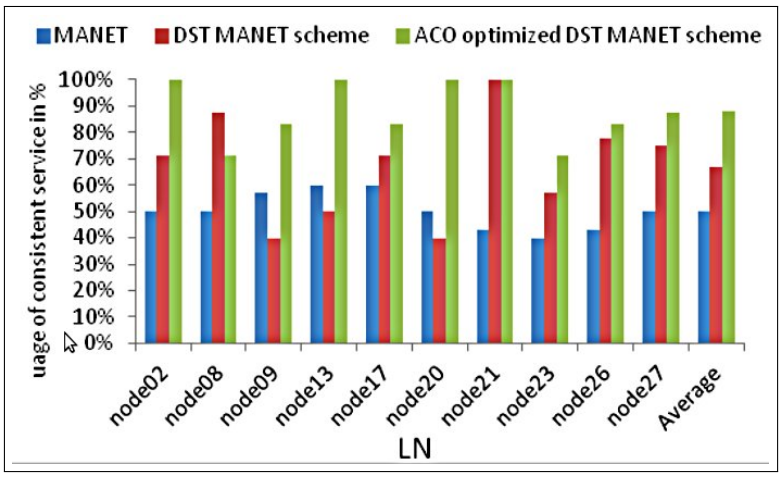


Figure 4: Comparison on % of utilization of consistent service item performance of three different schemes

Scenario 2: Switching of node from one DST to another is configured automatically by the HN by passing some messages with the node. Any LN can voluntarily hand-off itself from any HN take HN_1, if the condition satisfies that number of hops between LN and HN_1 exceeds the number of LNs under HN_2. And the LN can join to HN_2 in which number of hops between LN and HN_2 will be less than number of LNs under HN_2. This hand-off should be intimated to both HN_1 and HN_2. *It is must that every LN should be under any HN. If a request arises in the mean time between handover, the HN_1 will transmit a "Binding Warning" and intimate LN is now under HN2.*

S.No	LN	MANET				MANET with DST scheme				MANET with ACO optimized DST scheme			
		No. of data items cached	No.of data item request served	No. of request received	Precision (%)	No. of data items cached	No. of data item request served	No. of request received	Precision (%)	No. of data items cached	No. of data item request served	No. of request received	Precision (%)
1	node06	11	31	95	32.62	11	60	121	49.68	15	94	124	75.63
2	node12	13	53	127	41.77	20	70	145	47.98	11	91	148	61.80
3	node18	10	28	112	24.81	15	71	130	54.52	15	89	138	64.61
4	node24	17	29	94	30.62	12	62	109	57.13	14	74	115	64.39
5	node30	10	20	79	24.78	12	55	94	58.94	15	64	100	64.20
Aggregate Performance		60	160	507	30.9	70	318	599	53.7	70	413	625	66.1

Table 2: Comparison on precision for HNs involved in serving the nodes in all three different scenarios

Definition 3: Let v_i be a mobile node in MANET which is under HN_i and HN_j be another HN in the MANET. Then, v_i can decide to hand-off from HN_i and to join under HN_j, if it satisfies the rule of Eq. 9.
If,

$$HN_i, HN_j \in \{HN_1, HN_2, HN_3, ...HN_n\} \tag{8}$$

then,

$$nhop(v_i, HN_i) > n(LN_i) \;\&\; \&nhop(v_i, HN_j) > n(LN_i) \tag{9}$$

where,

- n be the total number of HNs in MANET.
- $nhops(v, HN)$ be the number of hops required to reach node v from HN.
- $n(LN_i)$ be the total number of LNs under the Head Node HN_i.

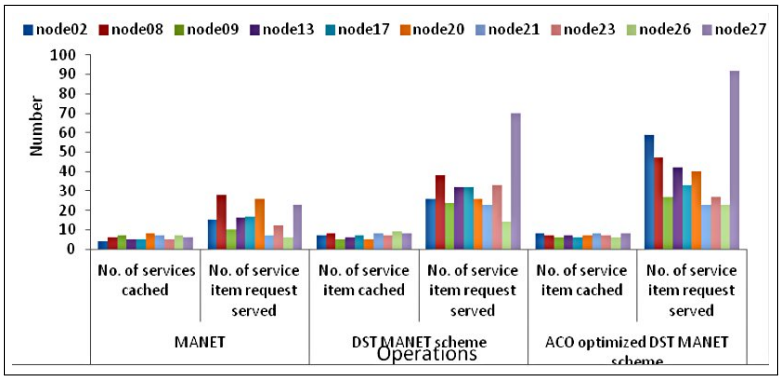


Figure 5. Comparison on data item cached and served using three different schemes.

Thus, any LN whose distance from existing HN is lesser than another HN can perform handoff from existing HN and join with new one. *The Mobility and Handoff model works under the assumption that the HNs will always be in access range within the MANET. This assumption is made as LNs under the MN should not be disconnected from the spanning tree and thereby to the network.*

3.4 Precision

The precision refers to the ratio of total number of service request received to the total number of requested service found in the MANET. The different performance data observed from the simulation in first 100 seconds are tabulated in Table 2; which contain HNs created, No. of services cached by LNs of each HN and No. of service item requests served by each HN, *either at an atomic service or a composition. Composition of web services is carried using the Petri net modelling by the following phases – identifying the similar web services using domain ontology, classification as compatible and non-compatible, execution of compatible web services pertaining to the goal of the service request.* To explain in clear manner, consider the first entry in Table 2, the mobile node, node06 is an HN and the total number of data items cached in its LNs, the total number of data item request served using the cache and the total number of request received for service item are 11, 31 and 95 respectively. Thus Precision for HN node06 is 32.62%.

Table 2 shows that precision percentage of HNs in the MANET can be improved using DST and ACO techniques. The maximum precision value recorded for MANET, MANET with DST scheme and MANET with ACO optimized DST scheme is 41.7%, 58.94 and 75.63% respectively which is illustrated in Figure 2. This is because the DST and ACO techniques reduce the number of message hops required for any operation, such as cache request and cache reply, by discovering an optimal path between the nodes in the MANET. So operations performed faster and more requests are served from nodes local cache within the stipulated time period.

Figure 3 illustrates the comparison on service request received and served using three different schemes. From Figures 3 and 4, it can be observed that in ACO optimized DST scheme outperforms other two schemes. Thus, precision performance of the system can be improved from DST scheme from 30.9% to 53.7% and which is further improved to 66.1% using ACO optimization technique.

S.No	LN	MANET				MANET with DST scheme				MANET with ACO optimized DST scheme			
		No. of service items cached	No.of service item request served	No. of service items deleted (expired) before use	Utilization % of consistent service item	No.of service items cached	No. of service item request served	No.of service items deleted (expired) before use	Utilization % of consistent service item	No. of Service items cached	No. of service item request served	No.of service items deleted (expired) before use	Utilization % of consistent service item
1	node02	4	15	2	50.0	7	26	2	71.4	8	59	0	100.0
2	node08	6	28	3	50.0	8	28	1	87.5	7	47	2	71.4
3	node09	7	10	3	57.1	5	24	3	40.0	6	27	1	83.3
4	node13	5	16	2	60.0	6	32	3	50.0	7	42	0	100.0
5	node18	5	17	2	60.0	7	32	2	71.4	6	33	1	83.3
6	node20	8	26	4	50.9	5	26	3	40.0	7	40	0	100.0
7	node21	7	7	4	42.9	8	23	0	100.0	8	23	0	100.0
8	node24	5	12	3	40.0	7	33	3	57.1	7	27	2	71.4
9	node26	7	6	4	42.9	9	14	2	77.8	6	23	1	83.3
10	node27	6	23	3	50.0	8	70	2	75.0	8	92	1	87.5
Aggregate Performance		60	160	30	50.3	70	318	21	67.0	70	413	8	88.0

Table 3: Comparison on % of utilization of consistent service item by LNs in all three different scenarios

3.5 Service reliability

Service reliability or consistency refers to the correctness of the cached service at the time of access within the MANET. Though consistency technique followed is Time-Based it is not required that every mobile should be synchronized in clock. Every HN which stores the service item type also store metadata about the service item S_i which contain the time at which S_i is being cached. This information is used to check the service item validity. 'T ' is the constant time value in the MANET which can be varied based on service item being updated in outside network. The factor used to measure the consistency of the system is percentage of consistent service item usage.

Table 3 shows the comparison on % of utilization of consistent service item by LNs in all three different scenarios for first 100 seconds of simulation run. From these statistics, it can be observed that utilization percentage of consistent service item is much improved DST MANET and ACO optimized DST MANET schemes. The same is illustrated in the Figure 4 and 5, which confirms the improved performance of the DST and ACO optimized DST schemes over MANET scheme.

Table 4 shows the utilization increases more using petrinets and deliver compositions. As per the assumption, the mobile agents tender similar service request, a service composition held in the cache will be a suitable candidate for most of the request, rather being a single atomic service. Without loss of generality, we can say that the service served ratio to the atomic service with respect to the composite service will be less. And it is trivial that if an atomic service can satisfy a customer request, a composite service (which includes compatible atomic services) will also satisfy the request in a more efficiently.

3.6 Discussion

To Summarize, the DST structure offers the capabilities that are necessary for a dynamic network like reduced size of routing table, minimizes routing overhead, easy network management, reduced message hops, load balancing and fault tolerance. ACO optimized DST structure provides the system to manage with the highly fragile nature of the MANET and to find the optimal route between HNs and HN & its LNs on demand fashion. This layered approach works in simple, effective and on demand way which makes the system to operate on a fragile environment with asymmetric links and constantly changing topology. Thus, the performance of service cache technique for *Web Service Composition* in MANET has been analysed for Cooperative Cache, Mobility Hand-off, Precision and Data Reliability method. An extensive experimenta-

S.No	LN	MANET with DST and ACO schemes				MANET with DST , ACO and Petri Net			
		No. of service items cached	No.of service item request served	No. of service items deleted (expired) before use	Utilization % of consistent service item	No.of service items cached	No. of service item request served	No.of service items deleted (expired) before use	Utilization % of consistent service item
1	node02	8	59	0	100.0	8	63	0	100.0
2	node08	7	47	2	71.4	7	58	1	85.7
3	node09	6	27	1	83.3	6	41	0	100.0
4	node13	7	42	0	100.0	7	48	1	85.7
5	node18	6	33	1	83.3	6	33	0	100.0
6	node20	7	40	0	100.0	7	42	0	100.0
7	node21	8	23	0	100.0	8	29	0	100.0
8	node24	7	27	2	71.4	7	35	2	71.4
9	node26	6	23	1	83.3	6	27	1	83.3
10	node27	8	92	1	87.5	8	95	0	100.0
Aggregate Performance		70	413	8	88.0	70	471	5	92.6

Table 4: Comparison on % of utilization of consistent service item Vs a composition by LNs

tion has been performed on precision and service reliability of the system which confirms that the ACO optimized DST scheme improves the efficiency of service cache technique in MANET environment.

4 Conclusion

The work presented in this paper described modeling and assessing the various performance factors for the Service Cache Mechanism *for composing web services* using DST and ACO techniques in MANET proposed in our previous works. A comprehensive theoretical model has been developed for the performance factors such Cooperative Cache, Mobility Hand-off, Precision and Data Reliability methods. In addition to these, the performance improvement of Web Service Cache Mechanism (WSCM) has been proved experimentally for improved based on Precision and Service Integrity factors using three different schemes such as MANET, DST MANET and ACO optimized DST MANET. The simulation results shows that the precision performance of the WSCM system is improved using the DST scheme from 30.6% to 53.6% and which is further improved to 68.1% using ACO optimization scheme. And the service reliability performance is enhanced from 50.3% to 67.0% and to 88.0% using DST and ACO optimization schemes respectively.

Acknowledgements

This work is a part of the Research Project sponsored under the Major Project Scheme, UGC, India, Reference No: F. 41-639/July 2012 (SR). The authors would like to express their thanks for the financial support offered by the Sponsored Agency.

Bibliography

[1] R. Baskaran, P. Victer Paul and P. Dhavachelvan (2012), Ant Colony Optimization for Data Cache Technique in MANET, *Int. Conf. on Advances in Computing (ICADC 2012), Advances in Intelligent and Soft Computing series, India,*, Springer, 873-878.

[2] Hassan Artail and Khaleel Mershad (2009); MDPF: Minimum Distance Packet Forwarding for search applications in mobile ad hoc networks , *IEEE Transactions on Mobile Computing*, 8(10): 1412 - 142.

[3] Lan Wang and Stephan Olariu (2005); Cluster Maintenance in Mobile Ad-hoc Networks, *Springer Science + Business Media*, 8: 111-118.

[4] P. Krishna, M. Chatterjee, N. Vaidya and D. Pradhan(1997); A cluster-based approach for routing in ad-hoc networks, *ACM SIGCOMM Computer Communication*, 27(2): 49-64.

[5] P. Victer Paul, N. Saravanan, S.K.V. Jayakumar, P. Dhavachelvan, R. Baskaran (2008); QoS Enhancements for Global Replication Management in Peer to Peer networks, *Future Generation Computer Systems*, Elsevier, 28(3):573-582.

[6] P. Victer Paul, D. Rajaguru, N. Saravanan, R. Baskaran and P. Dhavachelvan (2013); Efficient service cache management in mobile P2P networks, *Future Generation Computer Systems*, Elsevier, 29(6): 1505-1521.

[7] Sylvain Dahan (2005); Distributed Spanning Tree Algorithms for Large Scale Traversals, *11th International Conference on Parallel and Distributed Systems (ICPADS'05)*, DOI: 10.1109/ICPADS.2005.131 , 1: 453-459.

[8] P. Victer Paul, T. Vengattaraman, P. Dhavachelvan (2010); Improving efficiency of Peer Network Applications by formulating Distributed Spanning Tree, *Third International Conference on Emerging Trends in Engineering & Technology (ICETET-2010)*, IEEE, India, 813-818.

[9] R. Baskaran, P. Victer Paul and P. Dhavachelvan (2012); Algorithm and Direction for Analysis of Global Replica Management in P2P Network, *IEEE International Conference on Recent Trends in Information Technology (ICRTIT)*, May 2012, Chennai, 211 - 216.

[10] Sylvain Dahan, Jean-Marc Nicod and Laurent Philippe (2005); The Distributed Spanning Tree: A Scalable Interconnection Topology for Efficient and Equitable Traversal, *International Symposium on Cluster Computing and the Grid*, 2005 IEEE.

[11] Sylvain Dahan (2005); Distributed Spanning Tree Algorithms for Large Scale Traversals, *11th International Conference on Parallel and Distributed Systems (ICPADS'05)*, DOI: 10.1109/CCGRID.2005.1558561, 1: 243-250.

[12] R. Friedman (2002); Caching Web Services in Mobile Ad-Hoc Networks: Opportunities and Challenges, *Proc. Second ACM Intl Workshop Principles of Mobile Computing*, 90-96.

[13] J. Zhao, P. Zhang, G. Cao, and C. Das (2010); Cooperative caching in wireless p2p networks: Design, implementation, and evaluation, *Parallel and Distributed Systems, IEEE Transactions on*, 21(2):229-241.

[14] S. Lim, W. Lee, G. Cao, and C. Das (2006); A Novel Caching Scheme for Internet Based Mobile Ad Hoc Networks Performance, *Ad Hoc Networks*, 4(2):225-239.

[15] N. Chand, R. C. Joshi, and M. Misra (2006); Efficient cooperative caching in ad hoc networks. COMSWARE, DOI:10.1109/COMSWA.2006.1665190, 1-8.

[16] Fan Ye, Qing Li, and EnHong Chen, Adaptive caching with heterogeneous devices in mobile peer to peer network, *ACM Symposium on Applied Computing (SAC '08)*. ACM, New York, USA, 1897-1901.

[17] Guohong Cao, Liangzhong Yin, Chita R. Das (2004); Cooperative Cache- Based Data Access in Ad Hoc Networks, Pennsylvania State University, *IEEE Computer Society*, February 2004, 32-39.

[18] Hassan Artail and Khaleel Mershad (2009); MDPF: Minimum Distance Packet Forwarding for search applications in mobile ad hoc networks, *IEEE Transactions on Mobile Computing*, 8(10): 1412 - 142.

[19] Dorigo, M., Maniezzo, V., Colorni A. (1991); *The ant system: An autocatalytic optimizing Process. Tech.Rep. 91-016 Revised*, Politecnico di Milano, Italy.

[20] A. Colorni, M. Dorigo, V. Maniezzo (1991); Distributed optimization by ant colonies, *Proc. of ECAL91 European Conference on Artificial Life, Elsevier Publishing*, Amsterdam, The Netherlands, 134-142.

[21] P. Victer Paul, T. Vengattaraman, P. Dhavachelvan and R. Baskaran (2010); Improved Data Cache Scheme using Distributed Spanning Tree in Mobile Adhoc Network, *The International Journal of Computer Science and Communication (IJCSC)*, 1(2): 329-332.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):188-199, April, 2015.

Zero-watermarking Algorithm for Medical Volume Data Based on Difference Hashing

B.R. Han, J.B. Li, Y.J. Li

Baoru Han, Jingbing Li*, Yujia Li
College of Information Science and Technology
Hainan University
Haikou, 570228, China
Email:6183191@163.com, Jingbingli2008@hotmail.com, liyujia1219@gmail.com
*Corresponding author: Jingbingli2008@hotmail.com

Abstract: In order to protect the copyright of medical volume data, a new zero-watermarking algorithm for medical volume data is presented based on Legendre chaotic neural network and difference hashing in three-dimensional discrete cosine transform domain. It organically combines the Legendre chaotic neural network, three-dimensional discrete cosine transform and difference hashing, and becomes a kind of robust zero-watermarking algorithm. Firstly, a new kind of Legendre chaotic neural network is used to generate chaotic sequences, which causes the original watermarking image scrambling. Secondly, it uses three-dimensional discrete cosine transform to the original medical volume data, and the perception of the low frequency coefficient invariance in the three-dimensional discrete cosine transform domain is utilized to extract the first 4*5*4 coefficient in order to form characteristic matrix (16*5). Then, the difference hashing algorithm is used to extract a robust perceptual hashing value which is a binary sequence, with the length being 64-bit. Finally, the hashing value serves as the image features to construct the robust zero-watermarking. The results show that the algorithm can resist the attack, with good robustness and high security.
Keywords: zero-watermarking, medical volume data, difference hashing, Legendre chaotic neural network, three-dimensional discrete cosine transform.

1 Introduction

With the extensive application of digital technology, a large number of digital images are used in our daily life and work. Digital images meet the requirements of people's senses, and also provide convenience for people's life and work. People pay more and more attention to the copyright issues in digital image. Digital watermarking, as a new security measure, is widely used in digital image copyright protection [1, 2]. As an important branch of information security research area, digital watermarking also is an effective way to protect the integrity of digital image [3, 4]. It is an effective complement to traditional encryption techniques. Digital watermarking is the meaningful information hidden into the digital works, as a basis for the identification of copyright. At the same time, the watermarking information can be detected and analyzed by detecting algorithm, to determine the copyright holder and to protect the digital products. However, due to the digital watermarking technology research being multidisciplinary, the communication theory, computer science and signal processing, and many other areas involved, for which it is unable to avoid the inherent drawbacks in these fields [5, 6]. This brings certain difficulty and challenge to research work. At present, in the field of digital watermarking, the digital image watermarking has become a branch of the most widely used, the most mature development, and the most fruitful achievements [7].

Along with the construction of hospital informatization, digitalization has become more and more deeply into the medical field [8]. Medical diagnostic equipment will produce a lot of medical image information every day. These medical images as a basis for medical diagnosis

have an extremely important position. The establishment of medical digital image transmission standard has promoted the exchange of the digital medical imaging information. However, medical images information during network transmission may also encounter tampering, illegal copying and other information security issues [9,10]. Medical image is not only an important basis for doctors to diagnose the disease, but also involving the patient's privacy. In order to ensure the medical image security, reliability and availability, we must effectively solve the security problem of medical image management. The emergence of medical image digital watermarking can solve this problem [11]. It is the specific meaning digital information or some secret information embedded into the digital medical image, which can realize the information hiding and copyright protection. Medical image especially is strict to the requirement of image quality does not allow the distortion and not allowed to do any changes. In this case, people put forward the concept of zero-watermarking [12]. Zero-watermarking is a novel digital watermarking technique. It is different from general digital watermarking in its use of the characteristics of the original image to construct the watermarking, and not directly embed watermarking in the image, so it does not break the original image [13].

Image hashing is also called the digital fingerprint, it is a summary of multimedia information, which can be widely applied in image authentication, image indexing and retrieval, digital image watermarking, etc [14]. Image hashing represents the image itself with a short digital sequence, which is a kind of expression of image compression based on visual content. Usually, the image hashing should satisfy the requirements of perceptual robustness, uniqueness and security. Image hashing technique can be any image-resolution image data into a binary sequence of hundreds or thousands of bits. For a large database of image retrieval, this means greatly reducing the search time, but also reduces the cost of storage media images. At the same time, its robustness feature ensure that it can resist a variety of different types of attacks. In addition, the characteristics of image hashing technology security make the copyright protection of image become possible. The main image hashing methods now mainly focus on the characteristics of robustness study [15,16]. It is mainly divided into the method based on image statistics, a rough image representation, the relationship and visual feature points extraction. Combined the features of above mentioned extraction methods, in order to better satisfy the perceptual image hashing robustness, security, and uniqueness, the discrete cosine transform to extract the feature is considered to be an ideal method. According to the characteristics of medical volume data, this paper presents a zero-watermarking algorithm for medical volume data based on Legendre chaotic neural networks and differences hashing. The algorithm is based on three-dimensional discrete cosine transforms perception of the low frequency coefficient invariance and image hashing robust feature, which is a robust zero-watermarking method. The algorithm uses the robust hashing sequence in medical volume data transform domain to construct the zero-watermarking, instead of modifying the features of medical volume data. It can adapt to the characteristics of medical volume data, can resist strong attack, and has very strong robustness. And it uses Legendre chaotic neural network scrambling and encryption, which has good security and confidentiality.

2 Legendre chaotic neural network

The paper uses a new Legendre chaotic neural network. The Legendre chaotic neural network model is shown in figure 1. The Legendre chaotic neural network selects Legendre polynomials as the activation function of hidden layer. Performance close to the theoretical values of the chaotic sequence is generated by the Legendre chaotic neural network weights and the chaos initial value. The chaotic sequence is used for scrambling.

A polynomial defined by the following formula is called the Legendre polynomial.

Definition 1.

$$P_0(x) = 1, P_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}(x^2-1)^n, \quad n = 1, 2, 3, \cdots \tag{1}$$

$P_n(x)$ is known as Legendre polynomials. It is known as the weight function n orthogonal polynomials in space $[-1, 1]$.

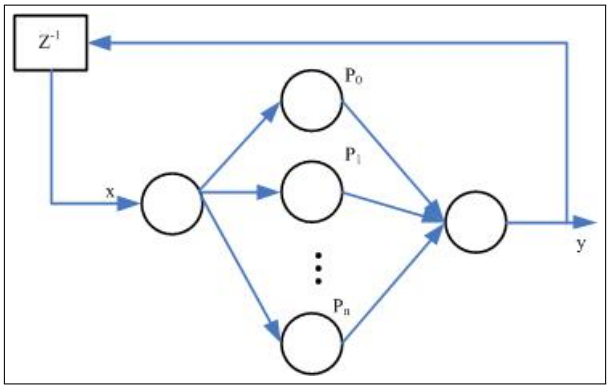


Figure 1: Legendre chaotic neural network

Set the input layer to the hidden layer weight is w_j , hidden layer to the output layer weight is c_j.The activation function of hidden layer neuron is Legendre orthogonal polynomials. The hidden layer neuron input is

$$net_j = w_j x \quad j = 0, 1, 2, \ldots, n \tag{2}$$

Hidden layer neurons output are as a set of legendre orthogonal polynomial terms $P_j(net_j), j = 0, 1, 2, \ldots, n$, which can be obtained by formula (1) recursive. Legendre chaotic neural network output is

$$y = \sum_{j=0}^{n} c_j P_j(net_j) \tag{3}$$

Set the training sample is $(T_t, D_t), t = 1, 2, \cdots, l$. Where l is the number of samples $T_t = (x_{1t}, x_{2t}, \cdots, x_{mt})$ is legendre chaotic neural network input. d_t is Legendre chaotic neural network desired output. The network is trained using BP learning algorithm.

Error formula is as follows:

$$e_t = d_t - y_t \tag{4}$$

$$E = \frac{1}{2}\sum_{t=1}^{l} e_t^2 \tag{5}$$

Network weights adjustment formula is as follows.

$$\Delta c_j = -\eta\frac{\partial E}{\partial c_j} = \eta e_t P_j(net_j) \tag{6}$$

$$\Delta w_j = -\eta\frac{\partial E}{\partial w_j} = \eta e_t c_j P_j'(net_j) x_j \tag{7}$$

$$\begin{cases} w_j(k+1) = w_j(k) + \Delta w_j(k) \\ c_j(k+1) = c_j(k) + \Delta c_j(k) \end{cases} \tag{8}$$

Where k is the training epochs $t = 1, 2, \cdots, l$; $j = 0, 1, 2, \cdots, n$.

3 Three-dimensional discrete cosine transform

Three-dimensional discrete cosine transform formula is as follows.

$$F(u,v,w) = c(u)c(v)c(w)\left[\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\sum_{z=0}^{P-1} f(x,y,z) * \cos\frac{(2x+1)u\pi}{2M} \cos\frac{(2y+1)v\pi}{2N}\cos\frac{(2z+1)w\pi}{2P}\right] \quad (9)$$

$u = 0, 1, \cdots, M-1; v = 0, 1, \cdots, N-1; w = 0, 1, \cdots, P-1$

In the formula,

$$c(u) = \begin{cases} \sqrt{1/M} & u = 0 \\ \sqrt{2/M} & u = 1, 2, \ldots, M-1 \end{cases} \quad (10)$$

$$c(u) = \begin{cases} \sqrt{1/N} & v = 0 \\ \sqrt{2/N} & v = 1, 2, \ldots, N-1 \end{cases} \quad (11)$$

$$c(u) = \begin{cases} \sqrt{1/P} & w = 0 \\ \sqrt{2/P} & w = 1, 2, \ldots, P-1 \end{cases} \quad (12)$$

Where $f(x,y,z)$ is volume data of the data values in the (x,y,z). $F(u,v,w)$ is the data corresponding to the three-dimensional discrete cosine transform coefficients. Three-dimensional inverse discrete cosine inverse transform formula is as follows:

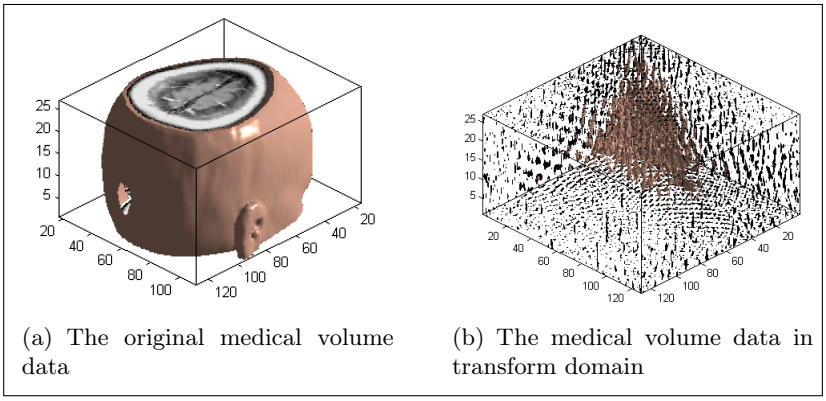
$$f(x,y,z) = \left[\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\sum_{z=0}^{P-1} F(u,v,w) * \cos\frac{(2x+1)u\pi}{2M} \cos\frac{(2y+1)v\pi}{2N}\cos\frac{(2z+1)w\pi}{2P}\right] \quad (13)$$

$u = 0, 1, \cdots, M-1; v = 0, 1, \cdots, N-1; w = 0, 1, \cdots, P-1$

Three-dimensional discrete cosine transform for medical volume data is shown in figure 2.

4 Difference hashing

Perceptual hashing has become a hot research topic in the field of multimedia signal processing and multimedia security. Perceptual feature extraction is the core part of the perceptual hashing structure. The validity and reliability of perceptual feature extraction will directly affect the robustness and uniqueness of image perception hashing sequence. Image perceptual hashing study is mainly for image authentication and image retrieval. Image features include image color, texture, edge, corner and image transform domain coefficient, etc. Compared with perceptual hashing, difference hashing in speed is much faster. Compared with the average hashing, the effect of the difference hashing is better in the case of almost the same efficiency. It is based on the gradual implementation. Based on the difference hashing algorithm, a difference hashing algorithm in three-dimensional discrete cosine transform domain is presents for medical volume



(a) The original medical volume data

(b) The medical volume data in transform domain

Figure 2: Three-dimensional discrete cosine transform for medical volume data

data feature extraction in this paper. Figure 3 depicts the algorithm flow. In the algorithm flow, medical volume data is represented by a three-dimensional map into a one-dimensional feature vector. The algorithm is used to extract the robust features of medical volume data, which can increase the robustness of watermarking algorithm.

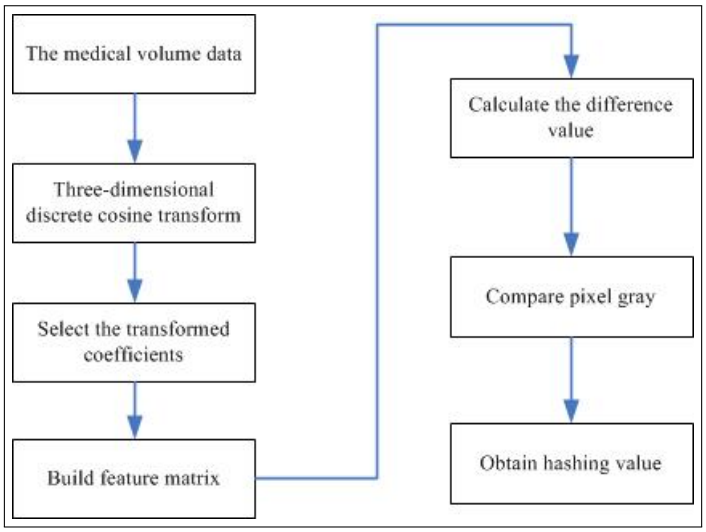


Figure 3: The difference hashing algorithm flow

5 Zero-watermarking algorithm

Zero-watermarking embedding

Figure 4 illustrates zero-watermarking embedding process.

Zero-watermarking extraction

Figure 5 illustrates zero-watermarking extraction process.

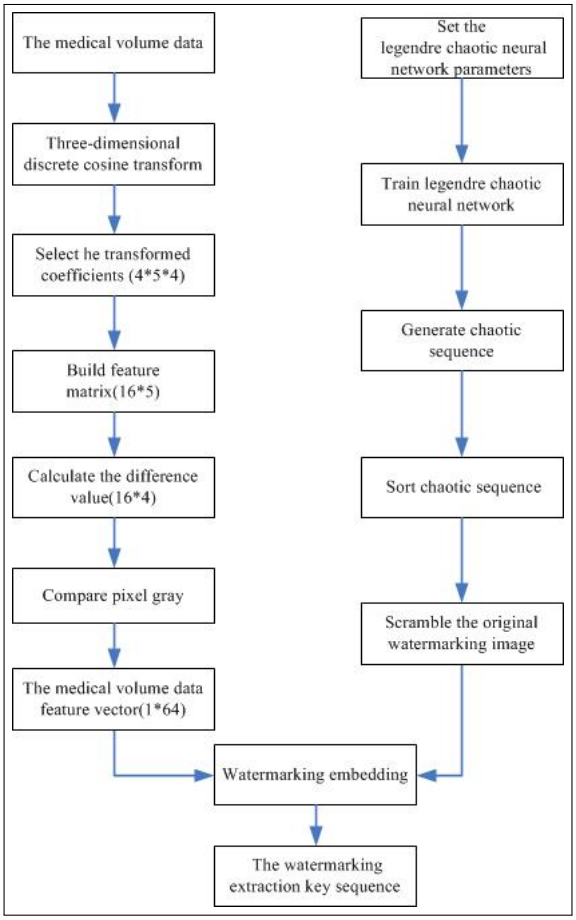


Figure 4: Embedding process

6 Simulation and analysis

The legendre chaotic neural network parameters are as follows. The number of hidden neurons is 3,$E = 10^{-4}$,$\eta = 0.1$,$l = 1000$ and the number of training is 1500 epochs. Its training error curve is shown in figure 6, in the 69 epoch it has converged to the expected error. Scrambling initial value is 0.46. The chaotic sequence for scrambling is shown in figure7. The scrambled watermarking image is shown in figure 8.

1. Without attack
   The medical volume data without attack is as shown in figure 9 (a). The slice is shown in the figure.9 (b). The extracted watermarking image is shown in the figure 9(c).

2. Filtering attack
   Medical volume data is filtered attack. Using $[5 * 5]$ median filter, repeat 10 times, the corresponding medical volume data is shown in the figure 10 (a). The slice is shown in the figure 10 (b). The extracted watermarking image is shown in the figure 10(c). This shows that the algorithm has a better anti-filter ability.

3. JPEG compression attack
   The percentage of compression quality is examined medical volume data after JPEG compression for the impact of watermarking. When the compression quality percentage is 8%, the corresponding medical volume data is shown in the figure 11 (a). The slice is shown

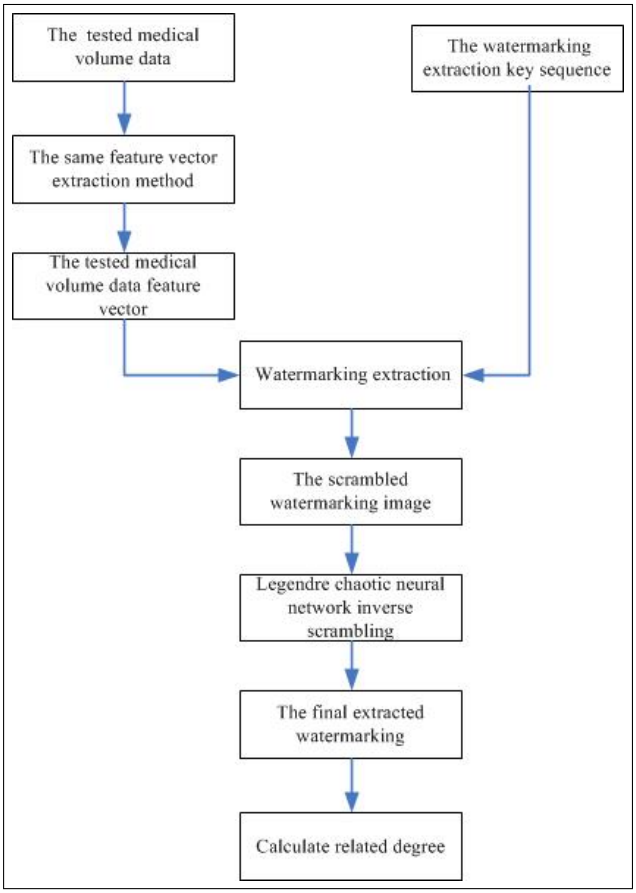


Figure 5: Extraction process

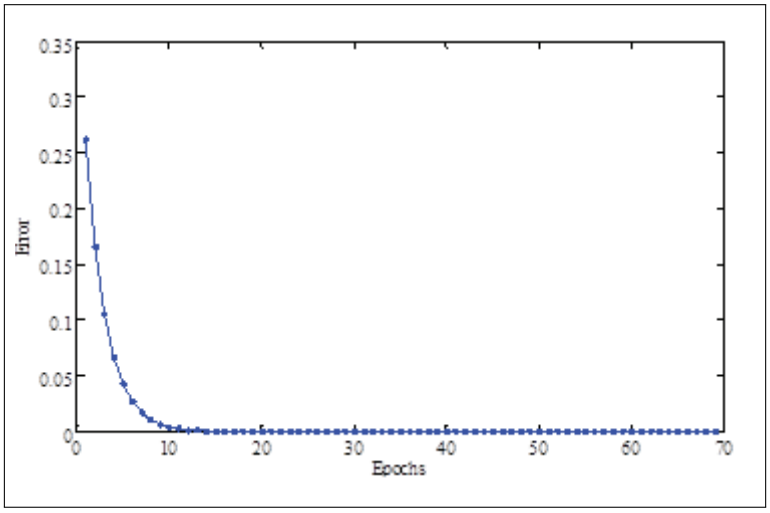


Figure 6: Training error curve

in the figure 11 (b). The extracted watermarking image is shown in the figure 11(c). This shows that the algorithm has better anti-JPEG compression capability.

4. Gaussian noise attack
Gauss noise intensity coefficient is measured the added noise interference size in medical

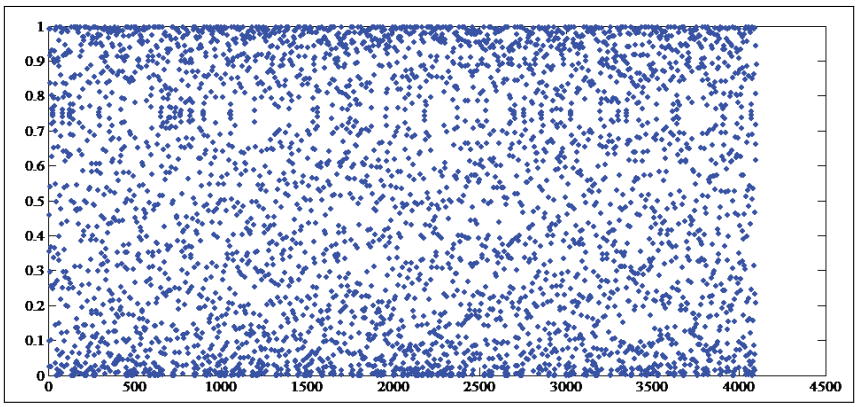


Figure 7: Chaotic sequence for scrambling

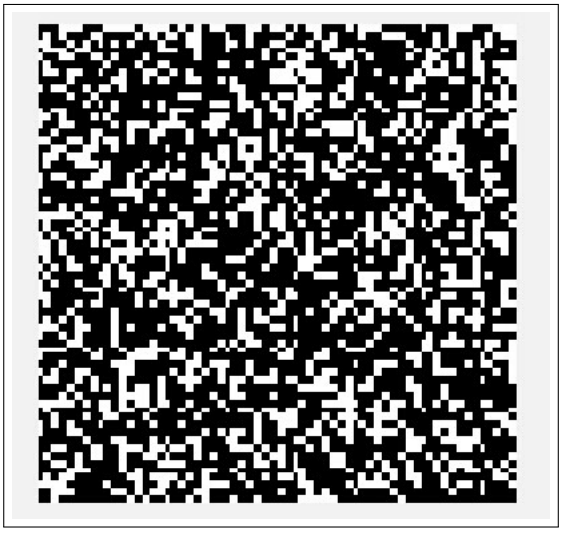


Figure 8: The scrambled watermarking image

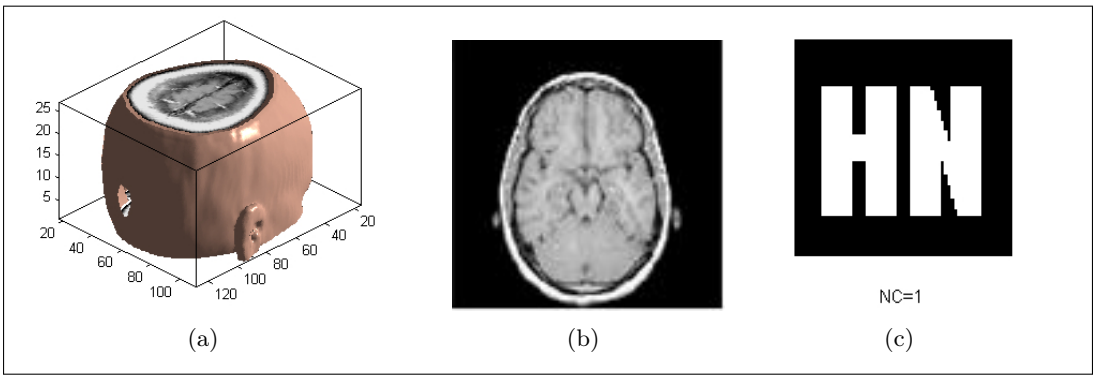


Figure 9: Simulations without attack

image. When the noise intensity is 20%, the corresponding medical volume data is shown in the figure 12 (a). The slice is shown in the figure 12 (b). The extracted watermarking is shown in the figure 12(c). This shows that the algorithm has strong robustness against noise attack.

5. Zoom attack

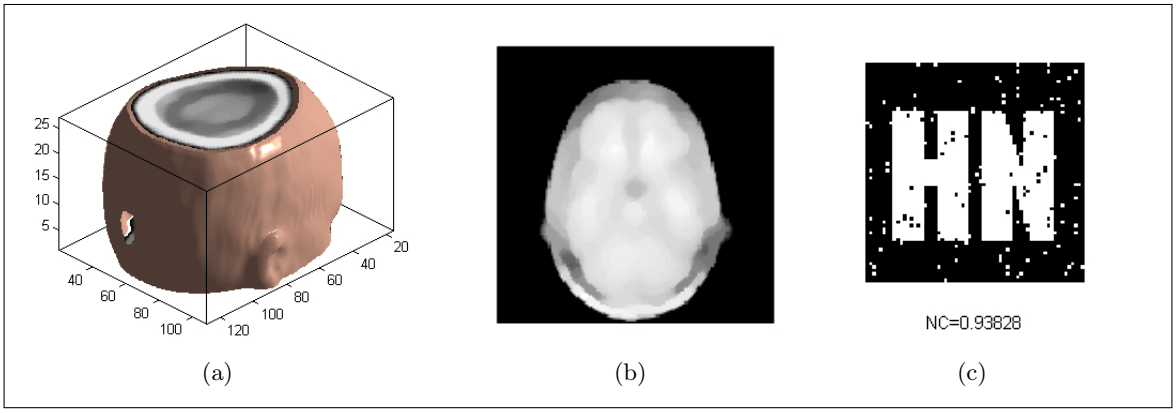


Figure 10: Simulations under filtering attack

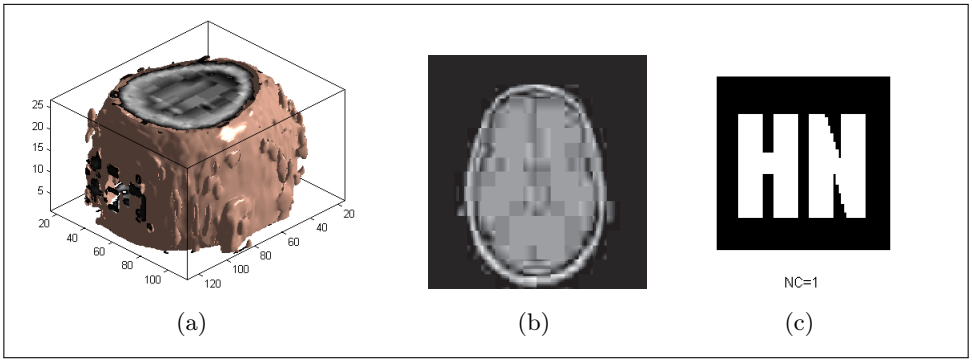


Figure 11: Simulations under JPEG compression attack

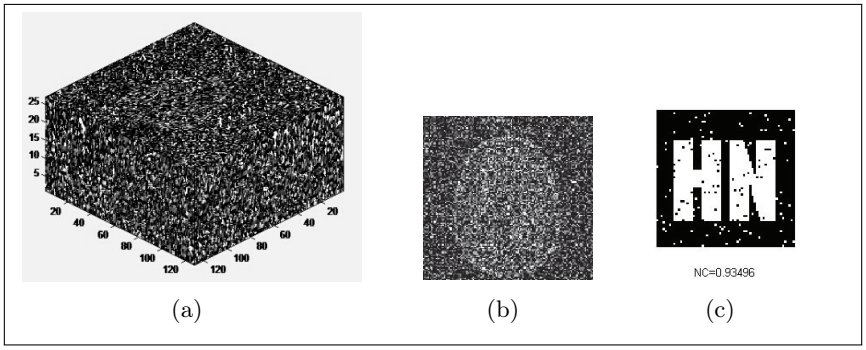


Figure 12: Simulations under filtering attack

Medical image is zoomed attack. When the zoom factor is 0.2, the corresponding medical volume data is shown in the figure 13 (a). The slice is shown in the figure 13 (b). The extracted watermarking image is shown in the figure.13(c). This shows that the algorithm has strong robustness against zoom attack.

6. Shear attack
   When the medical volume data is shear 10% from the Z-axis direction. The corresponding medical volume data is shown in the figure.14 (a). The slice is shown in the figure 14 (b). The extracted watermarking image is shown in the figure 14(c). This shows that the algorithm has a better anti-shear capability.

7. Translate attack

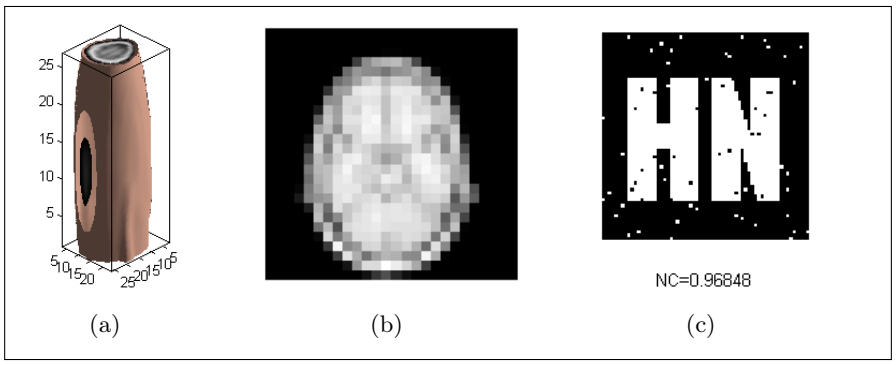


Figure 13: Simulations under filtering attack

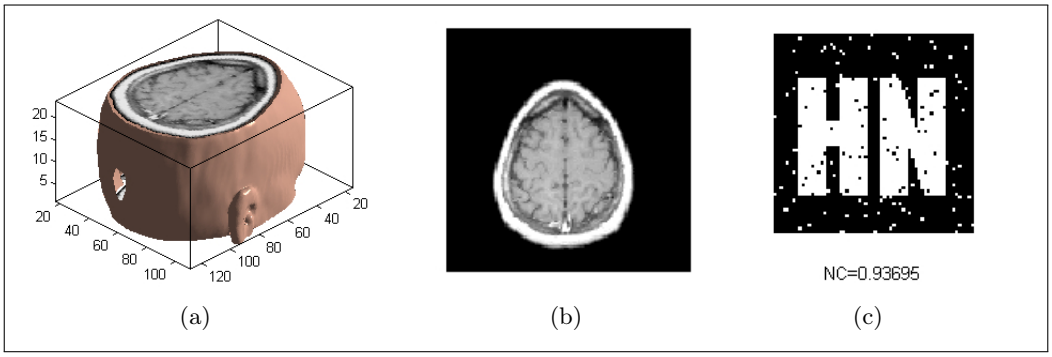


Figure 14: Simulations under filtering attack

Medical volume data is translated attack. When the vertical downward is 5%, the corresponding medical volume data is shown in the figure 15 (a). The slice is shown in the figure 15 (b). The extracted watermarking image is shown in the figure 15(c). This shows that the algorithm has a better anti-translate capability.

Figure 15: Simulations under filtering attack

8. Distort attack
   Medical volume data is distorted attack. When the distorting factor is 15, the corresponding medical volume data is shown in the figure 16 (a). The slice is as shown in the figure 16 (b). The extracted watermarking is shown in the figure 16(c). This shows that the algorithm has a better anti-distort capability.

9. Rotation attack.

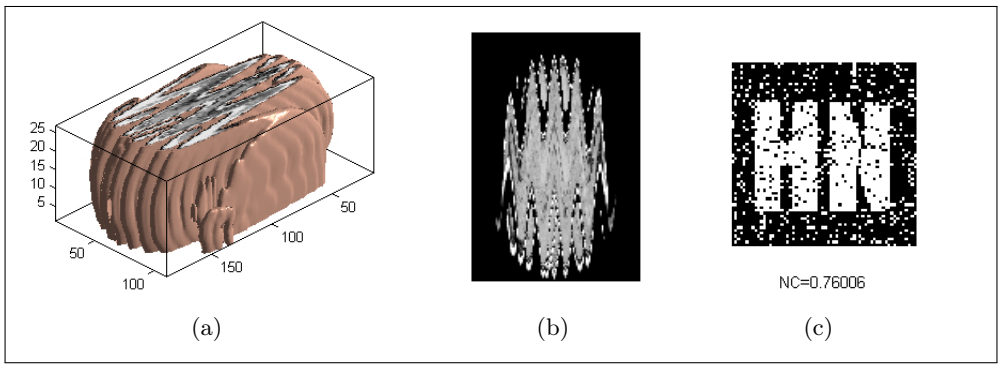


Figure 16: Simulations under filtering attack

Medical volume data is rotated attack. When the medical volume data rotated anticlockwise 10 degrees, the corresponding medical volume data is shown in the figure 17 (a). The slice is shown in the figure 17 (b). The extracted watermarking image is shown in the figure 17(c). This shows that the algorithm has a better resistance to rotation attack ability.

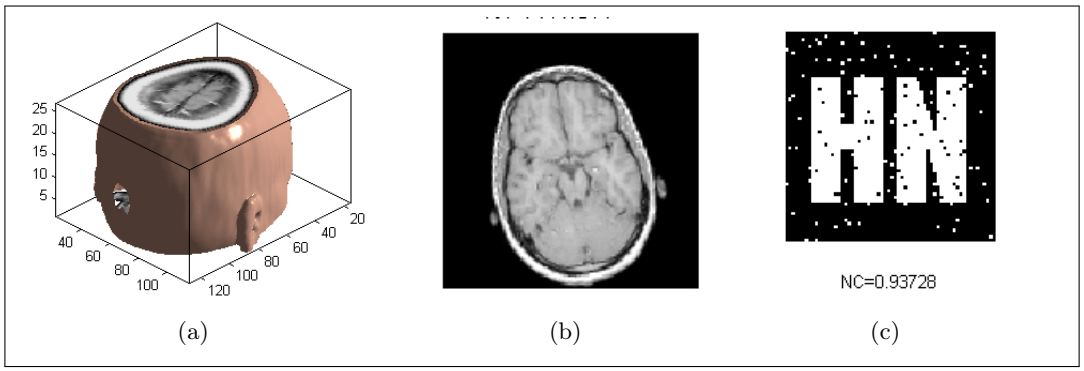


Figure 17: Simulations under filtering attack

7 Conclusion

According to the transform domains perception of the low frequency coefficient invariance and the robustness of image hashing, a new robust watermarking algorithm for medical volume data is presented in this paper. It shows that the algorithm has the following characteristics.

1. The medical volume data has the good transparency without making any changes to the original medical data.

2. It uses the discrete cosine transform coefficients of low frequency stability characteristics and difference hashing algorithm robustness, thereby increasing the robustness of the algorithm.

3. The relationship between the initial values and chaotic sequences is contained in the chaotic neural network, which is inherently unpredictable. Therefore, the algorithm is theoretically absolutely security.

4. The watermarking can be extracted without original medical volume data, and it realizes the blind detection. The results show that this algorithm has better robustness and security.

8 Acknowledgements

This work was supported by the National Natural Science Foundation of China (No: 61263033) and the Natural Science Foundation of Hainan Province (No: 614241).

Bibliography

[1] R.Barnett (1999); Digital Watermarking: applications. Techniques and Challenges, *Electronics & Communications of Engineering Journal*, 11(4):173-183.

[2] Cox I J, Kilianj, Leighton F T. (2003); Secure spread specture watermarking for multimedia . *IEEE Trans on Image Processing*, 6 (12):1673-1687.

[3] Xueming Li, Guangjun He.(2012); Efficient audio zero-watermarking algorithm for copyright protection based on BIC and DWCM matrix, *Int. J. of Advancements in Computing Technology*, 4 (6): 109-117.

[4] Bami M, Bartolinit F, Rosa A D. (2000); Capacity of full frame DCT image watermarks, *IEEE Trans. on Image Processing*, 9 (8):1450-1455.

[5] Guanghui Cao, Hu Kai (2013); Image scrambling algorithm based on chaotic weighted sampling theory and sorting transformation, *Journal of Beijing University of Aeronautics and Astronautics*, 39(1):67-72.

[6] Yaoli Liu, Jingbing Li (2013); A medical image robust multi watermarking method based on DCT and Logistic Map, *Application Research of Computers*, 30(11):3430-3433.

[7] J-L Dugelay, S. Roche, C. Rey, G. Doerr (2006); Still image watermarking robust to local geometric distortions. *IEEE Trans. on Image Processing*, 15(9):2831-2842.

[8] Giakoumaki A., Pavlopoulos S., Koutsouris D. (2006) Multiple image watermarking applied to health information management. *Information Technology in Biomedicine, IEEE Transactions on*, 10(4): 722-732.

[9] Wu J H K, Chang R F, Chen C J, et al.(2008); Tamper detection and recovery for medical images using near-lossless information hiding technique, *Journal of Digital Imaging*, 21(1): 59-76.

[10] Tan C.K., Ng J.C., Xu X.T. Security protection of DICOM medical images using dual-Layer reversible watermarking with tamper detection capability. *Journal of Digital Imaging.* 2011, 24(3):528-540.

[11] Deng X.H., Chen Z.G., Deng X.H., et al. (2001); A Novel Dual-Layer Reversible Watermarking for Medical Image Authentication and EPR Hiding, *Advanced Science Letters*, 4(11):3678-3684.

[12] Baoru Han, Jingbing Li (2013); A robust watermarking algorithm for medical volume data based on hermite chaotic neural network, *Int. J. of Applied Mathematics and Statistics*, 48(18):128-135.

[13] Baoru Han, Jingbing Li and Liang Zong (2013); A new robust zero-watermarking algorithm for medical volume data, *Int. J. of Signal Processing, Image Processing and Pattern Recognition*, 6(6):245-258.

[14] H. C. Yang, S. B. Zhang, and Y. B. Wang (2012); Robust and precise registration of oblique images based on scale-invariant feature transformation algorithm, *IEEE Geosci. Remote Sens. Lett.*, 9(4): 783-787.

[15] Q. L. Li, G. Y. Wang, J. G. Liu, and S. B. Chen (2012); Robust scale-invariant feature matching for remote sensing image registration, *IEEE Geosci. Remote Sens. Lett.*, 6(2): 287-291.

[16] Y. Lei, Y. Wang, and J. Huang (2011); Robust Image Hash in Radon Transform Domain for Authentication. *Signal Processing: Image Comm.*, 26(6): 280-288.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):200-210, April, 2015.

HAPA: Harvester and Pedagogical Agents in E-learning Environments

M. Ivanović, D. Mitrović, Z. Budimac, L. Jerinić, C. Bădică

Mirjana Ivanović*, Dejan Mitrović, Zoran Budimac, Ljubomir Jerinić
Department of Mathematics and Informatics
Faculty of Sciences, University of Novi Sad, Serbia
mira@dmi.uns.ac.rs, dejan@dmi.uns.ac.rs, zjb@dmi.uns.ac.rs, jerinic@dmi.uns.ac.rs
*Corresponding author: mira@dmi.uns.ac.rs

Costin Bădică
Computer and Information Technology Department
Faculty of Automatics, Computers and Electronics,
University of Craiova, Romania
cbadica@software.ucv.ro

Abstract: In the field of e-learning and tutoring systems two categories of software agents are of the special interest: *harvester* and *pedagogical* agents. This paper proposes a novel e-learning system that successfully combines both of these agent categories and introduces two distinct sub-types of pedagogical agents *helpful* and *misleading*. Whereas helpful agents provide the correct guidance for the given problem, misleading agents try to guide the learning process in the wrong direction by offering false hints and inadequate solutions. The rationale behind this approach is to motivate students not to trust the agent's instructions blindly, but to employ critical thinking. Consequently, students will be put in a "softly stressed" environment in order to prepare them for real working environments in their future work in companies. Nevertheless students themselves will decide on the correct solution to the problem in question.
Keywords: E-learning, adaptability, personalization, intelligent agent, harvester agents, pedagogical agents.

1 Introduction

Software agents (or simply *agents*), can be defined as *autonomous* software entities with various degrees of *intelligence*, capable of exhibiting both *reactive* and *pro-active* behavior in order to satisfy their design goals. From the point of e-learning and tutoring systems, two types of agents are of special research interest: *harvester* and *pedagogical*. Harvester agents collect learning material from online, heterogeneous repositories. The core properties of the agent technology (e.g. parallel and distributed execution, mobility, and inter-agent communication) can bring significant benefits to the harvesting process [16]. Pedagogical agents can be defined as "lifelike characters presented on a computer screen that guide users through multimedia learning environments" [6]. Their main goals are to motivate and guide students through the learning process [7].

This paper presents a stand-alone e-learning architecture named *HArvester and Pedagogical Agent-based e-learning system* (HAPA), and designed to help learners during solving programming tasks. HAPA consists of three main components: *Harvester agents*, *Classifier module*, and *Pedagogical agents*. The harvester agents collect the appropriate learning material from the web. Their results are fed into the Classifier module, which performs automatic classification of individual learning objects. Finally, a pair of specially designed Pedagogical agents - one *helpful* and one *misleading* - is used to interact with students and guide them to comprehend the learning material. The helpful pedagogical agent provides useful hints for the problem in

question. The misleading pedagogical agent guides the learning process in the wrong direction. Because the student is never sure with which agent (s)he is interacting, this novel approach encourages students not to follow the agent's/tutor's instructions blindly, but rather to employ critical thinking. We believe that this "softly stressed" environment could help learners to face stressful and competitive real working environments.

To the best of our knowledge, none of the existing e-learning systems employs this combination of pedagogical agents in conjunction with harvester agents for collecting additional learning material. This is the main idea and contribution behind the work presented in this paper. The initial ideas of using harvester and two types of pedagogical agents were presented in [8,9]. This paper concentrates on improvements and a concrete implementation of initial ideas, with a well-defined set of functional components. Harvester agents are now defined as *web crawlers* [11], specialized for collecting Java source code examples. The new Classifier module has also been defined, and a set of tools for preparing helpful and misleading hints, and visual representation of pedagogical agents has been implemented. Early evaluation results are presented as well.

The rest of this paper is organized as follows. Section 2 provides an overview of the existing work related to the employment of harvester and pedagogical agents. In Section 3, a detailed insight into the proposed system is presented. Section 4 brings implementation details and early evaluation results. Overall conclusions are given in Section 5.

2 Related work

There are many interesting approaches to using software agents in e-learning and tutoring environments. For example, *ABITS* [2], *MathTuthor* [3], and *Educ-MAS* [4] incorporate intelligent agents in order to improve the students' learning outcomes. However, none of these systems use harvester and pedagogical agents in the same environment.

It was shown in [17] that inherited properties of the agent technology – parallel and distributed execution – can be used to optimize the web crawling process of harvesting agents. The same approach has been taken in HAPA, except that our agents are highly specialized to search for syntactically correct Java source code examples.

Agent Based Search System (ABSS) relies on harvester agents to improve the quality of search query results [15]. The system is capable of not only harvesting heterogeneous remote learning object repositories, but also tracking changes in them. Similarly, *AgCAT* represents an agent-based federated catalog of learning objects [1]. The harvesting process is delegated between two agents - *Librarian* and *InterLibrarian* - that, respectively, maintain the local repository of learning objects, and perform the federated search and retrieval on remote repositories.

Both of these systems use sophisticated harvesting agents to retrieve the best-suited learning objects. The difference between HAPA and both ABSS and AgCAT is in the approach used to deliver the harvested content. ABSS and AgCAT are sophisticated search engines; they enable their users to *pull* the data using search queries. On the other hand, our system monitors and evaluates the student's progress through a course. If a decline in student's performance is detected, HAPA can harvest additional appropriate learning material and then *push* it to the student.

An interesting analysis of 39 studies related to the effects of pedagogical agents onto the learning outcome has been presented in [6]. The initial conclusion is that only 5 studies have detected positive effects of using pedagogical agents. However after a more detailed analysis it was observed that only 15 of the 39 studies used a control group without an agent, while actual motivational approaches were implemented in only 4 of these 15 studies.

Our first intention was not to implement visual representations of pedagogical agents in HAPA environment. However since several studies [5,6] discuss that un-appealing visual representations

of an agent can have a negative impact on the student's willingness to interact with the agent, we decided to visualize our pedagogical agent(s). We implemented a simple character (see Figure 2) and let students decide to use it or not during learning and assessment activities. Both versions of our pedagogical agents are represented with the same visual character.

SmartEgg is a web-based pedagogical agent that assists students in learning SQL [13]. It is integrated into an intelligent e-learning system *SQL-Tutor*. The agent includes a visual representation with animated gestures, and can express different behaviors: introductory, explanatory, and congratulatory. SmartEgg is relatively simple, and is employed just as a more pleasant way of presenting the learning material.

As shown, there is a lot of ongoing research related to the usage of harvester and pedagogical agents in e-learning environments. However, although many existing systems incorporate either type of agents, there have been no previous attempts to efficiently integrate both harvesting and motivational-level agents. An additional, and more important contribution of HAPA is the concept of helpful and misleading pedagogical agents.

3 HAPA System overview

HAPA is currently a stand alone e-learning system which helps students in learning and especially in solving programming problems. At a later stage HAPA could be included as a component in other tutoring systems devoted to learning programming languages. Our intention is to incorporate it in *Protus*, a tutoring system we designed to help learners in learning essentials of Java programming language [10, 19].

A high-level overview of the system architecture is outlined in Figure 1. HAPA includes several important components: harvester agents, the Classifier module, repositories of helpful and misleading hints, and pedagogical agents. The functioning of each component and their mutual interactions are described in more details in the following sub-sections.

3.1 Harvesting and classifying the learning material

HAPA is mainly focused on the *code completion* type of tasks, in which students are expected to fill-in missing parts of the program. The student is given a code snippet, and then requested to complete the source code to meet the program specification. Code completion tasks are well-suited for both testing and improving the student's programming skills, because they require a thorough understanding of the underlying programming concepts.

As the initial step of constructing the code completion tasks, the additional learning material is collected by the harvester agents. The learning material consists of Java source code examples. With the abundance of these examples available on the web, harvester agents have been implemented as web crawlers [11]. The system administrator (e.g. the teacher) specifies starting web pages. Harvester agents scan these pages in search of syntactically valid Java source code examples. First, a simple search for some more important Java keywords, such as *class*, is performed. A block of text that contains keywords is then processed by a syntax analyzer to determine whether it is a valid Java program. The text can either be directly embedded in the web page or attached to the page as an external file (e.g. a *ZIP* archive).

After it processes the current page, the agent continues the harvesting process on all pages linked from the current one, and so on. Many agents can be deployed on a computer cluster and perform the harvesting in parallel. A centralized repository of visited pages is maintained in order to avoid duplicate work. The harvested learning material is fed into the Classifier module (see Figure 1), which automatically associates each Java source code example with a concrete lecture topic. The classification is performed via the static source code analysis. The Classifier module

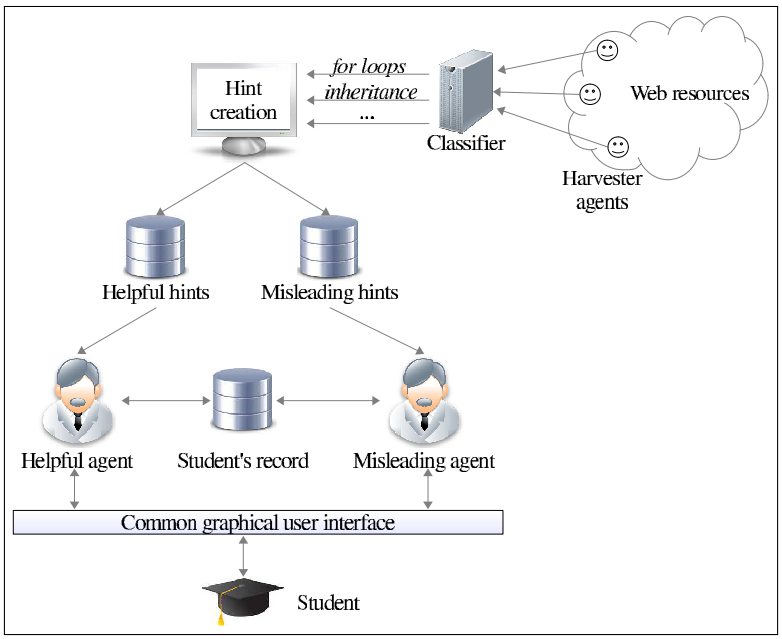


Figure 1: A high-level overview of the HAPA system.

constructs an *abstract syntax tree* for the given Java example, and then inspects programming constructs that appear in the tree. As a result, each example is assigned to the appropriate lecture topic. In return, the teacher is able to analyze and focus on examples of a particular interest, i.e. those that are directly attached to the lecture topic in question.

Currently, the Classifier's decision on which example belongs to which topic is a suggestion to the teacher. In the end, the teacher makes the final selection and filter the obtained source code examples. In order to improve the Classifier's performance, more intelligent source code classification techniques will be implemented in the future (e.g. [12]).

Once the harvested learning material has been classified, the teacher can use them select most appropriate solutions for learning topic and prepare the code completion tasks. This step is performed manually, using a specially designed GUI tool. The tool enables teacher to quickly scroll through the classified Java source code examples, select the ones to be actually used, and process them by removing parts of the code and constructing useful and misleading hints which will be offered to students. The hints are incorporated in pedagogical agent and used in learning.

3.2 Pedagogical agents

The significant novelty of this work is incorporation in a learning environment two different types of pedagogical agents – helpful, and misleading. Both agents are hidden from the student behind the same interface and visual representation (see Figure 2), and take turns in interacting with the student at random time intervals. Therefore, the student is never sure with which agent he/she is interacting. The rationale behind this approach is to motivate students not to trust the agent's hints blindly. Instead, they should critically analyze the problem and the proposed hint, and independently decide on the proper solution.

In the scientific literature and the actual software products, it is common to represent pedagogical agents as lifelike, animated characters. Although we feel that there is no real value in this approach we nevertheless decided to implement simple visual Pedagogical HAPA agent and let students decide if they will activate it or not, using the on/off switch button. But, although maybe "fun" to look at in the beginning, over the time the visual character stands in the way

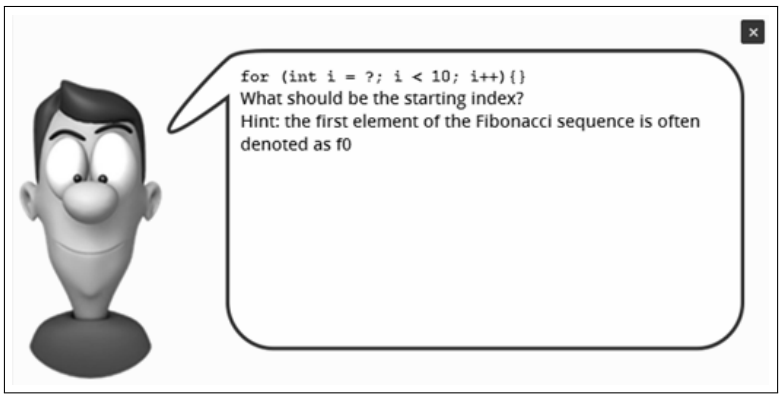


Figure 2: Visual representation of a Pedagogical HAPA agent.

of solving the assignment. This was somehow confirmed during the experimental phase with students. They distract the student from concentrating on the problem in question, and in the extreme case, may negatively affect his/her willingness to use the system.

Both pedagogical agents are capable of adapting to each individual student. Agents track a set of information about the student, including his/her personal data (such as class and age), the ratio of correct and incorrect solutions to each code completion problem, and the student's grade for each lecture topic. Based on the accumulated data agents can intervene if the student's success rate becomes unsatisfactory. For example, if the student gives to many wrong answers to questions regarding *for loops*, the pedagogical agent will recommend additional learning material. Additionally, it will repeat the appropriate code completion tasks until a certain success threshold is reached.

We believe that if students do not know if agent gives correct or wrong directions and hints, it will additionally motivate students to critically think and assess their knowledge. Also in future real working environments they will face different helpful but also malicious colleagues who will maybe suggest them wrong procedures and steps. So we would like to put students in unexpected situations and motivate them to reassess their knowledge and skills.

4 HAPA implementation and evaluation

Previously described functionalities of HAPA were used to guide the implementation process. For example, harvesting is a process that can and should be distributed and executed in parallel. Then, students should be able to interact with and use HAPA through a web interface. And, like all web-based systems, HAPA should be resilient to hardware and software failures, malicious attacks, etc. Given these implementation requirements, and its popularity in developing software agents and multi-agent systems, Java has been chosen as the implementation platform for HAPA.

4.1 Helpful and misleading hints

In order to provide the reader with a better insight into the evaluation of HAPA, some examples of the prepared code completion tasks are presented here. The two given tasks are tailored to topics on *for loops* and *classes* in Java, respectively. Helpful and misleading hints assigned to each task are also presented and discussed.

The task tailored to the topic on *for loops* in Java requires the student to complete a program for calculating the first 10 members of the *Fibonacci sequence*. The skeleton program presented to students is shown in Listing 1.

Listing 1: Code completion task related to *for loops*

```
class Fig {
  public static void main(String[] args) {
    int[] f = new int[10];
    // TODO : implement the for loop here
    print(f); } }
```

Based on this skeleton, the following set of helpful and misleading hints for pedagogical agents have been prepared [8].

1. *for (int i = ?; i < 10; i++){}* "What should be the starting index? Remember that the first element of the Fibonacci sequence has the index 0, while the expression for calculating other elements is $f_i = f_{i-1} + f_{i-2}$"

2. *for (int i = 0; i ≤ ?; i++){}* "What should be the ending index? Although you need 10 numbers, remember that the index of the first element is 0."

3. *for (int i = 0; i < 10; ?){}* "Should you use $++i$ or $i++$ to modify the value of i? Remember that this modification is always executed at the end of the for loop."

4. *for (int i = ?; i < 10; i++){}* "What should be the starting index? Hint: the first element of the Fibonacci sequence is often denoted as f_0."

5. *for (int i = 0; i ≤ ?; i++){}* "What should be the ending index? Hint: look at the initialization of the array f – how many elements does it have?"

6. *for (int i = 0; i < 10; ?){}* "Should you use $++i$ or $i++$ to modify the value of i? Remember that $++i$ first increases the value of i, and then uses the new value."

By suggesting that f_0 is the first element of the Fibonacci sequence in hint 4, the misleading agent tries to suggest the improper usage of 0 for the initial value of i. In the expression $f_i = f_{i-1} + f_{i-2}$, this decision would cause the index to go out of the array bounds. Similarly, in hint 5, the agent suggests that the student should use 10 as the final value of i (note the expression $i \leq ?$), disregarding the fact that Java array indexes are 0-based. The final hint 6 is there to confuse the student, since both $++i$ and $i++$ are correct.

Listing 2 shows the skeleton for a more complex task tailored to classes, fields and methods. The given class represents a rectangle, defined by its upper-left point *(x, y)*, *width*, and *height*. The student's task is to write a method that calculates the rectangle area. The focus is on the proper definition of the method's input parameter and the return value.

Listing 2: Code completion task related to classes.

```
class Rect {
  private float x, y, width, height;
  public Rect(float x, float y, float width, float height) {
    // the parameters are saved into corresponding fields
  }
  // TODO : calculate the rectangle area here
}
```

For this example the following set of hints are defined. Hints 7 and 8 are used by the helpful, while hints 9 and 10 are used by the misleading agent.

7. "To calculate the area, you need width and height, both of which are available as fields. Remember that a method can access its object's fields without limitations."
8. "While writing the method, remember that somebody needs to use the result; outputting it on the screen won't be of much use to anybody!"
9. "To calculate the area, you need both width and height. Make sure your method has access to these values!"
10. "Remember that long command for outputting values on screen? Here's a hint: it starts with 'System'."

Given misleading hints are based on our long-term experience in conducting exercises for the introductory Java programming course. It has been observed that initially, a relatively large portion of students has the problem of grasping the concept of fields, methods, and method arguments. That is, they tend to specify fields as method arguments, rather than to use them directly. This is what the misleading hint 9 tries to suggest. The given constructor implementation that receives both width and height also works in favor to this suggestion.

Similarly, when asked to calculate some value (in this case, the rectangle area), beginner students often tend to just print the value on screen, rather than to return it from the method. This is what hint 10 tries to lead them to.

After preparing these and many other code completion tasks and hints, the system has been evaluated in practice. We expected that evaluation results would obtain adequate feedback necessary to continue our efforts and improve system's functionalities.

4.2 Evaluation results

Following the implementation of HAPA, an evaluation of the system was conducted during school year 2013/14. The main goal of the evaluation was to examine the effects of helpful and, more importantly, misleading pedagogical agents.

Because the current implementation of HAPA is the first prototype of the system, the evaluation was performed on a small group of self-motivated students. 24 second-year students were selected, on the basis of their previous programming experience, as well as their scores in previous programming-related course at our Department of mathematics and informatics, Faculty of Sciences (DMI) and 18 students from Department of information technology at the Higher School of Professional Business Studies (DMT).

During the semester, from time to time the students were given a number of Java code completion tasks to solve. Once it presents a task, the system waits for the student's input for a certain amount of time. If no input is detected, a pedagogical agent provides a hint for solving the task. As noted earlier, helpful and misleading agents take turns at random time intervals, and the student is never sure with which agent he/she is interacting. Moreover, students were not even aware of the fact that there are two types of pedagogical agents.

Students' responses and actions were logged by the system. After the testing an analysis of the log was conducted in order to determine whether the hints have had any influence onto the students' though processes. As they used the system partially in blended learning style, for some lessons to obtain their opinion we did not use classical form of questionnaire. We just interview them and ask them about their opinion. We expected that in such friendly atmosphere they would be completely honest. The summary results are presented below.

Students from DMI showed better results and are were more eager in using HAPA system. Students from DMT made a lot of mistakes and were in majority of cases frustrated by using the system. More detailed explanations are given in the rest of the section.

Group DMI_G1 consists of eighteen students from DMI, they have had previous programming experience, have achieved great scores in a previous programming course and were among the best ones. Group DMI_G2 consists of six other students that have previous programming experiences but did not pass previous programming course.

12 students from DMI_G1 were very cautious and thought critically most of the time, and recognized and ignored hints from misleading agent. Other 6 students also employed critical thinking and in majority of cases recognized misleading hints. But in several cases when they were no sure about the proper choice they accepted hints form misleading agent.

4 students from DMI_G2 blindly followed all hints and did not think whether they are correct or not. 2 other students were a little bit confused, they thought that some hints were wrong, but as they were not absolutely sure about that they decided to follow all hints. This is probably the result of their inappropriate knowledge.

In the next several cases we will illustrate some specific students' behaviors. One student among the best students had no problem in solving the tasks, and, according to his own account, rarely considered hints. This is because he had already learned Java programming in high-school, and so could understand and solve the problems without any assistance.

Another student has considered the hints, but was also able to solve majority of the tasks without any assistance. However, he indicated that hint 7 helped him to solve the problem. He has also correctly observed that some of the presented hints were wrong.

A more interesting situation was with the two students who had no previous Java programming experience and have low scores on previous programming course. One of them took hint 4 for granted and received the *array-index-out-of-bounds* exception. On the second attempt, she was given hint 1 and was able to complete the task successfully. A similar scenario has been observed with the fourth student. By accepting hint 9, he initially wrote a method that accepts both width and height as input parameters. Then, after receiving hint 7, he was able to complete the task successfully.

The system was also tested and used among 18 students of the DMT. Involved students were programming beginners and they used system after completion of the first programming course.

The results of the experiments showed that the students of DMT had more trust in agents, and made more mistakes based on the hints of misleading pedagogical agents. Majority of them (even 13) blindly followed suggestions and believed that hints were well-intentioned. Other were more or less confused and do not know if can trust or not to obtained hints.

Several interesting conclusions can be drawn from these results. If students are unable to solve the problem on the first try, they have the tendency to trust the agent's/tutor's hints. This is because they have never encountered a misleading agent/tutor before and do not anticipate such behavior. The student who had noticed the wrong hint was confused but assumed that it was an implementation error. For some future work, it could be beneficial if students were informed that some hints might be intentionally misleading.

Secondly, students who are not confident in their Java skills find the presence of a virtual agent/tutor "reassuring." Additionally, the misleading behavior and the attempt of the agent to "trick them" transform HAPA into a kind of a game with the goal of beating the system. Both of these effects have a significant positive impact onto the students' motivation to use the system and employ critical thinking. These are the exact goals set for our proposed system.

Finally, results suggest that students from DMT were more misled by pedagogical agents than students from DMI. A reasonable explanation could be that students from DMI are better (achieved better results in secondary school, passed with high marks first programming course) and they are generally highly motivated (usually students from DMI are more ambitious about their future career and jobs) to master their programming knowledge and skills.

Both groups of students have the similar attitude to visual representation of pedagogical

agent(s) and their use in the system. As we mentioned, students were allowed to explicitly (de)activate visual forms of agents. In the beginning of using system it was attractive to them to see agents and communicate with them so they intensively used them. Lately, especially when they did not know how to solve task and as they were getting more tired and nervous about hints, visual agents irritated them and they decided to switch off visual forms of agents.

5 Conclusions and future work

The agent technology has been recognized as a useful tool in a wide variety of domains. For e-learning and tutoring systems, harvester and pedagogical agents are of the special interest.

The main contribution of this paper is the proposal of a new e-learning system named HAPA that incorporates both harvester and pedagogical agents. Harvester agents, along with the Classifier module, are designed to collect the best-suited Java source code examples from the web, and tailor them to particular lessons within a course.

A more important functionality, however, is achieved by defining two new sub-types of pedagogical agents – helpful and misleading. As noted, the helpful pedagogical agent provides correct suggestions and hints for the problem in question. On the other hand, the misleading agent tries to guide the problem solving process in a wrong direction, by offering false suggestions and hints. The main aim for this approach is to motivate students not to follow the agent's directions blindly, but instead to analyze both the problem and the suggestions thoroughly and employ critical thinking. According to our knowledge, none of the existing e-learning systems use this kind of helpful and misleading pedagogical agents in combination with harvester agents.

HAPA is currently realized as a prototype. Future improvements will be concentrated on integrating it into our existing, fully-featured web-based e-learning architecture Protus [10, 19], but it can also be integrated in some other types of available learning systems [14, 18]. In order to achieve adaptability and personalization, Protus incorporates several models, including: *Domain model*, which serves as a storage of the learning material, *Student model* for maintaining both static and dynamic information about each individual student, *Application model*, which applies different strategies on the input received from the learner model in order to ensure efficient personalization, and the *Adaptation model* that follows the instructional directions provided by the application module in order to organize learning resources into a navigational sequence tailored to the particular learner.

Obviously, modules that comprise HAPA fit nicely into the organizational models of Protus. Harvester agents, along with the Classifier module, can be used to obtain and generate learning material for the domain model. The learning material stored in the domain model consists of individual lessons, which are further decomposed into tutorials, accompanying examples, and tests. Therefore, harvester agents and the Classifier module can be used to collect examples and tests. Proposed helpful and misleading pedagogical agents can be integrated into the adaptation model. This integration will harness the benefits of both architectures. The resulting system will be capable of providing high-quality tasks and examples, and exhibiting adaptive and personalized behavior. It will offer motivational pedagogical agents that guide students through the learning process and encourage critical thinking.

Having in mind that usual way of employing pedagogical agents is to help students to systematically test their self-confidence and knowledge and somehow keep them less active and expecting constant positive help we decided to apply the opposite way. We believe that if students do not know if agent gives correct or wrong directions and hints, will additionally motivate students to critically think. Also in future real working environments they will face different helpful but also malicious colleagues who will maybe suggest them wrong procedures and steps.

So we would like to put students in unexpected situations and prepare them to face rather stress and competitive real working environments.

Acknowledgment

This work was partially supported by Ministry of Education, Science and Technological Development of the Republic of Serbia, through project no. OI174023: "Intelligent techniques and their integration into wide-spectrum decision support."

Bibliography

[1] Barcelos, C.F., and Gluz, J.C.(2011); An agent-based federated learning object search service. *Interdisciplinary journal of e-learning and learning objects 7*, 37-54.

[2] Capuano, N., Marsella, M., Salemo, S. (2000); ABITS: an agent based intelligent tutoring system for distance learning. In *Proceedings of the International Workshop in adaptative and intelligent web-based educational systems*, 17-28.

[3] Cardoso, J., Guilherme, B., Frigo, L., and Pozzebon, L.B. (2004); MathTutor: a multi-agent intelligent tutoring system. In *First IFIP conference on AIAI* , 231-242.

[4] Gago, I.S., Werneck, V. M., and Costa, R. M. (2009); Modeling an educational multi-agent system in MaSE, In *Proceedings of the 5th international conference on active media technology (AMT'09)* , 335-346.

[5] Haake, M., and Gulz, A. (2008); Visual stereotypes and virtual pedagogical agents. *Educational Technology & Society 11*, 4: 1-15.

[6] Heidig, S., and Clarebout, G. (2011); Do pedagogial agents make a difference to student motivation and learning? *Educational Research Review 6*, 27-54.

[7] Heller, B., and Procter, M. (2010); Animated pedagogical agents and immersive worlds: two worlds colliding. *Emerging Technologies in Distance Education*, 301-316.

[8] Ivanović, M., Mitrović, D., Budimac, Z., Vesin, B., and Jerinić, L. (2014); Different roles of agents in personalized learning environments. In *10th International Conference on Web-Based Learning (ICWL 2011), LNCS*, Springer, 7697: 161-170.

[9] Ivanović, M., Mitrović, D., Budimac, Z., and Vidaković, M. 92001); Metadata harvesting learning resources – an agent-oriented approach. In *Proceedings of the 15th International Conference on System Theory, Control and Computing (ICSTCC 2011)*, October 2011, 306-311.

[10] Klašnja-Milićević, A., Vesin, B., Ivanović, M., and Budimac, Z. (2011); Integration of recommendations and adaptive hypermedia into Java tutoring system. *Computer Science and Information Systems 8*, 1: 211-224.

[11] Kobayashi, M., and Takeda, K. (2000); Information retrieval on the web, *ACM Computing Surveys 32*, 2: 144-173.

[12] Linstead, E., Bajracharya, S., Ngo, T., Rigor, P., Lopes, C., and Baldi, P. (2009); Sourcerer: mining and searching internet-scale software repositories. *Data Mining and Knowledge Discovery 18*, 2: 300-336.

[13] Mitrovic, A., and Suraweera, P. (2000); Evaluating an animated pedagogical agent. In *Intelligent Tutoring Systems*, Springer, 73-82.

[14] Ocepek, U., Bosnic, Z., Serbec, I.N., and Rugelj, J. (2013); Exploring the relation between learning style models and preferred multimedia types. *Computers & Education 69* , 343-355.

[15] Orzechowski, T. (2007); The use of multi-agents' systems in e-learning platforms. In *Siberian conference on control and communications (SIBCON'07)*, 64-71.

[16] Prieta, F. D.L., and Gil, A.B. (2010); A multi-agent system that searches for learning objects in heterogeneous repositories. In *Trends in Practical Applications of Agents and Multiagent Systems, Advances in Intelligent and Soft Computing*, Springer, 71: 355-362.

[17] Sharma, S., Gupta, J.P.(2010); A novel architecture of agent based crawling for OAI resources. *International Journal of Computer Science and Engineering 2*, 4: 1190-1195.

[18] Stuikys, V., Burbaite, R., and Damasevicius, R. (2013); Teaching of computer science topics using meta-programming-based GLOs and LEGO robots. *Informatics in Education 12*, 1: 125-142.

[19] Vesin, B., Ivanović, M., Klašnja-Milićević, A., and Budimac, Z. (2013); Ontology-based architecture with recommendation strategy in java tutoring system. *Computer Science and Information Systems 10*, 1: 273-261.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):211-221, April, 2015.

Modelling and Analysis of Mobile Computing Systems: An Extended Petri Nets Formalism

L. Kahloul, A. Chaoui, K. Djouani

Laid Kahloul
LINFI Laboratory, Computer Science Department, University of Biskra
Biskra, 07000, Algeria.
E-mail: kahloul2006@yahoo.fr

Allaoua Chaoui
MISC Laboratory, Computer Science Department, University of Constantine
Constantine, 25000, Algeria.
E-mail: a_cahoui2001@yahoo.fr

Karim Djouani
LISSI Laboratory, Paris Est University, Paris, France
F'SATI at TUT, Pretoria South Africa.
E-mail: djouani@univ-paris12.fr

Abstract: In its basic version, Petri Nets are defined as fixed graphs, where the behaviour of the system is modelled as the marking of the graph which changes over time. This constraint makes the Petri Nets a poor tool to deal with reconfigurable systems as mobile computing systems, where the structure of the system can change as its behaviour, during time. Many extended Petri nets were proposed to deal with this weakness. The aim of this work is to present a new extension of Petri Nets, where the structure of the graph can be highly flexible. This flexibility gives a rich model with complex behaviours, not allowed in previous extensions. The second aim is to prove that even these behaviours are so complex; they can be translated into other low level models (as Coloured Petri Nets [21]) and so be analysed. This translation exploits Dynamic Petri Nets [11] as an intermediary representation between our model and Coloured Petri Nets.
Keywords: Petri Nets , Coloured Petri Nets, Dynamic Petri Nets, Mobile Computing System.

1 Introduction

The development of computer science technologies and increasing user requirements are the major drivers of the birth of sophisticated solutions. Mobility with its soft (code mobility) and hard (device mobility) aspects is one of these solutions. When some disaster menaces a critical system during its execution, it seems a good idea to transfer this system and to save its state to another, more secure site, where it can continue its execution. By soft mobility, we mean a system where code can migrate from one site to another site. Many reasons can cause such migration and many methods and techniques can be used. On the other hand, travelling users who request some computing services need also some specific mobile devices. In this last case, we talk about hard mobility. Applications using code mobility are increasing. Code mobility touches critical domains (military, spacial, medicine). Such domains require that used applications insure a set of properties. Safety, liveness, no deadlock, fault tolerance, and security are example of such required properties. Using formal methods, one can develop systems and proof (or verify) presence or absence for specific properties, in these systems. Formal methods are languages, tools, and approaches allowing specification and verification of systems. Formal languages are based on a well-defined syntax and a formal semantics. Their formal semantics allow developers to

verify specification written in such languages. For some languages, automatic tools are proposed to verify the specifications. Using formal methods in code mobility is not recent. Most currently methods can be considered as derived from processes-algebra [6] or state-transition systems.

As a state-transition model, Petri Nets [18] was proposed to model concurrent and parallel systems. This formalism has a graphic representation and a formal background. Using places, transitions and connecting arcs, this formalism can specify states, actions and transitions between states through which a system evolves. Using Petri nets, one can analyse behavioural or structural properties of a system. To model mobility with Petri nets, the most important contribution can be found in high level Petri Nets. Many extensions have been proposed to adapt Petri nets to mobile systems: Mobile Nets and Dynamic Petri nets [11], Nested Petri Nets [13], Hyper-Petri-Nets [14], Mobile Synchronous Petri Net [15]...

The first idea that has motivated our work was mobile code systems. In these systems, type of resources and their bindings play a central role in the migration procedure. Resources decide also the success or failure of the process. Proposed formal methods founded in the literature do not deal with these aspects and their problems. In our first work, we have proposed a *naive version* of "Labelled Reconfigurable Nets" [23] extended to "Coloured Reconfigurable Nets" in [24]. Our objective was to propose a graphical tool to model mobile code systems in an easy and intuitive way. In these works, we were interested to provide formalisms that model mobility, explicitly (The mobility is modelled through the reconfiguration of the net's structure when some transitions are fired). When trying to offer this quality in a model, we have to deal with the problem of interpreting this reconfiguration formally. In [23, 24], we have introduced specific labelled transitions: "reconfigure-transitions", which reconfigure the structure of the net when they are fired. The first drawback of this solution is that we must provide a specific treatment of these transitions when the model is analysed. In [25], we have proposed an interpretation of Reconfigurable Labelled Nets into a high order Maude (Reconfigurable Maude). The idea was to extend Maude [21] with some "reconfigure rewriting rules". These rules can represent the reconfigure-transitions. Reconfigurable Maude can be used to simulate Reconfigurable Labelled Nets. In the current work, we will present another version that we call Extended "Labelled Reconfigurable Nets", where the label of a reconfigure-transition is defined as a tuple. This tuple contains a set of values which belong to different types. Three specific types are defined: P (for places), T (for transitions), and A (for arcs). These types will contain signed (negative as well as positive) objects (places, transitions, and arcs). The presence of a positive object (resp. negative object) in a label of a reconfigure-transition can be the cause to add (resp. delete) this object to (resp. from) the structure of this net. Using these labels, the structure of the net can be expanded, reduced, or destroyed. Our second *contribution* is the proposition of a method to analyse this model, using Dynamic Nets [11]. Dynamic Nets can be translated into CPN [21], and can be analysed using CPN-tool [26]. We propose the encoding of our model into Dynamic nets. This encoding will be used to analyse this model. This encoding *has been proved*. The encoding and its proof are not presented in this paper.

This paper is organized as follows: The section two presents the formal definition of Extended Labelled Reconfigurable Nets (ELRN), its semantics, and an example of modelling. Section three discusses the analysis issue that we have developed for the analysis of ELRN using a translation of the model into Dynamic Nets [11]. Section four presents a comparison between our work and other similar works, and finally, section five will conclude this paper.

2 Extended labelled reconfigurable nets

Extended Labelled Reconfigurable Nets (ELRN) are an extension of Coloured Petri Nets [21]. In ELRN, the set of transitions is divided into two subsets: *ordinary transitions* (OT) and

reconfigure-transitions (RT). A reconfigure-transition has a label (a tuple of values). The firing of an ordinary transition will change the marking of the Net in an ordinary manner, as in CPN [21]. The firing of a reconfigure-transition changes the marking of the net as well as the structure of the Net. A reconfigure-transition changes the structure of the net, by adding or deleting a node (places, transitions, arcs). To add a place, a reconfigure-transition must have a label which contains: (the name of the place, its initial marking, its input (resp. output) transitions with their incoming (resp. outgoing) expressions). To add a transition, a reconfigure-transition must have a label which contains: (the name of the transition, its guard, its input (resp. output) places with their incoming (resp. outgoing) expressions). Finally, to add an arc, a reconfigure-transition must have a label which contains: (the name of the arc, its labelling expression, and its input/output nodes). Names of nodes (places, transitions, arcs) can be preceded by a negative sign. The presence of negative node in a label causes its elimination from the net (iff it existed in the original net), once the reconfigure-transition is fired. In the following subsection, we present the formal definition of this model, its semantics, and a modelling example.

2.1 Formal definition

Let $Name$ be a set of names. Let P, T, and A be three finite and disjoint subsets of the set $Name$.

An Extended Labelled Reconfigurable Nets $N_{P,T,A}$ is a 10-tuple (Σ, $P\prime$, $T\prime$, $A\prime$, C, G, E, I, L), where:

- Σ: a set of types (Colours). We denote by Σ^* the set of all multi-sets of the set Σ;
- $P\prime$: a set of places; $P\prime \subseteq P$.
- $T\prime$: a set of transitions; $P\prime \subseteq P$. $T\prime = OT \cup RT$ (OT for ordinary transitions, and RT for reconfigure-transitions).
- $A\prime$: a set of arcs. $A\prime \subseteq (T\prime \times P\prime) \cup (P\prime \times T\prime)$. For a place p in $P\prime$ and a transition t in $T\prime$, we can have an arc a in $A\prime$ written $a = (p, t)$ (resp. (t, p)), if it connects p to t (resp. if it connects t to p). We write $(p, .)$ (rep $(t, .)$) to denote the set of arcs that start from p (resp. to denote the set of arcs that start from t).
- C: a colour function associated with each place. $C : P\prime \to \Sigma$. For each place p, C associates a unique colour (type) $C(p)$;
- G: a guard function associated with each transition. $G : T\prime \to Exp$. Where Exp is the set of all Boolean expressions that can be constructed using constants and variables defined in types Σ;
- E: an expression function that associates to each arc a in $A\prime$ an expression $E(a)$. The expression $E(a)$ is a multi-set of $C(p)$ where $a \in (p, .)$.
- I: is an initial state of the net. I$=< M_0, S_0 >$, where M_0 is the initial marking of places P'. $M_0 : P\prime \to \Sigma^*$. S_0 is the initial structure of the net. We take $S_0 = P\prime \cup T\prime \cup A\prime$.
- L: a labelling function which associates to each transition rt in RT a label. We denote by $(P, Exp)^*$ (resp. $(T, Exp)^*$) the set of couples composed of a place in P and an expression (resp. composed of a transition in T and an expression). We denote by P^- the set of names defined in P preceded by a negative sign (idem for T^- and A^-). We denote by ϵ the empty untyped element. A label is a tuple of values. Three kinds of labels can be used:(i) ($P \cup P^-$, $\Sigma^* \cup \{\epsilon\}$, $(T, Exp)^* \cup \{\epsilon\}$, $(T, Exp)^* \cup \{\epsilon\}$) labels a transition which adds a place

to the net, (ii) ($T \cup T^-$, $Exp \cup \{\epsilon\}$, $(P, Exp)^* \cup \{\epsilon\}$, $(P, Exp)^* \cup \{\epsilon\}$) labels a transition which adds a place to the net, and (iii) ($A \cup A^-$, $\Sigma^* \cup \{\epsilon\}$, $Name \cup \{\epsilon\}$, $Name \cup \{\epsilon\}$) labels a transition which adds an arc to the net. So, we have: $L : RT \rightarrow (P \cup P^- \times \Sigma^* \cup \{\epsilon\} \times (T, Exp)^* \cup \{\epsilon\} \times (T, Exp)^* \cup \{\epsilon\}) \bigcup (T \cup T^- \times Exp \cup \{\epsilon\} \times (P, Exp)^* \cup \{\epsilon\} \times (P, Exp)^* \cup \{\epsilon\}) \bigcup (A \cup A^- \times \Sigma^* \cup \{\epsilon\} \times Name \cup \{\epsilon\} \times Name \cup \{\epsilon\})$.

2.2 Semantics

Let N be an Extended Labelled Reconfigurable Nets, and t a transition in T. As in CPN (Coloured Petri Nets) [21], we denote by $^\circ t$ the set of input places for the transition t, and by t° the set of output places for the transition t. Let $I_0 =< M_0, S_0 >$ be the current state of N. Firing t changes I_0 towards $I_1 =< M_1, S_1 >$. We denote this as: $< M_0, S_0 > \rightarrow^t < M_1, S_1 >$. In case of t in OT, we have: $S_1 = S_0$.

Preconditions to fire t

A binding β is a function that assigns some values to some variables. We denote by $E(p,t)[\beta]$ a binding in which every variable in E is assigned to some values depending on β. Now, the transition t can be fired iff there is a binding β on the variables of $E(p,t)$ such that $M_0(p) \leq E(p,t)[\beta]$, for each $p \in^\circ t$, and $G(t)$ is true.

Post-conditions of firing t

After the firing of t, N will transit from its current state I_0 to another state $I_1 =< M_1, S_1 >$. For each p in $^\circ t$, we will have: $M_1(p) = M_0(p) - E(p,t)[\beta]$. For each $p \in t\circ$, we will have: $M_1(p) = M_0(p) + E(t,p)[\beta]$.
If $t \in OT$ then $S_1 = S_0$. If $t \in RT$ then: S_0 (which is $P\prime_0 \cup T\prime_0 \cup A\prime_0$) will be updated to $S_1 = P\prime_1 \cup T\prime_1 \cup A\prime_1$. Three cases are possible:

- t changes $P\prime$:
  - by adding a place p: the label of t must be: (p, m_p, $\{(in\_t_1, in\_e_1), ..., (in\_t_n, in\_e_n)\}$, $\{(out\_t_1, out\_e_1), ..., (out\_t_l, out\_e_l)\}$); where: m_p is the initial marking of p, and (in_t_i, in_e_i) (resp. (out_t_j, out_e_j)) an input (resp. an output) transition with its incoming (resp. outgoing) expressions. So, $P\prime_1 = P\prime_0 \cup \{p\}$, $T\prime_1 = T\prime_0$, and $A\prime_1 = A\prime_0$.
  - or deleting the place p: the label must be ($-p$, ϵ, ϵ, ϵ). So, $P\prime_1 = P\prime_0 \setminus \{p\}$, $T\prime_1 = T\prime_0$, and $A\prime_1 = A\prime_0$.
- t changes $T\prime$:
  - by adding the transition at: the label of t must be: (at, g_{at}, $\{(in\_p_1, in\_e_1), ..., (in\_p_n, in\_e_n)\}$, $\{(out\_p_1, out\_e_1), ..., (out\_p_l, out\_e_l)\}$); where: g_t is a guard, and (in_p_i, in_e_i) (resp. (out_p_j, out_e_j)) an input (resp. an output) places with its incoming (resp. outgoing) expressions. So, $P\prime_1 = P\prime_0$, $T\prime_1 = T\prime_0 \cup \{at\}$, and $A\prime_1 = A\prime_0$.
  - or deleting the transition dt: the label of t must be: ($-dt$, ϵ, ϵ, ϵ). So, $P\prime_1 = P\prime_0$, $T\prime_1 = T\prime_0 \setminus \{dt\}$, and $A\prime_1 = A\prime_0$.
- or t changes $A\prime$:

- by adding the arc $a = (p, t\prime)$: the label must be $(a, e_a, p, t\prime)$; where: e_a is a labelling expression, p the name of a place, and $t\prime$ the name of a transition. So, $P\prime_1 = P\prime_0$, $T\prime_1 = T\prime_0$, and $A\prime_1 = A\prime_0 \cup \{a\}$.
- by adding the arc $a = (t\prime, p)$: $(a, e_a, t\prime, p)$; where: e_a is a labelling expression, p the name of a place, and $t\prime$ the name of a transition. So, $P\prime_1 = P\prime_0$, $T\prime_1 = T\prime_0$, and $A\prime_1 = A\prime_0 \cup \{a\}$.
- or deleting the arc a: the label of t must be $(-a, \epsilon, \epsilon, \epsilon)$. So, $P\prime_1 = P\prime_0$, $T\prime_1 = T\prime_0$, and $A\prime_1 = A\prime_0 \setminus \{a\}$.

2.3 A modelling example

In the example of Fig 1, we make an explicit subdivision of the net into a set of sub-blocs. Each sub-bloc can represent an agent or a site where many agents reside. Each sub-bloc has a title presented on the top of this sub-bloc. We use a specific graphical representation for the reconfigure-transitions, to distinguish them from the ordinary transitions. In Fig 1, we have three agents. $Agent_1$ and $Agent_3$ are immobile agents which existed on two different sites (S_1 and S_2). $Agent_2$ is a mobile Agent, which is located initially on the site S_1. On the site S_1, $Agent_2$ communicates with $Agent_1$ through the communication place C_1. $Agent_2$ receives an information (of a some type that we denote: **Information**) from C_1, then it moves towards the site S_2, where $Agent_3$ is located. On the site S_2, $Agent_2$ passes the information received from $Agent_1$ to the $Agent_3$, through the place C_2. To do the transfer of this information, a new arc (t_{22}, C_2) must be added to the model, and the arc (C_1, t_{21}) must be deleted from the model. To do this reconfiguration in the model, we use the two reconfigure-transitions: rt_1 and rt_2, with two labels: $L_1 =< -(C_1, t_{21}) >$ and $L_2 =< (t_{22}, C_2), t_{22}, C_2, X >$. X is the variable which transfers the information to $Agent_3$. Fig 2 shows the system after the movement of the $Agent_2$, where the system is reconfigured.

The initial marking of the places is $\{M_0(P_{11}) =< inf >, M_0(P_{21}) =< \bullet >\}$, where inf is a data of the type **Information** and $< \bullet >$ represents the constant black-token of the type **Black-token** (A type which contains only one value which is $< \bullet >$). The variable X is also of type **Information**. The non-labelled arcs are implicitly labelled $< \bullet >$.

3 On the analysis of ELRN

Our aim is to offer a way to analyse ELRN models. We have proposed that the analysis of ELRN can be done through the analysis of some equivalent models in CPN (Coloured Petri Nets [21]) or PN (Petri Nets [18]). Petri Nets and Coloured Petri Nets have been studied for many years and have many automatic verification-tools [22]. To profit from these tools, we must show that there is some correct transformation (an *Unfolding*) from the ELRN formalism (as high level nets) towards CPN (and PN). The transformation of ELRN directly into CPN or PN is a hard task; so we propose to prove that the Extended Labelled Reconfigurable Nets (ELRN) models can be encoded into Dynamic Nets (DN) [11]. We use the DN as an intermediary pass between ELRN and CPN.

At this stage, the obtained specification can be transformed into CPN (Coloured Petri Net Model) model and so, be analysed using CPN-tool for example. The unfolding of ELRN models into DN is developed and has been proved.

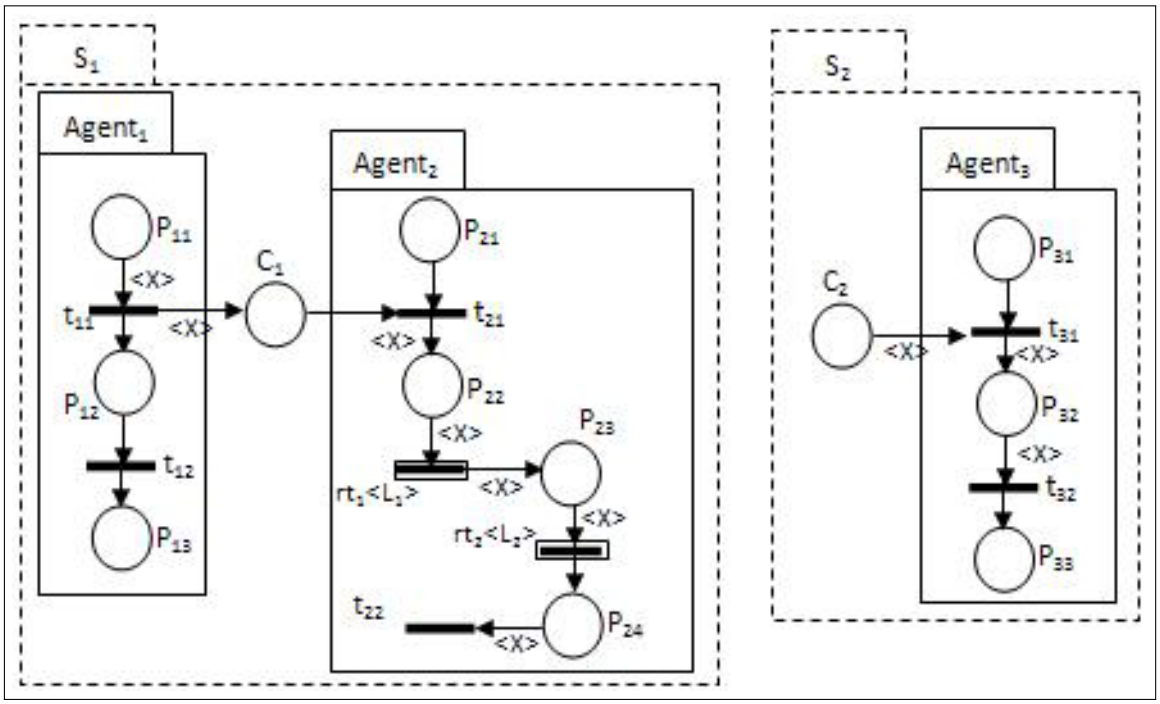


Figure 1: Example of an ELRN.

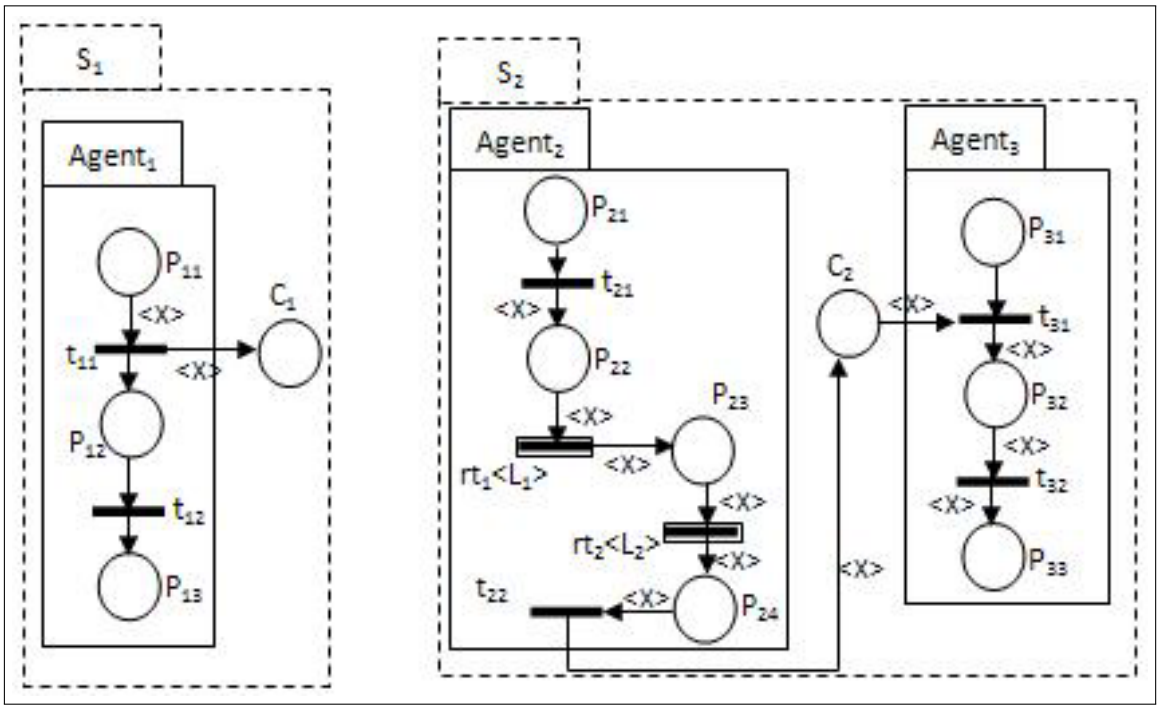


Figure 2: Example of an ELRN (after reconfiguration).

4 Related works

Research on extending Petri Nets (to model systems, with dynamic structure) has provided some remarkable results. We can distinguish between extensions that model mobility in an implicit way (no modification in the structure of the net), or in an explicit way (the reconfiguration, of the net structure, models mobility). The most important propositions are dedicated to mobile systems and mobile agents. In PrN (Predicate/Transition nets) [16], mobile agents are modelled through tokens. These agents are transferred by transition firing from an environment to another. In this work, the structure of the net does not change. The agents are represented as token, so this abstraction does not allow representing some complex behaviour of this kind of agents. In [15], authors proposed MSPN (Mobile synchronous Petri net) as formalism to model mobile systems and security aspects. They have introduced the notions of nets (an entity) and disjoint locations to explicit mobility. A system is composed of set of localities that can contain nets. To explicit mobility, specific transitions are introduced. Two kinds of specific transitions were proposed: new and go. Firing a go transition moves the net from its locality towards another locality. The destination locality is given through a token in an input place of the go transition. In this work, mobility is not also explicit. Mobility is implicitly modelled by the activation of some nets and the deactivation of other nets, using tokens. Migration of an agent is modelled by the deactivation of the net modelling this agent in a locality and the activation of the net that represents this same agent in the destination locality. So, this is a kind of simulation of mobility. In nested nets [21], tokens can be Petri nets themselves. This model allows some transition when they are fired to create new nets in the output places. Nested nets are hierarchic nets where we have different levels of details. Places can contain nets, and these nets can contain also nets as tokens in their places etcetera. So all nets created when a transition is fired are contained in places. So the created nets are not in the same level with the first net. This formalism is proposed to adaptive work-flow systems. Self Modifying Nets (SMN) [18] is an extension of Petri Nets. In this formalism, edges can be labelled by names of places. If the name p is used as the weight of an arc, then this means that the number of tokens to be moved through this arc is equal to the current marking of the place p. So, in self modifying nets, the weights of arcs are dynamic. These weights depend on the current marking of the net. Even SMN offers more computational power than PN; mobility was not the objective of this extension. Though, this formalism was the basis for other more important formalisms like "Reconfigurable Nets" [10]. In "Reconfigurable Net" [10], the structure of the net is not explicitly changed. No places or transitions are added in runtime. The key difference with coloured Petri nets is that firing transition can change names of output places. Names of places can figure as weight of output arcs. This formalism is proposed to model nets with fixed components but where connectivity can be changed over time.

In Object Petri Nets [21], and Elementary Object Nets (EON) [12], tokens can be Petri nets themselves. In an EON, we distinguish between System Nets and Object Nets. Object Nets play the role of object tokens that can appear in places of a System Net. Here, a two-level system modelling technique is introduced. The System Net which represents the external level and the Object Net which represents the internal level can have some synchronous transitions. In this case, these transitions must be fired simultaneously in the two levels. Object Nets used as tokens in a System Net can also interact. EON formalism was proposed to model some kind of systems like: work-flow, flexible manufacturing, and mobile agents. Based on the EON formalism, other proposals were done: Nested Nets [13], Petri Hypernets [14], Nets within Nets [19], etc In Nested Nets [21], firing some transitions creates new nets (called token nets) in their output places. Nested nets are also hierarchic nets, where we have different levels of details. Places can contain nets, and these nets can contain also nets as tokens in their places. This formalism was proposed to adaptive work-flow systems. Adaptive means an ability to modify processes in a structured

way, for example by replacing a sub-process or extending it. Petri Hypernets [14] are proposed to model mobile agents. Mobile agents are modelled as nets. These mobile agents are manipulated by other agents (modelled as nets) who can be also mobile. We call Open Net a net used to model a mobile agent. This open net plays the role of a token in another net; this last one is called hyper-marking Net. As a difference with Valk's proposal [12, 20], the inter-level synchronization in Hyper-Nets is achieved solely by means of exchanging messages. In [17], PEPA nets are proposed, where mobile code is modelled by expressions of the stochastic process algebra PEPA which play the role of tokens in (stochastic) Petri nets. The Petri net of a PEPA net models the architecture of the net, which is a static one. Mobile Petri nets (MPN) [11] extend coloured Petri nets to model mobility. MPN is inspired by join-calculus [4]. The output places of transition are dynamic. The input expression of a transition defines the set of its output places.

In all these formalisms, the structure of the net is not changed and mobility is modelled implicitly through the net's dynamic. In these models, an important work is required from the modeller to model mobility implicitly. MPN are extended to Dynamic Petri Net (DPN) [11]. Mobility in DPN is modelled explicitly, by adding subnets when transitions are fired. However, the Dynamic Petri nets formalism implies some constraints: (i) No transition without input places, (ii) Added nets, to the original net, must not modify the input of an existing transition in the original net, (iii) We can not add a connection between two disconnected existing nodes, (iv) and we cannot delete nodes (place, transition or connection).

In this paper, we have proposed an extension for Petri nets that can be used to model mobility (and in general, reconfigurable systems): Extended Labelled Reconfigurable Nets. Extended Labelled Reconfigurable Nets is more flexible and more expressive and does not imply constraints on the dynamic of the structure. We consider that Extended Labelled Reconfigurable Nets can be used by reconfigurable systems developers with more flexibility than other formalisms. This is due to the feature that it models mobility explicitly through mobility of nodes in the Labelled Reconfigurable Net. Developers can encode mobile aspects of their system directly and explicitly in the EARN formalism.

The power of Petri nets resides in its verification methods. When extending Petri nets, we reach some formalism with a high expressiveness, but the analysis becomes more complex or even impossible. Developers of new formalisms must propose analysis techniques. Mostly, they are proposing some translation (or encoding) of their formalisms into some well-known formalism or approach in modelling domain. Such translation allows the analysis of the new formalisms models using techniques of well-known formalisms. The most famous encoding can be found in the unfolding of Petri nets into automaton to apply model-checking, and then the unfolding of CPN [21] (Coloured Petri Nets) into PN [18] (Petri Nets) to analyse some properties that are not analysed on the CPN directly. We can find other works, in literature. In [22], author authors studied equivalence between the join calculus [4] and different kinds of high level nets. They proved the equivalence between Reconfigurable nets (RN) [10] (an extension version of PN) and the join calculus. This equivalence allows to interpret RN into join calculus and to verify those using join-calculus tools. In [19], Petri nets are translated into linear logic programming. This translation can be used to analyse Petri nets using Prolog model-checker. Authors of [20] encoded Synchronous mobile nets (SMN) [15] into rewriting logic [22]. This encoding allows the use of Maude [21] to verify SMNs specifications.

In this paper, we have discussed an encoding of ELRN behaviours into Dynamic nets [11]. This encoding was proved to be correct. The advantage of such encoding resides in the possibility to encode Dynamic nets into CPN (coloured Petri Nets). So, ELRN can be translated into CPN. Once translated into CPN, ELRN nets can be analysed using CPN verification tools.

5 Conclusion

Mobile Systems are systems with a dynamic structure. Their structure changes as they are executed. This class of systems can be found in many domains of our life. Mobile robots used to explore hostile environment, mobile agents used in the internet or in distributed systems, mobile nodes in a mobile wireless networks ... All these systems can be considered as reconfigurable systems. The use of these systems is in expansion for many reasons: their efficiency, their abstractions for the designer, their flexibility ...These characteristics make these systems in the kernel of many critical systems: aeronautics, military, medicine, commerce... The design of these systems becomes a critical activity. Their reliability and their correctness are crucial. To ensure the correctness of these systems, formal methods seem to be an adequate solution. Using formal methods, the designer specifies the system in a formal language. A formal language has a well defined syntax, and formal semantics which allows the verification of properties of the designed system. We found in the literature, many formal methods. Classical formal methods (proposed for classical systems) are well defined and are mature. However, these classical formal methods have not the expressiveness to specify reconfigurable systems. The use of the classical methods makes the designer's task a hard task. Extended versions are proposed to deal with the idea of reconfigurable systems. In the literature, we can find two principal classes: Processes algebra based methods, and state-transition based methods.

State-transition based methods can be found in extensions of Petri nets model. Petri nets are an elegant model for concurrency. With its graphical representation and its formal background, it was used to specify and verify concurrent multi-processes systems. The classical model has not the power of expressiveness to deal with current aspects such as mobility. To take benefits from the power of the model in mobility domains, several works have been proposed. These works try to extend Petri nets with the same ability to specify mobility (and more generally: reconfigurability).
In this paper, we have presented the Extended Labelled Reconfigurable Nets formalism. The formal definition of this formalism, its semantics and a modelling example are presented. The encoding of this formalism into another formalism Dynamic Nets [11] was proved using and offers a method to do the analysis of this model.

As perspectives of the current work, we propose the below axes as open domains:

- The experimentation of ELRN in the modelling of mobile systems: Mobile agents systems, mobile networks, ... This modelling work can prove the power of our formalism and shows its shortcomings and so allow us to introduce necessary adaptations;

- The work on automatic verification: The translation of dynamic nets into coloured Petri nets is presented in [11]. We are working on the development of a tool-kit to implement this translation. The encoding presented in section three is formal and proved to be correct; so it is possible to think of an implementation of this last encoding also;

- In the current time, complexity and decidability issues are not yet studied. These aspects are important, once a new formalism is proposed. These issues will be also developed in our future works.

Bibliography

[1] D. Sangiorgi and D. Walker (2001); *The π-Calculus: A Theory of Mobile Processes*, Cambridge University Press.

[2] F. Cédric, G. Gonthier (2000); The Join Calculus: a Language for Distributed Mobile Programming, *Applied Semantics*, International Summer School, APPSEM 2000, Caminha, Portugal, Sept. 2000, LNCS 2395, 268-332.

[3] J.C.M. Baeten (2003); Over 30 years of process algebra: Past, present and future, in L. Aceto, Z. Ésik, W.J. Fokkink, and A. Ingólfsdóttir, editors, *Process Algebra: Open Problems and Future Directions*, vol. NS-03-3 of BRICS Notes Series, 7-12.

[4] F. Cédric, G. Gonthier, J. J. Lévy, L. Maranget, D. Rémy (1996); A calculus of mobile agents, *Proc. 7th International Conference on Concurrency Theory (CONCUR'96)*, 406-421.

[5] E. Badouel, O. Javier (1998); Reconfigurable Nets, a Class of High Level Petri Nets Supporting Dynamic Changes within Workflow Systems, Rapports de recherche INRIA, ISSN 0249-6399.

[6] A. Asperti, N. Busi (2009); Mobile Petri Nets, Technical Report UBLCS-96-10, Department of Computer Science University of Bologna, *Mathematical Structures in Computer Science journal*, 19 (6): 1265-1278.

[7] R. Valk (1998); Petri Nets as Token Objects: An Introduction to Elementary Object Nets, *Applications and Theory of Petri Nets*, LNCS vol. 1420, 1-25.

[8] I.A. Lomazova (2001); Nested Petri Nets, Multi-level and Recursive Systems, *Fundamenta Informaticae*, 47(3): 283-293.

[9] M. A. Bednarczyk, L. Bernardinello, W. Pawlowski, L. Pomello (2004); Modelling Mobility with Petri Hypernets, *17th Int. Conf. on Recent Trends in Algebraic Development Techniques, WADT'04*, LNCS vol. 3423.

[10] F. Rosa-Velardo, O.M. Alonso, D. F. Escrig (2005); Mobile Synchronizing Petri Nets: a choreographic approach for coordination in Ubiquitous Systems, *1st Int. Workshop on Methods and Tools for Coordinating Concurrent, Distributed and Mobile Systems*, MTCoord'05. ENTCS 150.

[11] Dianxiang Xu, Yi Deng (2000); Modeling Mobile Agent Systems with High Level Petri Nets, *IEEE International Conference on Systems, Man, and Cybernetics*, 5: 3177-3182.

[12] S. Gilmore, J. Hillston, L. Kloul, M. Ribaudo (2003); PEPA nets: a structured performance modelling formalism, *Performance Evaluation*, 54(2):79-104.

[13] C.A. Petri (1962); KommuniKation mit Automaten, Schriften des IIM Nr.2, Institut fr Instrumentelle Mathematik, Bonn (1962). English translation: Technical Report RADC-TR-65-377, Griffiths Air Force Base, New York, vol. 1, suppl. 1, 1966.

[14] R. Valk, Self Modifying Nets (1978); A Natural Extension of Petri Nets, *Proceeding of ICALP'78, Lecture Notes in Computer Science*, 62: 464-476.

[15] M. Khler, D. Moldt, H. Rlke (2003); Modelling mobility and mobile agents using nets within nets, In W. van der Aalst and E. Best, eds., *Applications and Theory of Petri Nets 2003, Proceeding*, vol. 2679 of LNCS, 121-139.

[16] R. Valk (2004); Object Petri nets: Using the nets-within-nets paradigm, In Jrg Desel, Wolfgang Reisig, and Grzegorz Rozenberg, eds., *Advances in Petri Nets: Lectures on Concurrency and Petri Nets*, vol. 3098 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, New York, 819-848.

[17] K. Jensen (1994); An Introduction to the Theoretical Aspects of Coloured Petri Nets, In J.W. de Bakker, W.-P. de Roever, G. Rozenberg (eds.), *A Decade of Concurrency, Lecture Notes in Computer Science*, 803: 230-272.

[18] http://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/quick.html.

[19] M. Buscemi, V. Sassone (2001); High-Level Petri Nets as Type Theories in the Join Calculus, *Proc. of Foundations of Software Science and Computation Structure (FoSSaCS '01)*, LNCS 2030.

[20] F. Rosa-Velardo (2007); Coding Mobile Synchronizing Petri Nets into Rewriting Logic, *Electronic Notes in Theoretical Computer science*, 174(1): 83-98.

[21] M. Clavel, F. Durn, S. Eker, P. Lincoln, N. Mart-Oliet, J. Meseguer, J. Quesada (1999); Maude: specification and programming in rewriting logic, *SRI International*, http://maude.csl.sri.com.

[22] J. Meseguer (1992); Conditional rewriting logic as a unified model of concurrency, *Theoretical Computer Science*, 96 (1): 73-155.

[23] L. Kahloul, A. Chaoui, Code mobility modeling: a temporal labelled reconfigurable nets, *Proceedings of the 1st International Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications, MOBILWARE 2008*, Innsbruck, Austria, February 13 - 15, 2008. ACM International Conference Proceeding Series 278.

[24] L. Kahloul, A. Chaoui (2008); Coloured Reconfigurable Nets for Code Mobility Modeling, *International Journal of Computers, Communications & Control*, ISSN 1841-9836, Suppl. issue, 3(S): 358-363.

[25] L. Kahloul, A. Chaoui, LRN/R-maude based approach for modeling and simulation of mobile code systems, *Ubiquitous Computing and Communication Journal (UbiCC journal)*, Vol. 3 No. 6, 12/20/2008. http://www.ubicc.org/search_advanced.aspx.

[26] http://wiki.daimi.au.dk/cpntools/cpntools.wiki.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):222-229, April, 2015.

Logging for Cloud Computing Forensic Systems

A. Pătraşcu, V.V. Patriciu

Alecsandru Pătraşcu*
1. Military Technical Academy, Computer Science Department
39-40 George Coşbuc Street, Bucharest, Romania
alecsandru.patrascu@gmail.com
2. Advanced Technologies Institute
10 Dinu Vintila, District 2, 021102, Bucharest, Romania
ati@dcti.ro
*Corresponding author: ati@dcti.ro

Victor Valeriu Patriciu
Military Technical Academy, Computer Science Department
39-40 George Coşbuc Street, Bucharest, Romania
victorpatriciu@yahoo.com

Abstract: Cloud computing represents a different paradigm in the field of distributed computing that involves more and more researchers. We can see in this context the need to know exactly where, when and how a piece of data is processed or stored. Compared with classic digital forensic, the field of cloud forensic has a lot of difficulties because data is not stored on a single place and furthermore it implies the use of virtualization technologies.
In this paper we present a new method of monitoring activity in cloud computing environments and datacenters by running a secure cloud forensic framework. We talk in detail about the capabilities that such system must have and we propose an architecture for it. For testing and results we have implemented this solution to our previous developed cloud computing system.
Keywords: cloud computing; data forensics; logging framework; distributed computing; binary diff

1 Introduction

Cloud Computing to put it simply, means Internet Computing. It is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

The cloud computing model offers the promise of massive cost savings combined with increased IT agility. It is considered critical that government and industry begin adoption of this technology in response to difficult economic constraints. However, cloud computing technology challenges many traditional approaches to datacenter and enterprise application design and management. Cloud computing is currently being used. However, security, interoperability, and portability are cited as major barriers to broader adoption.

In this context, a new need for IT experts is increasing: the need to know exactly how, where and in what condition is the data from the cloud stored, processed and delivered to the clients. We can say with great confidence that cloud computing forensics has become more and more a need in todays distributed digital world.

In this paper we are going to present a new way in which we can integrate a full forensics framework on top of a new or existing cloud infrastructure. We will talk about the architecture behind it and we will present its advantages for the entire cloud computing community. We will present also the impact that our technology proposal will have on existing cloud infrastructures

and as a proof of concept we will present some particular implementation details over our own cloud computing framework that we have already developed in [6].

The rest of the document is structured as follows. In section 2 we present some of the related work in this field, that is linked with our topic and in section 3 we present in detail our proposed cloud forensics logging framework. Section 4 is dedicated to presenting our results from our implementation made so far, and in section 5 we conclude our document.

2 Related work

The integration of cloud computing logging in the field of forensics is not new and we find thesis in this directions, such as the one of Zawoad et al [1] which present an architecture for a secure cloud logging service that collects information from various sources around the datacenter, both software (hypervisors) and hardware (network equipments)in order to create a complete image of the operations done in a datacenter.

The same challenges are evidenced by Marty [2] and Sibiya et al [3]. In their papers they present a perspective over a custom logging framework and talk about the way in which forensics investigators can be provided with reliable and secure data using a standardized way. They propose using a single centralized log collector and processor, in order to save business's and users time.

In order to face the many challenges involving digital forensics in general, but also to take benefits from the opportunities cloud computing is offering, we have to rethink most of the classic network established principles and re-organize well-known workflows, even include and use tools not previously considered viable for forensic use, such as machine learning or large scale computing. Furthermore we must submit to the classic digital forensic main rule and keep all digital evidence intact. All of our investigation is done on a digital copy of the original data.

In our previous research [10] we have also focused on choosing a proper data representation format that will be used between the modules of our framework and between the modules and the central forensic processing core. In the next paragraphs we present a brief comparison of two existing proposals in this directions, that are applicable in our context.

The first one is the "Management metalanguage" [12] proposed by the UnixWare community. Its advantage is that it can be used as a transparent API in the kernel modules as it provides an interface for an external host. The downside is that it needs a lot of auxiliary binary data to be sent in order to re-create the entire picture at the other end, and using it we get quickly a traffic larger than the one that can be obtained by sending only the basic snapshots.

On the other side, the CEE (Common Event Expression) organization [11] proposes a set of specifications using the JSON and XML markup languages for event logging on disk or in transit over a network. These requirements are designed for maximum interoperability with existing event and interchange standards to minimize adoption costs. The advantage of this approach is that CEE expresses its interfaces and does not promote an actual implementation.

After thoroughly analyzing these two proposals we have chosen to use a combination between them, meaning that we want to full details that the management metalanguage encapsulates, under the form of JSON data representation.

3 Logging framework architecture

In the following section we will talk about the top view architecture of our cloud computing enabled forensic system. We will present the main building blocks and modules and then we

focus on the logging sub-system. The entire architecture will follow also the perspective from the forensic investigator part.

3.1 General forensics architecture

The framework presented in this paper has a modular architecture and each of the modules is presented in the following paragraphs in detail. It is also easy to see that the entire framework can be extended with other modules or plugins to support various workloads and even processing elements. In order to have a working platform, we must first introduce the concept of a cloud computing framework. In Figure 1 we can see that the top view of a cloud computing framework contains two main layers: a virtualization layer and a management layer.

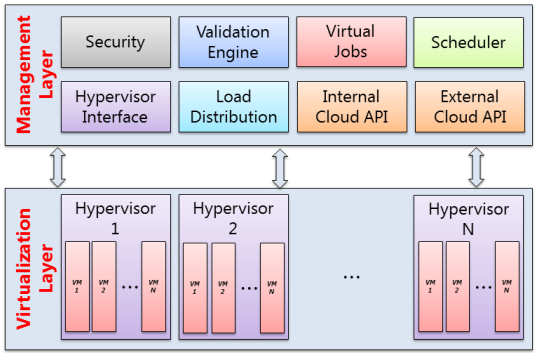


Figure 1: Basic cloud computing architecture

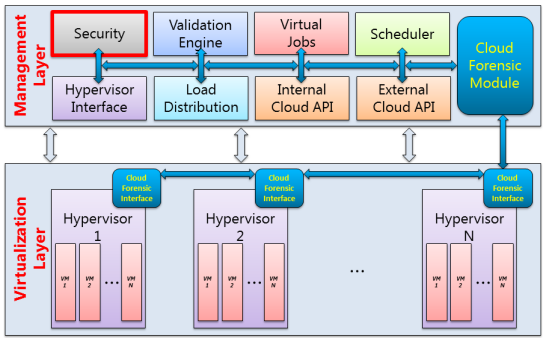


Figure 2: Forensic enabled cloud architecture

In the *Virtualization layer* we find the actual platforms/servers that host the virtual machines and have virtualization enabled hardware. In the *Management layer* we find the modules responsible for enabling the entire operations specific to the cloud. These modules are, in order: **Security** (responsible with all security concerns related to the cloud system - intrusion detection and alarming module), **Validation engine** (receives requests to add new jobs to be processed), **Virtual jobs** (creates an abstraction between the data requested by the user and the payload that must be delivered to the cloud system), **Scheduler** (schedules the jobs to the virtualization layer), **Hypervisor interface** (acts like a translation layer that is specific to a virtualization software vendor), **Load distribution** (responsible with horizontal and vertical scaling of the requests received from the scheduler), **Internal cloud API** (intended as a link between the virtualization layer and the cloud system), **External cloud API** (offers a way to the user for interacting with the system).

Now that the notion of a cloud computing framework was presented, we will talk about the modifications that must be made to it in order to create an forensic enabled cloud computing architecture. As can be seen in Figure 2 the modification affects all the existing modules and includes two new modules, the *Cloud Forensic Module* and the *Cloud Forensic Interface*. Their main goal is to gather all forensic and log data from the virtual machines that are running inside the virtualization layer and it represents the interface between the legal forensic investigator and the monitored virtual machines. The investigator has the possibility to monitor one or more virtual machine for a targeted user for a specific amount of time.

3.2 Cloud logging architecture

In this section we present how our framework is working and how it is created in order to run on top of new or existing cloud computing infrastructures. As example for it we will

present the integration with our previously implemented Cloud Computing framework. The cloud architecture presented in our previous work makes use of the concept of leases, in which we can specify the amount of time the job must run, or specify between what hours in a day it is running.

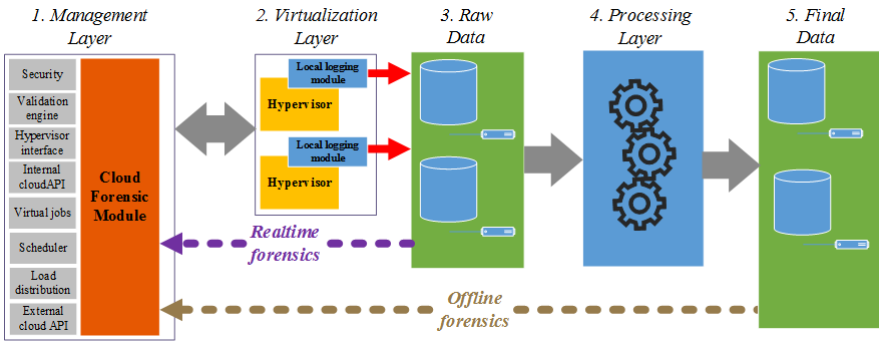


Figure 3: Cloud Forensics Logging Framework

The cloud logging system architecture is a layered one, containing five layers, each with its own purpose. We will present each of them in detail in the following paragraphs. We also used a graphical representation, in Figure 3, where the whole layers and the relationship between them can be seen. The layers are all implemented using the distributed computing paradigm and they represent jobs in our cloud computing environment.

The first layer, as presented in our previous work [4], represents the *management layer* in a cloud computing deployment. All the modules that are responsible with all cloud specific operations can be found at this level, together with the forensic targeted ones, such as "Cloud Forensic Module".

The second layer represents the *virtualization layer* in a cloud computing implementation. At this level we can find the workstations and servers that host the virtual machines. The fact that the main building blocks are represented by the virtual machines, the hardware must also have virtualization enabled. Inside the *Cloud Forensic Interface*, a dedicated "Local logging module" must be installed into the existing physical machine. It is responsible with the RAW data gathering from the monitored virtual machines. The data quantity can be adjusted by the investigator and he can choose to monitor a particular virtual machine or monitor the entire activity existing inside that machine.

In order to gather data reliably from the virtual machines the local logging module must be integrated fully with the running hypervisor inside the physical machine. In this paper we focus on the integration with the "KVM" virtualization technology that exists in modern Linux kernel releases. We have chosen it because it is a full open-source virtualization solution, integrated with the Linux kernel since 2007 and it is actively used by many companies across the world.

An important thing that must be taken in consideration is *what* data are we intercepting from the virtual machine and send it to further processing. Since all the activity can be intercepted, there is the risk of severe time penalties and processing speed. In order to solve this problem, at this point we will offer the possibility for an investigator to choose the logging level for a certain virtual machine. This is helpful considering that, for example, an investigator only wants to analyze the virtual memory for its contents, and it is not interested in virtual disk images or virtual network activity. Also at this step we must consider the problem of network transmission overhead.

The third layer represents a storage layer for the RAW data sent from the local logging modules existing in the virtualization layer. The logging modules will send RAW data, in the

form they are gathered from the hypervisor. Thus, this layer has the function of a distributed storage and it contains a series of nodes, each running a database. We have chosen this approach in order to create a flexible and scalable layer architecture that can face the data traffic coming from the upper layer.

Since the data that is going to be sent from the physical virtualization host to the central forensic management unit can reach important size, we will implement a mechanism of *"diff"* between two pieces of data. For example, if an investigator will want to analyze a virtual machine memory over a period, the local forensic module will sent only one initial memory snapshot and after that only what has been changed will be sent. Of course we can use the full potential of the host and provide a local aggregation module that will pre-process the data collected before sending it to the central forensic module. This approach is new to the field of cloud computing forensics and we consider it a great way to reduce the impact over the network.

The process will run in the following manner. Initially the logging modules will send a reference file and then, at an user defined time period, the modules will send a delta file, that represents the difference between the previous reference file and the current state. Thus, it will implement a snapshot mechanism at the hypervisor level. We have chosen this approach because we want to offer to the forensic investigator the possibility to have an image of what is happening inside a virtual machine between two snapshots. This feature is currently not available in other hypervisors, such as VMware's; in their case we can have a snapshot at time t_0 and one at time t_i, but we cannot know the state of the virtual machine between the *0* and i step.

This layer has also another purpose. In case of extreme emergency, the forensic investigator can "see" a real-time evolution of the monitored virtual machine by issuing a direct connection to this layer. This feature is made available through the Cloud Forensic Module, which has the ability to by-pass normal RAW data processing.

The fourth layer has the purpose of analyzing, ordering, processing and aggregating the data stored in the previous layer. Since all these steps are computing intensive, the entire analysis process will be made in an offline manner and will be available to the investigators as soon as the job is ready. After this entire process the investigator will have a full image of what happened over the monitored remote virtual machine in a manner such as the one encountered in software source code version tools, thus permitting him to navigate back and forth into the history of the virtual machine.

This layer is implemented also as a distributed computing applications. We have chosen this approach due to the processing power needs that our framework demands, more exactly it needs to do correlations between different snapshots in a fair amount of time.

Finally, **the fifth layer** represents the storage of the results published by the previous layer. An forensic investigator will interact with the monitored virtual machine snapshots at this layer, by using the Cloud Forensic Module from the Management layer.

4 Results

In this section we are going to present details regarding the results collected after the implementation of our Cloud Logging modules.

4.1 Network configuration

For testing, the modules have been implemented and split across multiple workstations, as can be seen in Figure 4.

They are represented as a cluster of servers, each having the functionality presented in detail in the architecture section. As it can be seen, the entire modules found in the dotted perimeter,

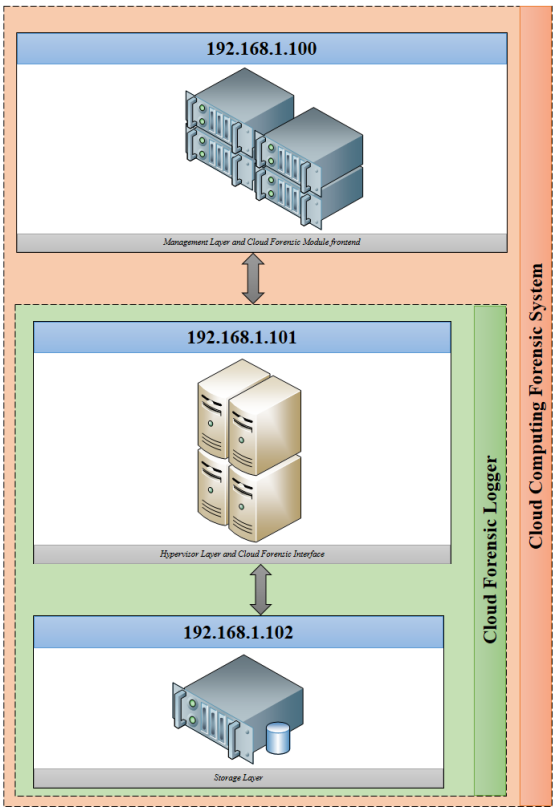


Figure 4: Mapping modules to workstations.

called "Cloud Computing Forensic System", can also be ran all on one workstation. Elements like network switches are not represented in order not to burden the graphic, but the IP address of the hosts are kept. In our configuration we have used three distinct workstation, each having the functionalities and network addresses presented in the figure.

The hardware platform used was composed from an AMD Phenom II X6, 6 cores, 8GB RAM, RAID0 configured hard-disks running KVM as hypervisor and QEMU as a hypervisor interface, an Intel DualCore, 4GB RAM as the storage layer and an AMD C-60 DualCore, 4GB RAM as the management layer. The network used is 10/100 MB.

4.2 Experimental results

The experiments were made using KVM as a hypervisor and QEMU and libvirt as drivers for the hypervisor. The tests had the target set on the virtual machine used memory (RAM snapshot) and the virtual machine storage (DISK snapshot).

The process of recording the virtual machine activity was made over a period of several hours, at a time step of 10 minutes. The CPU load when conducting records using all the 6 cores was about 20%. The results are interesting, if we take in consideration the technologies used internally by KVM. For example, RAM snapshots are made entirely from host machine RAM and do not contain necessarily consecutive RAM location. Nevertheless, in our experiments the RAM snapshots were the largest, reaching even gigabytes in size.

Bellow you can see the actual RAM tests that were made. We have split the tests in two distinct zones, one up to 100 MB and one after this barrier. Table 1 and Figure 5 presents the data collected from our modules and the time needed to process it. The transfer time between

the Cloud Forensic Interface module and the Storage module is not taken in consideration, as being a constant time, of about 82 seconds for a 800 MB file. Table 2 and Figure 6 presents the data collected from our modules and the time needed to process it.

Table 1: Tests up to 100 MB in size

Size (KB)	Time (ms)
4	296
454	517
1227	1136
5505	4929
10813	8000

Table 2: Tests over 100 MB in size

Size (KB)	Time (ms)
108036	58982
740032	401156
4251346	2277855

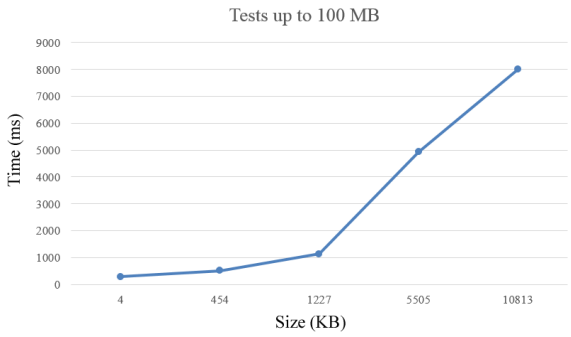


Figure 5: Tests up to 100 MB in size.

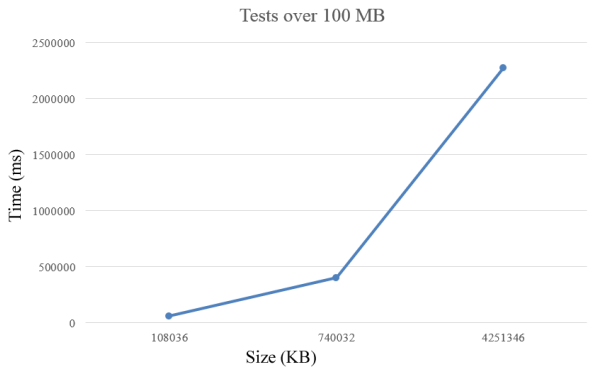


Figure 6: Tests over 100 MB in size.

5 Conclusion

As we have seen in this paper, the topic of cloud computing forensics is very large and poses great challenges in the field of logging. Due to the fact that together with incident response they represent a new field for research, more and more scientists are trying to develop new methods for assuring security in cloud systems. Furthermore, as the environment is purely distributed offers new fields of development much larger than a regular workstation.

In this paper we presented a novel solution that provides to the digital forensic investigators a reliable and secure method in which they can monitor user activity over a Cloud infrastructure. Our approach takes the form of a complete framework on top of an existing Cloud infrastructure and we have described each of its layers and characteristics. Furthermore, the experimental results prove its efficiency and performance.

Acknowledgment

This paper has been financially supported within the project entitled "Horizon 2020 - Doctoral and Postdoctoral Studies: Promoting the National interest through Excellence, Competitiveness and Responsibility in the Field of Romanian Fundamental and Applied Scientific Research", contract number POSDRU/159/1.5/S/140106. This project is co-financed by European Social Fund through the Sectoral Operational Programme for Human Resources Development 2007 - 2013. Investing in people!

Bibliography

[1] S. Zawoad, A.K. Dutta and R. Hasan (2013); SecLaaS: Secure Logging-as-a-Service for Cloud Forensics, in *ACM Symposium on Information, Computer and Communications Security*, DOI: 10.1145/2484313.2484342, 219-230.

[2] R. Marty (2011); Cloud Application Logging for Forensics, *Proceedings of the 2011 ACM Symposium on Applied Computing*, 178-184.

[3] G. Sibiya, H. Venter, T. Fogwill (2012); Digital forensic framework for a cloud environment, *Proceedings of the 2012 Africa Conference*, 1-8.

[4] A. Pătraşcu and V. Patriciu (2013); Beyond Digital Forensics. A Cloud Computing Perspective Over Incident Response and Reporting, *IEEE International Symposium on Applied Computational Intelligence and Informatics*, 455-460.

[5] B. Grobauer and T. Schreck (2010); Towards incident handling in the cloud: challenges and approaches, *Proceedings of the 2010 ACM workshop on Cloud computing security workshop*, New York, DOI: 10.1145/1866835.1866850, 77-86.

[6] A. Pătraşcu, C. Leordeanu, C. Dobre and V. Cristea (2012); ReC2S: Reliable Cloud Computing System, *European Concurrent Engineering Conference*, Bucharest, 1-9.

[7] M. Simmons and H. Chi (2012); Designing and implementing cloud-based digital forensics, *Proceedings of the 2012 Information Security Curriculum Development Conference*, 69-74.

[8] T. Takahashi, Y. Kadobayashi and H. Fujiwara (2010); Ontological Approach toward Cybersecurity in Cloud Computing, *SIN'10 Proceedings of the 3rd international conference on Security of information and networks*, DOI: 10.1145/1854099.1854121, 100-109.

[9] NIST SP800-86 Notes, *Guide to Integrating Forensic Techniques into Incident Response*, http://cybersd.com/sec2/800-86Summary.pdf

[10] A. Pătraşcu and V. Patriciu (2014); Logging system for cloud computing forensic environments, *Journal of Control Engineering and Applied Informatics*, 16(1): 80-88.

[11] http://cee.mitre.org/language/1.0-beta1/cls.html

[12] http://uw714doc.sco.com/en/UDI_spec/m_mgmt.html

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):230-237, April, 2015.

Implementing BPMN 2.0 Scenarios for AAL@Home Solution

L. Rusu, B. Cramariuc, D. Benta, M. Mailat

Lucia Rusu
Babeş Bolyai University of Cluj-Napoca
Romania, 400591 Cluj-Napoca, M. Kogălniceanu, 1
lucia.rusu@econ.ubbcluj.ro

Bogdan Cramariuc
IT Center for Science and Technology Bucharest
Romania, 011702 Bucharest, Av. Radu Beller, 25, sector 1
bogdan.cramariuc@ieee.org

Dan Benţa*
Agora University of Oradea
Romania, 410526 Oradea, Piaţa Tineretului, 8
*Corresponding author: dan.benta@univagora.ro

Marinela Mailat
Babeş Bolyai University of Cluj-Napoca
Romania, 400591 Cluj-Napoca, M. Kogălniceanu, 1
marinelamailat@yahoo.com

Abstract: Aging tendency of European population and live longer and independently desire requires AAL solution for particular elders (chronic diseases, disabilities, aso). NITICS project aim is to develop advanced ITC solutions including monitoring and navigational support for indoor to support elderly in their daily activities. This paper offers a BPMN implementation for indoor assistance based on IoT (sensor monitoring) and Activity workflow implementation. Our solution offers an intelligent Care Center solution for caregivers monitoring and elders support.
Keywords: IoT, AAL@Home, BPA, BPM, BPMN, workflow.

1 Introduction

The European population is aging and tends to live longer and independently. This requires for a good quality of life staying at home. European Commission (EC) Ambient Assisted Living Joint Program (AAL JP) offers an opportunity to incorporate the technological progress in communications, Internet-of-Things (IoT) and Artificial Intelligence (AI). Moreover, advances in medical and assistance or caring sciences are increasingly making people with disabilities autonomous and self-sufficient. Advanced ICT services including monitoring and navigational support are needed to support the mobility of elderly and disabled persons in their home during their daily activities [1].

IoT has many definitions, focused on infrastructure, object interconnection, context awareness, communication interoperability, security, privacy or other particular features. IoT means "things having identities and virtual personalities operating in smart spaces using intelligent interfaces to connect and communicate within social, environmental, and user contexts" [2]. SAP definition offered by Stephan Haller, SAP AG, focuses on business processes "a world where physical objects are seamlessly integrated into the information network and where the physical objects can become active participants in business processes". Services are available to interact with these 'smart objects' over the Internet, query and change their state and any information associated with them, taking into account security and privacy issues.

Based on IoT features for addressing elements such as Convergence, Content, Collections (Repositories), Computing, Communication, and Connectivity from the context, this concept could be applied for homecare solutions which needs interconnection between people and things. Ambient Assisted Living (AAL) focused on several areas AAL4persons: AAL@Home, AALon-the-move, AAL@Work and AAL@Community.

Other authors talk about AAL as home care systems (HCS) which are focused on the support of living assistance for people with special needs (elderly, disabled) in their own homes. HCS domain can be divided into three parts: emergency treatment services, autonomy enhancement services, and comfort services. Emergency Treatment covers assistance, detection, prediction, prevention services. These services offer early prediction and recovery from critical conditions, emergency and safe detection and alert propagation of emergencies (sudden falls, heart attacks, strokes, panics, aso) .Autonomy Enhancement is focused on eating, drinking, dress, medication, cooking, cleaning, shopping services. Comfort Services cover all areas that do not fall into the previous categories: logistic, finding things, home automation, social contacts, infotainment, safety [3].

This paper offers a solution for AAL@Home using SmartHome concept as an Assisted Living Technologies (ALT), by offering a BPMN 2.0 implementation which helps to assist elder people. After an introduction, in the second section we deal with business process analysis (objectives and specific features, actors and roles in our project), and then we develop several workflow scenarios using AAL@Home features, based on end-user (elderly and care-givers) requirements, which include ALT and Telecare features. Last section presents conclusions and future work.

2 Related results

There were previously mentioned some AAL solutions for particular elders.

ROBOCARE, is an Italian AAL@Home research project focuses on the development of distributed systems with software and robotic agents for generating active services in environments for humans may need assistance and guidance. This solution allowing vulnerable elderly people to lead an independent lifestyle in their own homes. ROBOCARE has two scenarios: the ROBOCARE Domestic Environment (RDE) and the Health-Care Institution (HCI) scenario. It covers several fields of research: robotic platforms, sensory systems, activity supervision in complex environments, and human-technology interaction [4].

ROBOCARE investigate the integration of robotic, sensory and automated reasoning components into RDE and HCI scenarios. This project offers main aspects related to technology for elderly care: development of the enabling domotic components (intelligent sensors and robotic platforms) for deployment the target scenarios and development functionalities of domotic components (activity supervision and diagnostics) using service-providing software infrastructure [4].

Smart-home architecture was described by Bregman and Korman using interaction of four modules: Central Management Units (CMU), User Interface (UI), Home Equipment and Appliances Interface (HEAI), External Communication Interface (ECI). The CMU has several components: Operating System (SHOS - Smart Home Operating System), the Smart-Home Database (SHDB), AI (Artificial Intelligence) Engine or Home Intelligence (HI) and Application Services (AS) [5]. An implementation of this model was done in www.liorzehome.com using object-oriented approach. Each event is defined as a set of properties: Event Name: a description of the event name, Input Devices with devices inside the eHome that trigger an event, and functions: Event Triggers: that return TRUE when a trigger is invoked, Output Operation with the desired output operation, Output Alert, the eHome Main Alerts defines the UI for main alerts. If an event requires an alert, Output Alert describes the alert. The design follows Object-Process Methodology (OPM), OPCAD, OPDs (Object-Process diagrams) [6].

Another solution for Ambient Intelligence based home care systems (AHCS) was offered by amiCA (AmI Care and Assistance) prototype for monitoring Monitoring drinking, Monitoring Food Quality, Location Tracking and Fall detection, using different amiCA services interconnected by service platform which was developed in BelAmI project (Bilateral German-Hungarian Collaboration Project on Ambient Intelligence System), based on SOA paradigm. COAALAS (COmpanion for Ambient Assisted Living on Alive-ShareIt platforms) aim is to model the sensor network around disabled users as societies, with the expected behavioral patterns, for supporting smart assistive tools [7]. From a huge HCS we mention AALOA (AAL Open Association) which is supported by several AAL projects: BRAID (FP7), MonAMI (FP6), OASIS (FP7), OsAmI-Commons (ITEA2), PERSONA (FP6), SOPRANO (FP6), universAAL (FP7) and WASP (FP6) [8].

3 NITICS Business Process Analyze (BPA)

NITICS Project within the AAL Program (Network Infrastructure for Innovative home Care Solutions) aim is to develop an advanced ICT solution which includes monitoring and navigational support for indoor and to support elderly people in their daily activities (nutrition, personal hygiene, home care) in the context in which an integrated solution is still missing [9].

Main *Operational objectives* are [1]:

- a) Defining and designing a flexible service platform allowing and facilitating the integration and consolidation of existing elements, leading to a continuous improvement and extension of end-user services. NITICS also provides opportunities to design new customized services to better take care about the end-user;

- b) Improving the quality of life of elderly and disabled persons by allowing them to be mobile in a safe way inside the house and sustain them in their daily life activities;

- c) Improving self-sufficiency of elderly and disabled persons, by self-caring at home (self-check of health conditions and life-style, medication reminder, nutrition status monitoring and alerting), in order to avoid excessive workload and cost from the involved caregivers

- d) Improving the response in terms of efficiency (quality and speed) from the care providers and from the individual's family in emergency situations, by an alarming system, by sensors/cameras feedback to carers, by remotely controlling devices and by video conversations with carers.

Providing care in an efficient way, by making use of reliable information on the condition of the elderly, which will indicate whether an informal carer is needed for aid or if a more specialized formal carer has to intervene.

In our project we have identified four categories of actors: end-users, caregivers, IT specialists, application and service providers. End-users are elderly and people with diseases and disabilities.

Because of agile methodology for system development application and service providers should be able to develop and implement applications and offer maintenance and upgrade services, according to end-users or caregivers demand.

First step in our system development was to analyze the end-user requirements. We have applied two questionnaires for elderly people (over 65 years) in order to identify the main activities in which they need help and also to identify the type of help needed. After first questionnaire we have identified two scenarios to address the needs of the users:

- activities that require physical help - in which case a platform is needed that triggers a request for help either automatically (e.g. vital parameters are out of range, falling alarm, etc) or through the interaction with the user (e.g. button pressing). The request is sent to a caregiver which can be a professional, family, neighbor or friend (AAL@home features).

- needs that can be addressed fully by an ITC platform alone: reminders, day organizers, virtual entertainment and interaction, etc (AAL@home and AAL@Comunity features).

Based on our results, we re-define Siciliano classifications for NITICS project, ordering need items by importance, as obtained from the interviews and we update challenges and electronic support according to end-users requirements. In hierarchical classification we used three clusters: needs, support and challenges and we obtained connections between these clusters linked with basic needs for elders living [1].

Second questionnaire has 61 items (Q1-Q61), and was applied on 59 persons, elder people between 65 and 83 years, with chronically or severe diseases, several of them (12%) have also disabilities. End-users hierarchically needs and electronically support was classified hierachicaly from very important need (first row) to less important need (last row). Another needs classification is focused on caregivers help to solve their needs. Needs that require help from a caregiver prevail and needs that can be addressed without direct help form caregiver.

We focused on first category of needs for the third survey regarding the caregiver perspective on the elderly needs and their perception towards various dimensions of daily life (i.e., health, interaction with technology, health care etc.). The interviews were based on a questionnaire specifically developed for this purpose which comprised 32 items (Q1-Q32) divided in seven sections: demographic data and other primary measures, specificity of caregiving services, the client related impact, evaluation the opinion of the caregivers, according to their own experience, on the importance of the functionalities of an ICT platform towards facilitating the delivery of caregiving services to the elderly, technology acceptance and the users' perception in relation with the project goals (interest in future participation in surveys and system testing).

4 NITICS Business Activity Monitoring (BAM)

NITICS has developed using RDE and HCI scenarios detailed by Cesta and Pecora based on several domotic components and functionalities of domotic components using service-providing software infrastructure [4]. This section offers only HCI scenarios for Emergency Treatment of elders with special needs and implementation was developed based on agile methodology, using domotic components, SOA and intelligent agents. If we consider a private Care Center that can be assimilated as an organization, which has its own Business Process (BP). Care Center activity can be a Business Process, which is defined as a collection of related tasks that produce a specific service or product (to serve a particular goal, in our case elders care) for a particular customer: elderly people [10]. BPMN 2.0 specification defines three models for different aspects of processes: Process Model, Choreography Model and Collaboration Model. To achieve the planned objectives of the project, of the center as an organization we use the Process Model that describes how operations are carried out [11].

Because our target was a care center like an entity, we model the process at a private level of abstraction, as an internal Business Processes. In future work-package we shall develop public levels as collaborative B2B Processes.

We used the software Activiti 5.13 as a platform for BPMN 2.0, released under the Apache open source, written in Java, which can run in any Java application, on a server, in the cluster or in the cloud. It is an alternative implementation of BPMN 2.0 jBPM (JBoss BPM). Activiti

Modeler offers a solution modeling analyst BPMN 2.0 business process that can be compiled in a web browser. This process can be easily distributed because it requires no client software before one starts modeling. Activiti Designer is an Eclipse plug-in (editing application framework) that allows developers to improve its process modeling activity in a process Modeler BPMN 2.0, which can be executed by the processing engine. Activiti Explorer provides an overview of the processes implemented, management, interacting with them and viewing tables in the database managed processing engine [12].

The control of this Care Center is done through a web portal in which access is allowed to both health care professionals, patients, and their relatives. Recording and monitoring equipment in patients' rooms uses XMPP server, which are scripts that monitor and take the equipment alarms and sending the data to the interpretation workflows PubSub method (Publish and Subscribe). Workflows are managed by Active framework, which runs each process. Sending data is ensured by BOSH transport protocol and for identifying each user equipment and network, Jabber employs a unique identifier (JID), consisting of the user name @ name of the server that houses the respective users. Equipment registered is required to be monitored by assigning a JID unique for each.

Portal scripts require monitoring a particular device. When it receives an alarm from one of the sites, it creates a device notification and follows the chain of tasks assigned for each type of alarm process diagrams, from active to complete execution of that process. Scripts are responsible for announcing persons assigned a specific task through the portal by notification by SMS or email. Client control section presents the apartment of a patient with all the equipment found in him. In the living room are the following sensors:

- Main Entrance Access Point: represented by the apartment alarm system, it indicates that the system is armed or not;
- Entrance sensor: a sensor that transmits if the apartment front door is opened or closed;
- Living room siren: it is a siren that can start in case of serious alarms;
- Window Sensor: a sensor that transmits if the window is open or not;
- Living room heating: a sensor attached to a radiator that indicates whether heater is switched into the chamber;
- Living room thermostat: A thermostat that can show several indicators, such as whether the heater is switched on, room temperature or desired temperature.

In the bedroom are some of the sensors that are in the room

- Presence Bedroom: a motion sensor that indicates whether someone is in the room or when he or she has moved last time.
- Bedroom light: indicates whether the bedroom light is on / off, and brightness level.
- Monitoring Camera: camera that monitors the patient's room.
- Bedroom Alarm Button: is a button next to the bedside alarm that can be pressed in an emergency.
- Bedroom Floor sensor: in the floor in front of the bed there is a sensor to track whether this is down from the bed and cannot get up.

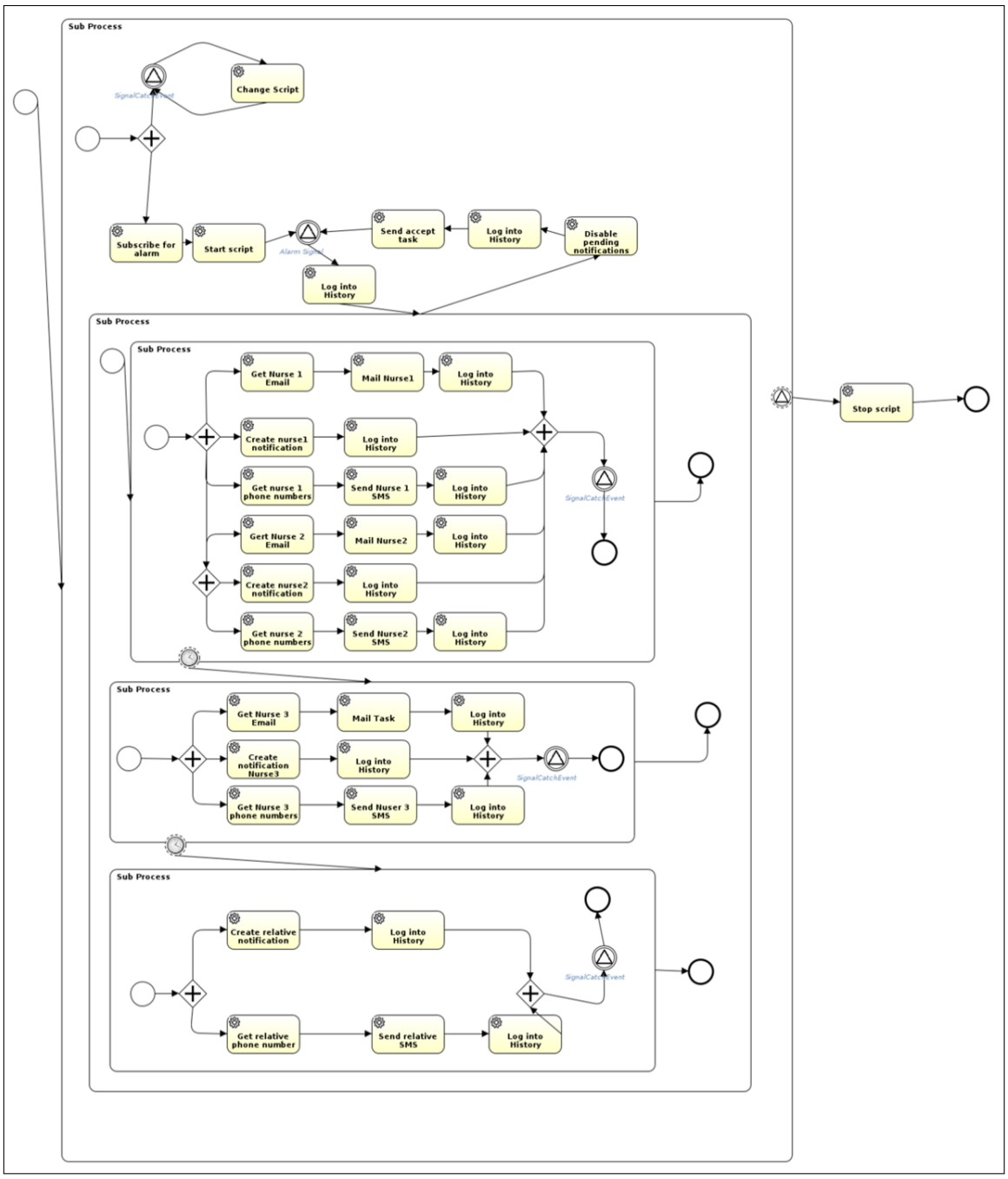


Figure 1: BPMN implementation for Bed Alarm workflow

- Bed sensor (Bed occupancy sensor) indicates whether the bed is occupied or not, specifically whether or not it is the patient bed.

In addition, there are other sensors as required for patient safety: Presence Bedroom, Bedroom light, Monitoring Camera, Bedroom Alarm Button, Bedroom Floor sensor, Bed occupancy sensor. All sensors have the option to get the history of the events that have triggered, and the ability to perform certain tasks directly accessing the portal, without having displacement chamber. These tasks can be reinforcement safety system, turning heat, light switch, changing luminosity, opening and closing the door or window.

We exemplify BPMN implementation for Bed Alarm workflow (Fig. 1). Equipment that governs the workflow is the sensor bed - Bed Occupancy Sensor. This sensor triggers an alarm if the patient exceeds a certain number of hours without getting up from bed.

For example, if the patient does not rise over 10 hours suggesting that the patient may have an ailment. From here the patient is monitored Bed Alarm workflow is started by setting specific parameters. PubSub technique requires scripting to monitor the equipment. From the moment the script start monitoring waiting sensor bed alarms. When an alarm is received, it is recorded in history, and then starts a sub alarm to those who need to take alarm and to execute the task.

Depending on the parameters set to start workflow are determined people who will receive this alert. For the default parameters in parallel performs the following tasks: obtain assistance one email address, it sends an e-mail that is announced by this alarm, it creates a portal notification for this nurse to obtain his phone number and to send an SMS to this notice. At the same time it is announced and nurse 2, following the previous steps (similar to nurse 1). Once these tasks have been completed, it sends a signal that the first two nurses were announced and the first sub-ending. After the time set to start workflow, if neither of the two nurses did not take the alarm then goes ahead and announces the following persons.

Patient's relatives are notified only by SMS and notification portal, and the response time is greater than of the nurses. When one of them takes the alarm notification, then the script is disabled. The historical record sends permission to perform the task, and the script goes back to the state of waiting for a new alarm monitoring. Demonstration parameters that can start this workflow are the following: o Maximum occupancy: 8:00 p.m. o Alarm Priority: high o Type of alarm : SMS, email and notification portal o The receiver alarm 1: nurse 1 and nurse 2 if they do not take the alarm in 5 minutes; o Alarm receiver 2: two nurse (who has already been notified by the portal, but not via email or SMS) unless you take alarm in 5 minutes, o Alarm receiver 3: relative 1 and relative 2 (will be notified through the portal and SMS), if those do not accept within 30 minutes; o Alarm receiver 4: audible alarm emergency room.

5 Conclusions and future works

Our solution offers a BPM approach for an intelligent home care solution. Elderly care in a specialized institution may be considered a business process. The existence of repetitive tasks enables automation of these processes. Each process is modeled and it is presented as a set of individual activities or by compound several sub-processes. The modeling process shows the activities in the order in which they occur, the actors performing the activities, inputs and outputs for each activity, information flows input/output during the process, the rules used in the process. All these are made with the Activiti framework, which may possess all components to develop complex workflows. BPMN 2.0 standard underpinning this framework and it facilitates the work of business analysts and developers, providing the right set of graphics required modeling process. Process automation is a necessity because it allows labor efficiency for nurses and specialized caregivers. Indeed, they are not forced to make regular visits to patients, just only when the visits are needed. In addition the health, elderly people can be easily traced. The proposed model is safe for patients being monitored continuously and it also ensures that their relatives or friends are well cared for. By comparison with other existent solution, we focused on process automation and modeling workflows using IoT elements.

Acknowledgment

The work presented has been founded by the research Grant FP7 AAL Joint Programme, Networked InfrasTructure for Innovative home Care Solutions, and was financed by Center IT

for Science and Technology, Romania.

Bibliography

[1] NITICS (Network Infrastructure for Innovative home Care Solutions) (2013), AAL Joint Program - © NITICS (4 of 33) Call AAL-2012-5, http://pl.ssw.org.pl/portofolio-view/dolore-magna-aliqua

[2] Botterman, M., (2009); Internet of Things: an early reality of the Future Internet. *Workshop Report, European Commission Information Society and Media.*

[3] Nehmer, J., Karshmer, A., Becker, M., Lamm, R.(2006); Living Assistance Systems - An Ambient Intelligence Approach. *Proceedings of the International Conference on Software Engineering (ICSE)*, DOI: 10.1145/1134285.1134293, 43-50.

[4] Cesta, A., Pecora, F., (2005); The ROBOCARE Project: Intelligent Systems for Elder Care, American Association for Artificial Intelligence, www.aaai.org

[5] Bregman D.; Korman, C. (2009); A Universal Implementation Model for the Smart Home, *International Journal of Smart Home*, 3(3): 15-30.

[6] Zoref, L.; Bregman, D.; Dori, D. (2009); Networking Mobile Devices and Computers in an Intelligent Home, *International Journal of Smart Home*, 3(4): 15-22..

[7] Becker, M.; Werkman, E.; Anastasopoulos, M.; Kleinberger, T. (2007); Approaching Ambient Intelligent Home Care Systems, *Pervasive Health Conference and Workshops, 2006*, DOI: 10.1109/PCTHEALTH.2006.361656, 1-10.

[8] Universal open platform and Reference Specification for Ambient Assisted Living, http://www.universaal.org/index.php/en

[9] Serbanati, A.; Medaglia, C.M.; Ceipidor, U.B. (2011); emphBuilding Blocks of the Internet of Things: State of the Art and Beyond, ISBN 978-953-307-380-4, 382 pages, InTech Publisher.

[10] White, S.A.; Miers, D. (2008); *BPMN Modeling and Reference Guide: Understanding and Using BPMN*, Future Strategies Inc., Lighthouse Point, Florida, USA.

[11] OMG, 'Business Process Model and Notation (BPMN): Version 2.0 specification', *Technical Report formal/2011-01-03*, Object Management Group, (January 2011).

[12] http://www.activiti.org/userguide/, last access: 12.07.2013.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):238-247, April, 2015.

Multicriteria Supplier Classification for DSS: Comparative Analysis of Two Methods

J.M. Sepulveda, I.S. Derpich

Juan M. Sepulveda*, Ivan S. Derpich
University of Santiago
3769 Ecuador Ave., Santiago, Chile, CP 7254758
*Corresponding author: juan.sepulveda@usach.cl
ivan.derpich@usach.cl

Abstract: In this paper the analysis of two multicriteria decision making (MCDM) methods for sorting suppliers in industrial environments is presented. The MCDM methods correspond to Electre and FlowSort and both are applied to the classification of providers in an actual case of the local softdrink bottling industry in Chile. The results show that Electre as an outranking method it may well classify suppliers in a similar manner as FlowSort does. Nevertheless, due to the intrinsic underlying fuzzy multicriteria nature of the problem, FlowSort is found to be more suitable method for building a rule-based system based on preference functions for automating the process of suppliers clustering when developing strategies of relationship management in the sense of the Kraljic categories in supply chain management.
Keywords: Decision Support Systems, Supply Management, Electre, FlowSort.

1 Introduction

Supplier management is crucial in order to improve the benefits that a company can have at the operational, functional, economic, and financial levels and in terms of the supply chain it is embedded. Supplier evaluation is a regular process of operational management in many companies; the need arises from finding suppliers for new products, parts, or materials, or for assessing performance of the current supplier base in order to decide continuation of their services. However, in supplier management a more strategic task is classifying suppliers into categories for deciding the relational approach to be followed. Kraljic [1] in his pioneering work established categories of suppliers according to the economic impact for the purchasing company and the risks the suppliers may experiment in their respective market. As recently reviewed by Monczka et. al. [2] Kraljic's categories are defined by a matrix of four quadrants, namely: routine, leverage, bottleneck and critical quadrant. Each category means different strategies ranging from automated transactions by the use of ERP and/or EDI of commodity items of low total purchasing expenses, i.e. routine category, up to the critical category in which for example the expenses are very high and the suppliers situation in its market is risky due to uncertainties, uniqueness or high dependence. See Figure 1 for an example of the Kraljic matrix. The number in parenthesis is a level in a scale from 1 to 4, as explained below.

Level 1: Many alternative products and processes; abundant sources of supply; low value-small individual transactions; routinary use, unspecified items; no specialized purchasing knowledge required.

Level 2: Complex specifications requiring complex manufacturing or service processes; few alternate production/sources of supply; big impact on operations / maintenance; new technology or untested processes.

Level 3: High expenditures, commodity items; large marketplace capacity, big inventories; many alternate products and services; many qualified providers; market/price sensitive demand.

Level 4: Critical to profitability and operations; few qualified sources of supply; large expenditures; design and quality critical; complex and/or rigid specifications.

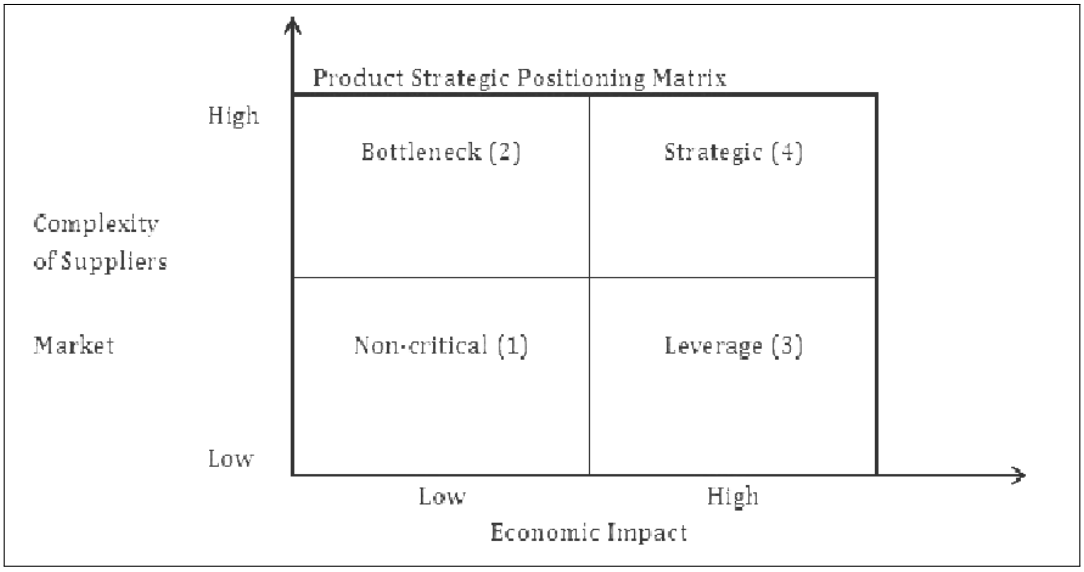


Figure 1: Kraljic Matrix 1 for strategic positioning

In this work the classification or categorization of suppliers in the case study, defines three categories: (a) transactional, (b) collaborative, and (c) integrated. These categories are specific for the company dealt with; however the method is general enough as to be applied to other similar situations.

A transactional supplier is understood as the profile of routine suppliers combining low purchasing expenses, easy-to-obtain commodity-type products or services, and with low mutual attraction in the relationship. A collaborative supplier is one of importance to the company due to high purchasing expenses. For this reason, close monitoring of performance is needed in order to assure the service level agreements and to control the costs. The logistics complexity is of medium range as well as the mutual attraction. An integrated supplier is a critical one, of such an importance that a strategic alliance is needed, such as the case of a third party fully manufacturing a component or a product for a given customer company's brand (OEM strategy), as it occurs in the food, pharmaceutical, or car industries. It combines criticality in product positioning with high logistics complexity and high mutual attraction of the relationship.

Clearly, the three categories above can be understood as fuzzy sets where the implementation process of the method allows the identification of strengths and weaknesses of using formalized supplier selection models to tackle the supplier sorting problem. This highlights potential barriers preventing the adoption of these types of methods. For this purpose, the paper presents a number of relevant issues arising from the application and managerial implications for both customer and suppliers, concerning differentiated management practices according to their classification.

Besides Matrix 1, two other matrices are used to classify suppliers. Matrix 2 values the complexity of logistics, and Matrix 3 the mutual attraction of the SRM (supplier relationship management).

In Matrix 2 (Lead Time and Stock Rotation) the lowest level of logistics complexity is for supplies of fast replenishment and slow moving items. It the follows the case of suppliers of items with short lead time (hours, days, within a week) but of high consumption or inventory turnover. The third level is for slow moving items and long lead times (e.g. imported products); finally, the most complex logistics is for suppliers of items with high rotation and long lead time due the risk of inventory breakdown.

The numbers assigned as well as the criteria are depending on the specific application and are

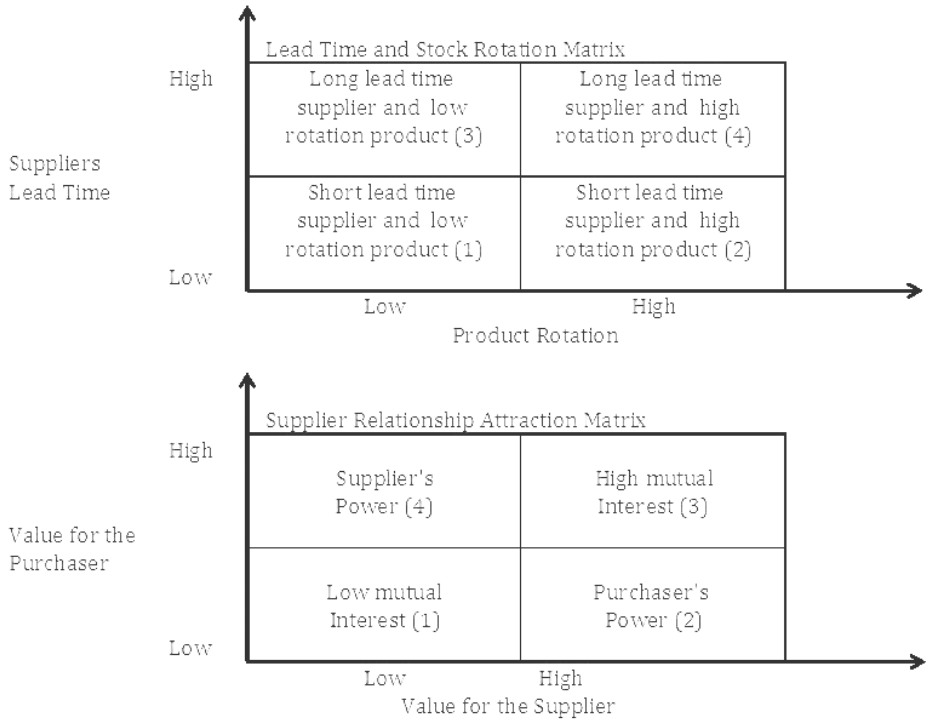


Figure 2: Matrices 2 and 3 for logistics complexity and SRM attraction level

provided by experienced purchasing executives, without loss of generality. In Matrix 3 (Supplier Relationship Attraction) the lowest level is assigned when both parties have little mutual interest. It follows when the purchaser is important to the supplier (level 2). For the specific case of the application, the managers gave the highest importance to suppliers with superior power in the relationship (generally due to size).

2 Strategies by category

With the suppliers being categorized, the organization can apply differentiated approaches to SRM. Next, a brief description of each category is given.

2.1 Transactional category

The transactional category has as main characteristics the existence of many alternative products and services, many sources of supply, low value small individual transactions, commodity-type items, requiring little or no specialization of the purchasing executive. Hence, this category is non-critical and besides the logistics aspects are simple because lead times are short and rotation is low. The mutual relationship is of low level. The general strategy for this category is the simplification of the acquisition process. The tactics relate to increase the role of computer based systems and reducing the buying effort. Specific actions include rationalization of the supplier base, the automation of the purchasing process by the use of automated requisitioning, electronic data interchange, stockless procurement, minimization of administrative costs. Little negotiation is needed.

2.2 Collaborative category

The collaborative category has as main characteristics the existence of many alternative products and services, many sources of supply, commodity-type items, requiring little or no specialization of the purchasing executive. However, the purchasing expenses and volumes are higher. Hence, the optimization of operations serve as leverage for obtaining commercial advantage. Logistics may be more complex due to longer lead times and higher rotation of items. The type of relationship is of purchaser's power. The general strategy for this category is maximizing the commercial advantage. The tactics involve concentration of purchasing to increase power and also to maintain competition. Specific actions involve the promotion of competitive bidding, procurement coordination, adoption of industry standards to reduce dependency.

2.3 Integrated category

This category has as main characteristics a heavy control of the providers performance, partnerships in the form of a strategic outsourcing, cooperation to optimize mutual benefits. For instance, a strategic supplier performing outsourced implant critical tasks of production and operations. The complexity of operations demands close control and information exchange. The importance of the purchaser and the criticality of the supplier calls for relations of mutual benefits. The attraction force of both parties is high.

3 Basics of the two methods

The ELECTRE methodology is based on the concordance and discordance indices. The simplest method of the ELECTRE family is ELECTRE-I. See Figueira et al. [3].

This method starts from the data in the decision matrix, and assumes that the sum of the weights of all criteria is equal to one. For an ordered pair of alternatives (A_j, A_k), the concordance index c_{jk} is the sum of all the weights for those criteria where the performance score of A_j is least as high as that of A_k, as in formula (1). It can be seen that c_{jk} lies between 0 and 1.

$$c_{jk} = \sum_{i:a_{ij} \geq a_{ik}} w_i \qquad j,k = 1,..,n, j \neq k \tag{1}$$

The computation of the discordance index d_{jk} is more elaborated: $d_{jk} = 0$ if $a_{ij} > a_{ik}$, $i = 1, ..., m$, that is, the discordance index is zero if A_j performs better than A_k on all of the criteria. Otherwise, d_{jk} is calculated as in (2). That is, for each criterion where A_k outperforms A_j, the ratio is calculated as the difference in performance level between A_k and A_j and the maximum difference in score on the criterion concerned between any pair of alternatives.

$$d_{jk} = max_{i=1,..,m} \frac{a_{ik} - a_{ij}}{max_{j=1,..,n} a_{ij} - min_{j=1,...,n} a_{ij}} \qquad j,k = 1,..,n, j \neq k \tag{2}$$

The maximum of these ratios (which must lie between 0 and 1) is the discordance index. A concordance threshold c^* and discordance threshold d^* are then defined such that $0 < d^* < c^* < 1$. Then, A_j outranks A_k if the $c_{jk} > c^*$ and $d_{jk} < d^*$, i.e. the concordance index is above and the discordance index is below its threshold, respectively.

FlowSort is based on the Promethee ranking methodology (Figueira et al., 2005) [3], this new sorting method by Nemery and Lamboray [4] is utilized for assigning actions to completely ordered categories; these categories are defined either by limiting profiles (i.e., min and max values) or by central profiles (or centroids). This method has also been applied by (Sepulveda

et al., 2010) [5] in another management decision problem in the innovation field for diagnosing capabilities in small enterprises.

In what follows, limiting profiles will be used. The assignment of an action (i.e., an object to be sorted) into a category is based on the relative position of this action with respect to the defined reference profiles in terms or incoming or outgoing net flows. Let $A = (a_1, a_2, ..., a_n)$ be the set of n actions or alternatives to be sorted. These actions are evaluated on q criteria $g_j (j = 1, ..., q)$; all criteria are supposed to be maximized in the decision making problem. The categories to which the actions must be assigned are denoted by $C_1, C_2, ..., C_k$. Let $R = (r_i, ..., r_{K+1})$ be the set of limiting profiles in the case when a category is defined by an upper and lower limit. Let $\pi(x, y)$ be the preference of an action x over an action y, as in the Promethee method.

Figure 3 shows typical shapes of preference functions where the x-axis is the degree of difference between actions x and y. Thus, the positive, negative and net flows ϕ of each action x in R, are computed by equations (4) (5) (6) where $\dot{R}_i = R \text{ U } \{a_i\}$ is the extended set of profiles either for the limiting profile case or the central profiles. The rules for assigning actions a_i to a category C_h are given by equations (7) and (8) in the case of limiting profiles.

$$\pi(x, y) = \sum_{j=1}^{q} w_j P(x, y) \tag{3}$$

$$\phi^+_{\dot{R}_i} = \frac{1}{\left|\dot{R}_i\right| - 1} \sum_{y \in \dot{R}_i{}^+} \pi(x, y) \tag{4}$$

$$\phi^-_{\dot{R}_i} = \frac{1}{\left|\dot{R}_i\right| - 1} \sum_{y \in \dot{R}_i{}^-} \pi(x, y) \tag{5}$$

$$\phi_{\dot{R}_i} = \phi^+_{\dot{R}_i} - \phi^-_{\dot{R}_i} \tag{6}$$

$$C_{\phi^+}(a_i) = C_h, \quad if \quad \phi^+_{\dot{R}_i}(r_h) \geq \phi^+_{\dot{R}_i}(a_1) > \phi^+_{\dot{R}_i} \tag{7}$$

$$C_{\phi^-}(a_i) = C_h, \quad if \quad \phi^-_{\dot{R}_i}(r_h) < \phi^-_{\dot{R}_i}(a_1) \leq \phi^-_{\dot{R}_i}(r_{h+1}) \tag{8}$$

4 Results of the two methods

First, the sorting will be made by the Electre method. Table 1 shows the score combination for six randomly chosen suppliers and the resulting category according to the values predefined for the combination by using Electre. The six cases are for illustrative purposes and show the type of results without loss of generality.

Table 2 shows the limiting profiles for the FlowSort method.

The scores for each criteria in FlowSort are the same as the combination values in Table 1.

Table 3 shows the results for FlowSort. In the table, the six extended sets R correspond to suppliers A,B,C,D,E,F, respectively. The category is obtained by applying the rules defined by equations (7) and (8). It can be observed that for the six suppliers, with the exception of C, the assigned categories are the same.

This is encouraging since the basis of each method is very different. It can be said that in case C this provider was better classified by FlowSort. The difference obtained in one of the suppliers (Supplier C), is mainly because of the assessment data that show that this provider

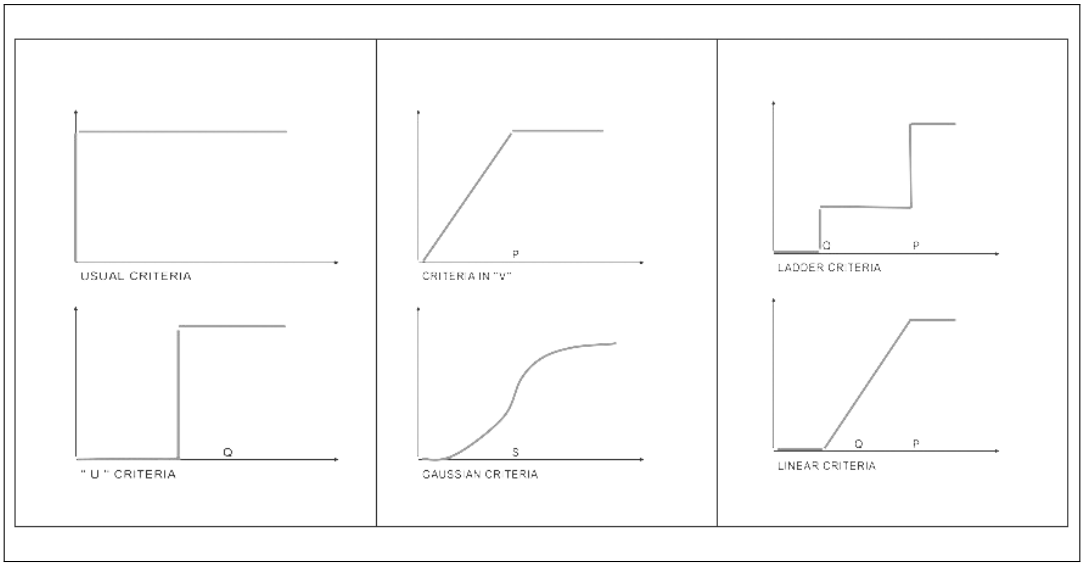


Figure 3: Types of Preference functions

Table 1: Results by Electre

Supplier	**Score combination**	**Category**
A	"4-3-3"	Integrated
B	"3-2-2"	Collaborative
C	"1-2-3"	Transactional
D	"4-3-4"	Integrated
E	"4-3-4"	Integrated
F	"4-3-2"	Integrated

Table 2: Limiting profiles chosen for FlowSort

Profile	C_1	C_2	C_3
r_1	4.5	4.5	4.5
r_2	3.0	3.0	3.0
r_3	1.5	3.0	3.0
r_4	0	0	0

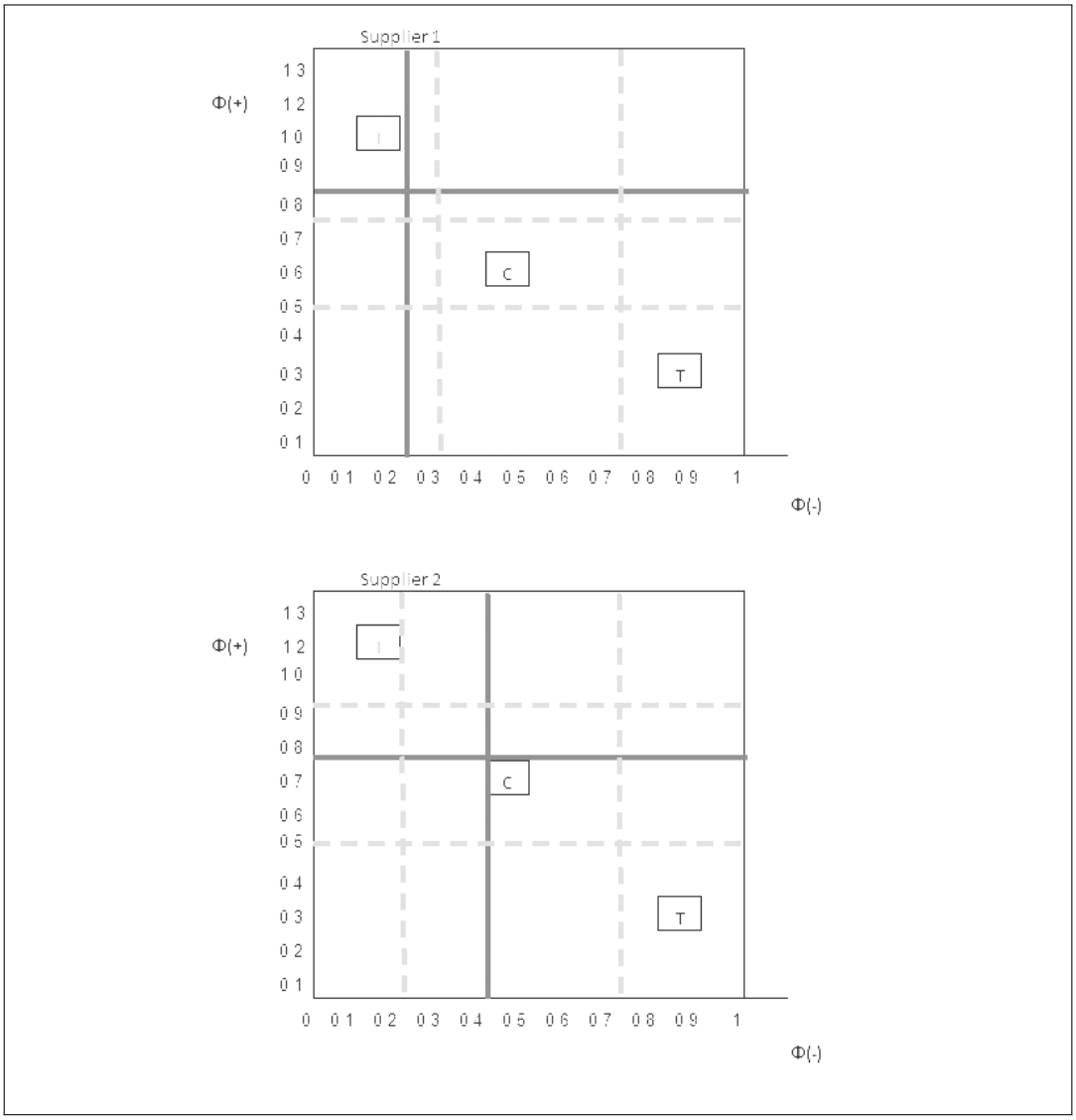


Figure 4: Graphical Representation of FlowSort Results for Two Suppliers

Table 3: Results for FlowSort

Flow Sort	**Results**	r_1	r_2	r_3	r_4	$A(i)$	**Category**
$\dot{R}1$	ϕ_+	1.25	0.75	0.5	0	0.83	Integrated
	ϕ_-	0	0.33	0.75	1	0.25	Integrated
	ϕ_{net}	1.25	0.42	-0.25	-1	0.58	Integrated
$\dot{R}2$	ϕ_+	1.25	0.92	0.5	0	0.75	Collaborative
	ϕ_-	0	0.25	0.75	1	0.42	Collaborative
	ϕ_{net}	1.25	0.67	-0.25	-1	0.33	Collaborative
$\dot{R}3$	ϕ^+	1.25	0.92	0.58	0	0.67	Collaborative
	ϕ^-	0	0.25	0.67	1	0.5	Collaborative
	ϕ^{net}	1.25	0.67	-0.08	-1	0.17	Collaborative
$\dot{R}4$	ϕ^+	1.25	0.75	0.5	0	0.92	Integrated
	ϕ^-	0	0.42	0.75	1	0.25	Integrated
	ϕ^{net}	1.25	0.33	-0.25	-1	0.67	Integrated
$\dot{R}5$	ϕ^+	1.25	0.75	0.5	0	0.92	Integrated
	ϕ^-	0	0.42	0.75	1	0.25	Integrated
	ϕ^{net}	1.25	0.33	-0.25	-1	0.67	Integrated
$\dot{R}6$	ϕ^+	1.25	0.83	0.5	0	0.83	Integrated
	ϕ^-	0	0.33	0.75	1	0.33	Integrated
	ϕ^{net}	1.25	0.5	-0.25	-1	0.5	Integrated

has some lower scores in some criteria (matrix- criterion 1) and high in other ones (2 and 3) and the final class depends on the decision maker.

However, note that FlowSort is a method that reflects the in and out flows generated by the alternatives, while the classification made in the Electre is arbitrary at some extent by the threshold values chosen for each category. Nevertheless, both methods solve the problem and are suitable for automating the classification process as part of a decision support system for supply management.

Figure 4 shows a graphical representation of the sorting for Suppliers 1 and 2, as given by Table 3.

In Table 1 the category is chosen according to the values in Table 4. This shows the results of the Electre method for the 64 possible combinations of the three matrices with the aggregated dominance index (concordance minus discordance). The score values of Table 4 in columns Score Matrix 1,2,3 (indicated as SM1, SM2, SM3) for each supplier score combination (SC) were given by experienced supply managers same as in the FlowSort method. Current work is addressing the obtention of these numbers from the key performance indicators in the database of a business intelligence module connected to the ERP system.

The categories are obtained by sorting in increasing order of the aggregated dominance index (A.D.I.) and by defining thresholds for each category. The weights were chosen as 1/3 each matrix (w.l.g). For instance, as shown in Table 1, Supplier A with scores "4-3-3" corresponds to the combination "a60"in the Appendix, Supplier B with scores "3-2-2" corresponds to "a35", and so on. In this case, the same cases were used for both the Electre and FlowSort methods.

Table 4: Categories by Electre

SC	SM1	SM2	SM3	A.D.I.	Category	SC	SM1	SM2	SM3	A.D.I.	Category
a1	1	1	1	0	Transactional	a34	3	1	4	17	Collab.
a2	1	1	2	1	Transactional	a37	4	1	3	17	Collab.
a3	1	2	1	1	Transactional	a38	4	3	1	17	Collab.
a4	2	1	1	1	Transactional	a30	1	3	4	18	Collab.
a5	1	1	3	2	Transactional	a31	1	4	3	18	Collab.
a6	1	3	1	2	Transactional	a36	3	4	1	18	Collab.
a7	3	1	1	2	Transactional	a40	2	2	4	20	Integrated
a8	1	1	4	3	Transactional	a43	4	2	2	20	Integrated
a9	1	2	2	3	Transactional	a41	2	4	2	21	Integrated
a10	1	4	1	3	Transactional	a46	3	2	3	21	Integrated
a11	2	1	2	3	Transactional	a47	3	3	2	21	Integrated
a12	2	2	1	3	Transactional	a45	2	3	3	22	Integrated
a13	4	1	1	3	Transactional	a44	4	4	1	27	Integrated
a14	1	2	3	6	Transactional	a39	1	4	4	28	Integrated
a15	1	3	2	6	Transactional	a42	4	1	4	28	Integrated
a16	2	1	3	6	Transactional	a50	3	2	4	31	Integrated
a17	2	3	1	6	Transactional	a53	4	3	2	31	Integrated
a18	3	1	2	6	Transactional	a48	2	3	4	32	Integrated
a19	3	2	1	6	Transactional	a49	2	4	3	32	Integrated
a23	2	2	2	7	Collaborative	a51	3	4	2	32	Integrated
a20	1	2	4	10	Collaborative	a52	4	2	3	32	Integrated
a21	1	4	2	10	Collaborative	a54	3	3	3	35	Integrated
a22	2	1	4	10	Collaborative	a56	4	2	4	43	Integrated
a24	2	4	1	10	Collaborative	a57	4	4	2	43	Integrated
a25	4	1	2	10	Collaborative	a55	2	4	4	44	Integrated
a26	4	2	1	10	Collaborative	a58	3	3	4	45	Integrated
a27	1	3	3	11	Collaborative	a59	3	4	3	45	Integrated
a28	3	1	3	11	Collaborative	a60	4	3	3	45	Integrated
a29	3	3	1	11	Collaborative	a61	3	4	4	55	Integrated
a32	2	2	3	12	Collaborative	a62	4	3	4	55	Integrated
a33	2	3	2	12	Collaborative	a63	4	4	3	55	Integrated
a35	3	2	2	13	Collaborative	a64	4	4	4	63	Integrated

5 Conclusions and future works

In this article, the comparison of two methods for supplier sorting for determining the supplier management strategy in large organizations has been presented. Assigning a category to a supplier is an important task within supply chain management since many types of suppliers are commonly in place and differentiated management approaches are needed in order to accomplish the efficiency and service objectives. The sorted categories combined three main dimensions: strategic positioning (matrix 1), logistics complexity (matrix 2), and attraction of the mutual relation (matrix 3). While the original Kraljic's matrix is concerned only with strategic positioning, giving four categories of suppliers: non-critical or routine, leverage, bottleneck, and critical, the contribution in this work is that the analysis has been extended to other dimensions, such as logistics complexity (matrix 2) and the attraction of mutual relation (matrix 3).

By using the extended Kraljic's matrix concept developed in this work, a portfolio of strategies may be identified according to the defined categories: transactional, collaborative, and integrated suppliers. Commonly, the analysis of suppliers is performed based on experienced managers over a restricted number of cases. In large organizations, however, because of the high number of suppliers, such manual method becomes difficult and prone to error. With multicriteria sorting

models it is possible to overcome this weakness and even automate this task. Flowsort was compared to ELECTRE. A good coincidence was obtained between these two methods. However Electre requires previously defined categories by using explicit enumeration and threshold values entered manually by experts. In this sense, Flowsort requires less human input being more adaptable for automated processing.

As direction for future works, further research is needed in order to examine the robustness of the results and the effects of the scales used in the assessment. Also, in order to minimize human data input, the scores in the matrices ideally should be obtained in a direct manner from the key performance indicators (KPI) obtained from the ERP system of the organization, or from a business intelligence (BI) module, among other aspects. Ongoing work of the authors is addressing these issues.

Acknowledgments

The authors are very grateful to DICYT (Scientific and Tecnological Research Office), Project Number 061117SS and the Industrial Engineering (IE) Department, both of the University of Santiago of Chile for their support in this work. Also to IE graduates Marcos Melin and Stephanie Sepulveda who helped in the data collection and model implementation.

Bibliography

[1] Kraljic, P. (1983); Purchasing must become supply management, *Harvard Business Review*, September-October, 1983.

[2] Monczka, R.M., Handfield, R.B., Giunipero, K.C., Patterson, J.L. (2011); *Purchasing And Supply Chain Management*, 5th Edition, Cengage Learning.

[3] Figueira, J., Greco S., Ehrgott M. (2005); *Multiple Criteria Decision Analysis: State of the Art Surveys*, Springer-Verlag.

[4] Nemery P.; Lamboray C. (2008); FlowSort: a flow-based sorting method with limiting or central profiles, *TOP*, 16, 90-113, Springer-Verlag.

[5] Sepulveda, J., Gonzalez, J., Alfaro, M. (2010); A Metrics-based Diagnosis Tool for Enhancing Innovation Capabilities in SMEs, *International Journal of Computers Communications & Control*, 5(5):919-928.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):248-262, April, 2015.

A Comprehensive Trust Model Based on Multi-factors for WSNs

N. Wang, Y. Chen

Na Wang
1. MoE Research Engineering Center for Software/
Hardware Co-Design Technology and Application
East China Normal University
No.3663 North Zhongshan Rd, Shanghai
2. Faculty of Engineering
Shanghai Second Polytechnic University
No.2360 Jinhai Rd, Shanghai
wnoffice@126.com

Yixiang Chen*
MoE Research Engineering Center for Software/
Hardware Co-Design Technology and Application
East China Normal University
No.3663 North Zhongshan Rd, Shanghai 200062 China
*Corresponding author: yxchen@sei.ecnu.edu.cn

Abstract: The goal of this paper is to introduce a novel trust model for wireless sensor networks. This trust model calculates trust value of nodes through two kinds of trusts: private trust and interactive trust of a node. Private trust focuses on the past record of a node's sensing and its remaining energy. Interactive trust cares for the interaction of a node with its neighbors. This trust model can recognize faulty nodes inside a network, reduce their impaction to data acquisition, and select a trust routing for precise data transmission. A simulation is given and shows that this trust model has a higher performance than TMS and ECCR in some aspects. But it consumes more energy than ECCR for its comprehensive structure of data.
Keywords: Wireless Sensor Networks, Trust Model, Private trust, interactive trust, Trust routing.

1 Introduction and related work

A wireless sensor network (WSN, shortly) consists normally of thousands of tiny embedded computers which are equipped with a specific type of sensor to sense information from the surrounding environment. The collected information is relayed from sensor to sensor, using a secure multi-hop routing protocol, until the data reaches the desired destination node, which is called as a sink. The WSN technology has been applied in many areas, such as industry, environment, seismology, construction, transportation, military warfare, traffic control and agriculture [1].

Sensor nodes in WSNs suffer often from resource constraints such as low computational capability, limited storage capacity, limited communication bandwidth, and the use of insecure communication channel. WSNs are also prone to varied types of attacks [4–8] such as black hole attack and sniffing attack. Cryptographic solutions can successfully defend against outsider attack but may fail under insider malicious attacks. This vulnerability along with the cooperative nature of sensor networks requires one for assessing the trust relationships among the nodes in the network [10].

Recently, some researches focusing on trust in WSN have been practiced based on different background such as GTMS [12], RFSN [13], HATWA [14] and [15]. But all the above schemas are based on either routing or data sensing respectively. In [13], a method considering key factors in running of WSN is proposed. But the researches are far from adequate. In [11], NBBTE (Node

Behavioral Strategies Banding Belief Theory of the Trust Evaluation Algorithm) is proposed, which integrates the approach of nodes behavioral strategies and modified evidence theory. The same authors as in [11] proposed in [3] a TMS schema based on the method in [11]. Direct trust value on each neighbor node is calculated by considering trust factors which are defined according to node behaviors in order to detect malicious attacks. At the same time, recommended trust value from common neighbor nodes is obtained through conditional transitivity and the weight of each recommendation is obtained by revised D-S evidence theory. Both [11] and [3] consider factors of received packets rate, successfully sending packets rate, packets forwarding rate, data consistency, time frequency, node availability and security grade. Data consistency is defined that the packets sent among neighbor nodes are similar in the same area according to the application. Time factor is defined that the size of time grade is dependent on the specific situation. If it is established too large, then integrated trust value is affected by history heavily. On the contrary, if it is established too small, then trust value relies on a single period overly. Except for the above three factors, others factors mainly aims on communication. The multi-factors in [11] make a progress in trust management for WSN. With regardless of the core algorithm in [11], the factors considered are reasonable and adequate to compute trust value of a common node. But it does not deal with the relationship of the factors. The reference [2] introduces a notion of trust evaluation to build trust mechanism for each node inside network.

About routing, in order to improve reliability, some multipath routing technologies have been mentioned. The k heavier path is used between the source node and purpose and the packet is divided into different packet to transmit in [19]. In literature [20], they take to adjacent cluster head number as a topological construction weights, looking forward to a constant approximate rate based minimum network connected dominating sets, but this method does not consider cluster head of network node energy influence on the performance of the whole. According to the size of the nodes energy to network between the influence of choice, the literature [21] considers energy as weights, ensure constant approximation rate at the same time as a priority high-energy node cluster of communication between nodes, and, to some extent, improve the network energy efficiency, but it ignores the routing communication costs between clusters, existence of high communication costs of premature failure of the head node limitations. In [22], defining link reliability strategy constructed by remain energy, and communication cost of nodes as topology weight to synthetically reflect the energy efficiency of dominator, an Energy-radio and communication cost route (ECCR) is proposed to solve the problem that the average energy consumption in cluster and minimum communication cost. The author takes both node residual energy and distance into account to compete cluster head, at the same time, in order to reduce the cluster head energy cost, link reliability and hop are used to establish topological structure. The experimental results show that the algorithm not only has the energy saved characters, but also ensures the reliability of topology links and extends the network life-cycle efficiently. But ECCR focus on energy efficiency but not data precision while trust may decide which route is trust for transmission to get more precise data, so the performance can be improved in view of trust value.

In order to meet both data and energy requirement, it is necessary to build a trust model based on multi-factors which consider data, communication, clock and energy to help more application such as data aggregation, fault detection and route selection. The contribution of this paper is:

1. Create a node trust model based on interactive factors and private factors.
2. Give a routing algorithm to compute routing trust based on nodes' trust value.
3. Apply our model in fault detection.

4. Evaluate and compare our model with other models.

This paper introduces a novel trust model for wireless sensor networks. This trust model calculates trust value of nodes through two kinds of trusts: private trust and interactive trust. Private trust focuses on the past record of a node's sensing and its remaining energy. Interaction trust cares for the interaction of a node with its neighbors. This trust model can recognize faulty nodes inside a network, reduce their impaction to data acquisition, and select a trust routing for precise data transmission. A simulation is given and shows that this trust model has a higher performance than TMS and ECCR in some aspects. But it consumes more energy than ECCR for its comprehensive structure of data.

The rest of the paper is organized as follows. The definitions and models are proposed in section 2 and 3. The routing algorithm based on our trust model is depicted in Section 4 and we apply this model in fault detection on section 5. The comparison and evaluation of our trust model with other models are given in Sections 6. The conclusions and future work are presented in Sections 7.

2 Definition of key attributes for trust in wireless sensor networks

In order to defeat various attacks, we have to take all kinds of factors that depend on the interactions between neighbor nodes into account. However, there is an obvious trade-off between the number of factors and the energy consumption. In this paper, we pay attention on four key attributes which are called as to be connectivity, consistency, synchronization and adequacy to the node's trust. Since the structure of WSN is divided into inner indicating the level within one cluster and outer intra indicating the level between clusters heads. The structure is shown in Figure 1.

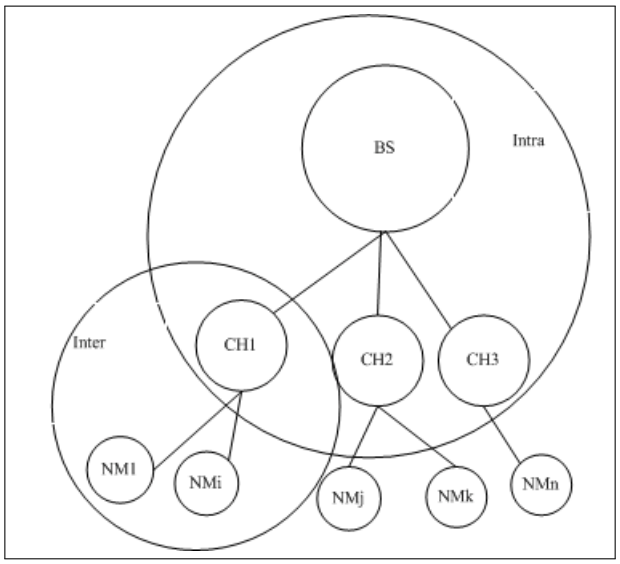


Figure 1: The structure of WSN

We focus on the evaluation of nodes in one cluster and then use the same method to evaluate the trust between cluster heads. Before doing it, we depict the running process of stages within a WSN as follows.

Event-driving stage: When there is a request of detecting in a certain field from sink node, sink node will send a sensing order to its neighbor nodes.

Self-organizing stage: The nodes received the request will be as the first level cluster heads. And the cluster heads will select their members according to cluster protocol such as LEACH.

Detecting stage: Member node senses data after receiving the request message from its cluster head. The sensing action is frequent according to the sampling period.

Communication stage: member node sends its data to the cluster head and its neighbors. In this stage, interaction and data aggregation are crucial to trust value. When a member node sends data to other nodes, there will be two cases that are successful connection and failed connection with regard to communication. And the data sent from a certain member node may have two results comparing with other data that is consistent and inconsistent with regard to data. Furthermore, the sensing moment should take into account when measure data consistency since data has time attribute.

Data aggregation: After data exchange, each node has the information of its neighbors and cluster head has the information of all members in cluster. In order to reduce information and energy consumption, data from member nodes will be aggregated in head with certain format and will be sent to high-level head.

Convergence stage: Data from each level head will be aggregated and sent hierarchically till to the sink.

During the process, attacks such as bad mouth and fault such as hardware fault may influence the result of aggregation. Both attack and fault are called abnormal cases in this paper. In order to exclude the abnormal nodes, one of the resolve methods is giving a trust value to each node. When the trust value is keeping lower for certain duration, it should be deleted from the network. Since the communication and sensing are the main action in WSN, the interaction and data should be regarded as the main parameters to construct trust value.

3 Node trust model based on multi-factors

In this section, we divide node trust into two parts: interactive trust and private trust. Interactive trust describes the trust of a node's interaction with its neighbor nodes based on interactive factors. Private trust focuses on describing a node's nature reputation based on private factors.

3.1 Introduction of factors

In WSN, there are interactions between nodes, so we abstract interactive factors. During the interactions, data including its time attribute will be exchanged in cluster head. We call them as interaction, data and time. Considering the factors, we should give a combined trust model. But as the relations between factors are not simple, traditional model cannot be applied here. Generally, relations between two factors are divided into three classes that are promoted, opposite and uncorrelated. Promoted relationship means the promoting of A will make B promote. Opposite relationship means the promoting of A will make B decline. And uncorrelated relationship means the change of A will make no influence on B. The factors within this paper have relationship as depicted in Table 1. Table 1 describes the relationship of interactive factors which involve success interaction between nodes, data similarity between nodes and approximate sensing time of two nodes. Here, symbol plus means promoted relation between two factors, minus means opposite relation and blank means uncorrelated relation. It is shown in table 1 that when valid interaction increased, similar data will increase (Here, we consume general case that normal data is more than fault data). Time which is an attribute of data has the same relation with other factors as data.

Table 1: Relationships between interactive factors

	Interaction	Data	Time
Interaction		+	+
Data	+		+
Time	+	+	

Table 2: private factors

Factors:	Data	Energy

We also consider the private factors of a node in this paper. Table 2 describes the private factors for a single node in which the variation of sensing data in a period and remaining energy are taken into account. The data factor relates the data correctness and energy factor relates to the node's working ability. We give a reward coefficient to correct data and a penalty coefficient to error data. Energy is a natural factor that can deduce from initial energy and remaining energy. It decides whether the node is running.

3.2 Interactive factors

We define the interactive factors as:

Interaction: When considering the number of continuous communication during t, we compute it as

$$\mathrm{DCT}_{i,j}(\Delta t) = \left\lfloor \left(\frac{100 \cdot s_{i,j}(\Delta t)}{s_{i,j}(\Delta t) + f_{i,j}(\Delta t)} \right) \left(\frac{1}{\sqrt{f_{i,j}(\Delta t)}} \right) \right\rfloor. \tag{1}$$

In order to compress the size of data for energy consumption reduction, the trust value is multiplied by 100 and get integer. Where, $s_{i,k}(\Delta t)$ is the success number of communication between node i and j in time Δt, and $f_{i,j}(\Delta t)$ is the failure number of communication between node i and j in time Δt. When failure number is larger than success number, we may think that these two nodes i and j are distrust. We make the value decline sharply by dividing $\sqrt{f_{i,j}(\Delta t)}$. Specially, if $f_{i,j}(\Delta t) = 0$, we set $\mathrm{DCT}_{i,j}(\Delta t) = 100$ [16].

Data:

$$\mathrm{DST}_{i,j}(\Delta t) = \left\lfloor \left(\frac{100 \cdot c_{i,j}(\Delta t)}{c_{i,j}(\Delta t) + d_{i,j}(\Delta t)} \right) \left(\frac{1}{\sqrt{d_{i,j}(\Delta t)}} \right) \right\rfloor. \tag{2}$$

$$c_{i,j} = \frac{X_i X_j}{X_i^2 + X_j^2 - X_i X_j}. \tag{3}$$

$c_{i,j}(\Delta t)$ is the total number of similar data comparison of node i with j in Δt time, and $d_{i,j}(\Delta t)$ is the total number of dissimilar data comparison. X_i is sensing data of node i. Specially, if $d_{i,j}(\Delta t) = 0$, we set $\mathrm{DST}_{i,j}(\Delta t) = 100$.

Time:

$$T_{i,j} = \left\lfloor \frac{100 \cdot T_i T_j}{T_i^2 + T_j^2 - T_i T_j} \right\rfloor. \tag{4}$$

The factors above can fulfill the complexity of trust evaluation. Furthermore, the factors will change with the elapse of time not only by themselves but by other factors.

3.3 Interactive trust related to interactive factors

Now, we present the interactive trust based on interactive factors. Interactive factors mentioned above are abstracted from WSN which are crucial when computing the trust value between nodes. In order to depict the relations and factors' importance in the model, we use a weighted trust model to compute trust value between two nodes as formula (5).

$$T_I = \Big\lfloor \prod_{i=1}^{n} y_i^{\alpha_i} \Big\rfloor \qquad \Big(\sum_{i=1}^{n} \alpha_i = 1 \Big). \tag{5}$$

T_I presents the direct interactive trust between two nodes, and y_i is value of the ith factors with its weight α_i. Here, y_i indicates four factors which come from formula (1)-(4). Since α_1 is the weight of factors, its value should display the importance of a factor in T_I. With considering the relations described in table 1, we use reciprocal matrix to decide each factor's weight.

For example, from the perspective of optimization, the approach of measuring sort vector according to reciprocal matrix is based on the fact that when $A = \big(a_{i,j}\big)_{n\times n}$ is a reciprocal matrix, $A\omega = n\omega$ and $\omega = \big(\omega_1, \omega_2, \ldots, \omega_n\big)^T$ where $\big(a_{i,j}\big) = {}^{\omega_i}/_{\omega_j}$.

In our paper, we set the reciprocal matrix based on table 1 as Figure2. And use method naming right characteristic root to calculate weight vector as:

(Interaction, Data, Time)=(0.5869, 0.3238, 0.0893).

$$\begin{bmatrix} 1 & 2 & 6 \\ 1/2 & 1 & 4 \\ 1/6 & 1/4 & 1 \end{bmatrix}$$

Figure 2: Reciprocal matrix

3.4 Private trust based on private factors

In WSN, a node's private trust will depend on its previous action. This private trust must be penalized when its sensing data is deviated far from the average and be awarded when its sensing data is correctly consecutively. For example, a fire can start near a sensor, so that sensor will read values higher than its neighbors at round one. If this is the case, and a large penalty is given to the sensor then it will considered as a fault node where in fact it is not. In our work, the node will be penalized with a small factor and will be rewarded in the next round, since the average will tend to be that of a disaster state. Private trust has a combination method that is shown in formula (6).

$$T_p = \begin{cases} \theta_1 \cdot T_{p-1} + \theta_2 \cdot F + \theta_3 \cdot (\mathrm{e}^{-R} - 1) + \theta_4 \cdot E, & \text{if } D > \text{threshold;} \\ T_{p-1}, & \text{if } D \leq \text{threshold.} \end{cases} \tag{6}$$

In this formula, $\theta_1 + \theta_2 + \theta_3 + \theta_4 = 1$. F presents the number of consecutive same sensing out of a predefined number whose value varies between 0 and 1, T_{p-1} is the last trust value in the previous round, D is the deviation from normal value of sensing, R is the number of misreading and E is whether the node can be found by its parent. θ_2 and θ_3 are reward and penalty coefficient respectively whose value can vary between 0 and 1. E is deduced from ${}^{E_r}/_{E_i}$ where E_r is remaining energy and E_i is initial energy. When ${}^{E_r}/_{E_i}$ is lower than that can support transmit, E is set as -1, otherwise, E is set as 0.

Once T_p is equal to 0, the node is regarded as faulty to be deleted from the network. Once E_r/E_i is equal to 0, the node is out of work to be deleted.

In order to keep consistent with T_I, T_p should be multiplied by 100 to get an integer too.

4 Routing algorithm based on multi-factors

4.1 Double-weight trust diagram

When consider nodes in one cluster, we get a $G = (V, E, W_v, W_E)$ consisting of vertexes V, edges E and weight W. Each vertex is a node and each edge is the connection of two neighbors. We set private trust of a node as the vertex weight and interactive trust as the edge trust. The Figure 3 is a double-weight trust diagram.

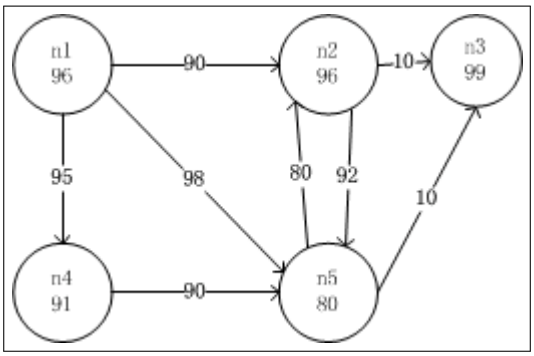


Figure 3: Double-weight trust diagram

In this diagram, node $n1$ has its private trust 0.96 and three interactive trusts 90, 95, 95 with $n1$, $n4$ and $n5$ respectively. It is assumed that $n3$ is the cluster head, when $n1$ transmits its sensing data to cluster head, it can route as $\{1, 2, 3\}$, $\{1, 5, 3\}$, $\{1, 4, 5, 3\}$, $\{1, 2, 5, 3\}$, $\{1, 5, 2, 3\}$ and $\{1, 4, 5, 2, 3\}$. Here, we do not consider circle since the circle can't increase the trust of a routing. In order to select the most trustful routing, we must compute the trust of each routing.

4.2 Routing trust

In [18], Chen et al propose a matrix-based computing method for max-mean measurement. The model defines a series of matrices $S^k = (s^k_{uv})_n$ for max-mean degree inductively as $S^1 = A$ and $S^k = A \odot S^{k-1}$, for any $k2$,

$$s^k_{uv} = \begin{cases} 0, & u = v; \\ {}^1\!/_k \max\{a_{ur} \oplus s^{k-1}_{uv}\}, & \text{otherwise.} \end{cases} \tag{7}$$

Where

$$a \oplus b = \begin{cases} 0, & \min\{a, b\} = 0; \\ a + b, & \text{otherwise.} \end{cases} \tag{8}$$

Our trust diagram in Figure 3 describes the private trust and interactive trust respectively, but, when selecting a route, a combined trust should be considered. We indicate private trust as P_t and interactive trust as I_t, then the combined trust can be computed through formula (9).

$$C_t = \sqrt{P_t \cdot I_t}. \tag{9}$$

The trust matrix for Fig.1 is as Figure 4.

$$\begin{pmatrix} 0 & 93 & 0 & 95 & 95 \\ 0 & 0 & 92 & 0 & 92 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 91 \\ 0 & 80 & 30 & 0 & 0 \end{pmatrix}$$

Figure 4: Trust matrix for figure.3

According to formula (7) and (8), the six routing trust is shown in Table 3.

Table 3: routing trust for Figure.3

Routing	$\{1,2,3\}$	$\{1,5,3\}$	$\{1,4,5,3\}$	$\{1,2,5,3\}$	$\{1,5,2,3\}$	$\{1,4,5,2,3\}$
Trust	92	93	72	71	89	89

It is clearly that, if precise is prior to length, routing $\{1,5,3\}$ should be selected as routing between $n1$ and the cluster head $n3$.

5 Application of our model in fault detection

One of the critical tasks in designing a wireless sensor network is to monitor, detect, and report various useful occurrences of events in the network domain which is determined by the result of data aggregation. But sensor nodes are neither reliable nor stabile due to outer factors as environment and inner factors as energy. Then fault detection is critical to the efficiency of data aggregation scheme. In our former work [18], we present an improved k-means data aggregation algorithm considering the proposal of outliers. Each cluster includes three types of sets: aggregation data set, fault data set and abnormal data set. Abnormal nodes can be detected according to the aggregation result. But the detection of fault nodes is completed in a hierarchical structure till the level of sink.

When trust value reserved as an attribute of a node, the detection of fault node and outlier during data aggregation can deduce from trust values. If a node's private value is higher than a threshold, it is regarded as a normal node. When nodes are not regarded as normal, they may be outlier or abnormal nodes which must be recognized with using interactive value. If its interactive trust is higher than a threshold while its private trust is low, it may be the case that the node is located on the edge of the event area and detects an event such as fire or insect pest. Otherwise, it must be a fault node. The process is described in Figure 5.

6 Validations and evaluations

6.1 Properties of T_I

As interactive trust value is depended on multi-factors, the value of each factor should impact the trust value. That is to say the increasing of factors' value can lead to the increasing of interactive trust value. In another aspect, impacting of one factor must be limited. Generally, trust value is set as a real that lower than 1. We can prove the property above.

Assertion 1. $T_I \leq 1$.

Proof: In formula (5): $T_I = \prod_{i=1}^{n} y_i^{\alpha_i} \quad \left(\sum_{i=1}^{n} \alpha_i = 1\right)$, y_i means interaction, data or time.

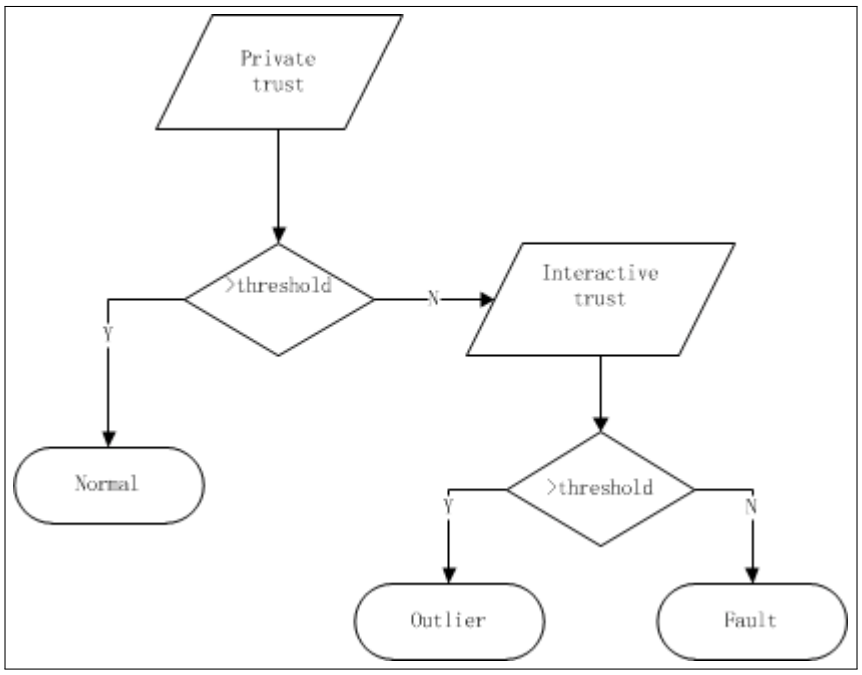


Figure 5: Process of fault detection based on trust

According to formula (1), (2) and (4):
Interaction ≤ 1;
Data ≤ 1;
Time ≤ 1;
$\sum_{i=1}^{n} \alpha_i = 1$, $\alpha_i \leq 1$.
Then $T_I \leq 1$. □

Assertion 2. T_I is monotone.

Proof:

$$\bar{T_I} = \alpha_i \cdot y_i^{\alpha_i - 1} \cdot y_1^{\alpha_1} \cdot \cdots \cdot y_{i-1}^{\alpha_{i-1}} \cdot y_{i+1}^{\alpha_{i+1}} \cdot \cdots \cdot y_n^{\alpha_n} \geq 0.$$

□

This attribute ensure that the increasing of factors' value can lead to the increasing of interactive trust value.

Assertion 3. T_I is agglomerate.

Proof:

$$\bar{\bar{T}}_I = \alpha_i(\alpha_i) \cdot y_i^{\alpha_i - 2} \cdot y_1^{\alpha_1} \cdot \cdots \cdot y_{i-1}^{\alpha_{i-1}} \cdot y_{i+1}^{\alpha_{i+1}} \cdot \cdots \cdot y_n^{\alpha_n} \leq 0.$$

□

This attribute ensure that impacting of one factor is limited.

6.2 Properties and evaluation of T_p

In formula (6), the initial private trust value is 1, with the running of WSN, the value either keeps unchanged or iterates with the penalty and reward coefficients which make the value being lower than 1. If we set $\theta_2 = 0.5$ and $\theta_3 = 0.5$, where the same 0.5 has different impact on correct reading and misreading since the misreading is depicted in exponential manner. When a round

of simulating is set as 10 reading and initial value of trust is set as 1, the private trust value of normal node, event node and fault node is shown in Figure 6. If we only consider data, normal nodes' private trust will not change during the process, while event nodes' will decline at the beginning of event, then raise at the next round since the penalty will offset by the reward. Fault nodes' trust will decline immediately until reaching zero and the nodes will be deleted from the network.

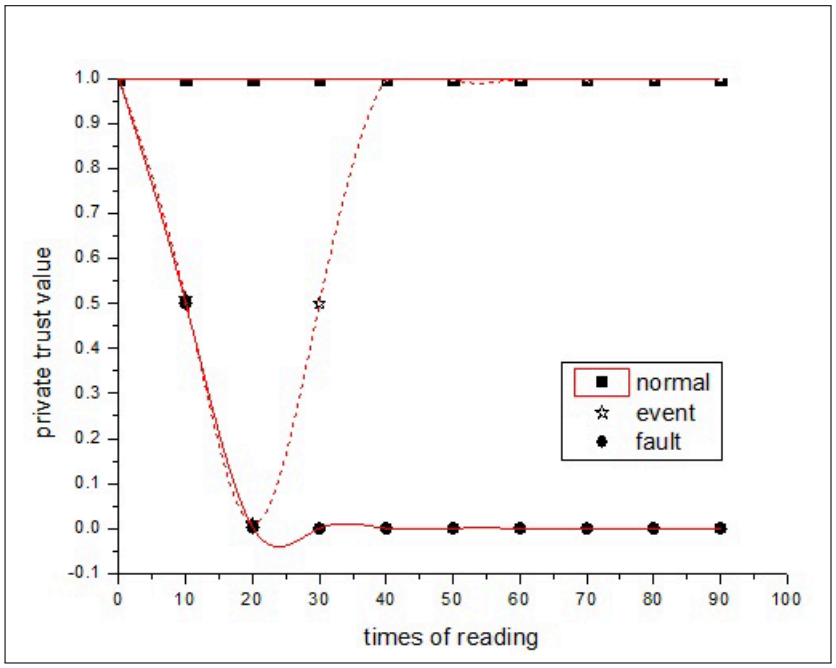


Figure 6: Private trust value for different nodes

6.3 Evaluation of routing selection based on our model

In WSN, it is important to choose a routing protocol, because the efficient routing paths between the sensor node and the sink change with time, especially in case of considering different factors. The above greedy forwarding is a candidate because it is simple and efficient about data transmission. In greedy forwarding, each node just needs to know three pieces of information: the trust value of its own and interaction with neighbors, its location, the location of neighbors. The relative location is displayed in Figure 3 by directed arc, the direction is from nodes farther away from destination to nodes nearer to the destination. The impacts of interaction trust based on channel failure rate and private trust base on sensing failure rate on routing reliability can be described in Figure 7.

Our simulation experiment is based on ns3. Fifty sensor nodes are distributed in a space of 500×700, and the communication radius is set as 60. Each node has two to five neighbors in the experiment and the node's location is already known. The detailed value is shown in Table 4. The comparison of routing reliability for ECCR and our model with different fault rate is shown in Figure 8. Here, reliability displays valid interaction and precise data transmission. We can get that our model has a higher reliability than ECCR in the whole. Especially, when there are fault nodes in routing, our model is effected little in reliability while ECCR's reliability declines a lot just because it does not consider sensing failure caused by faulty nodes. But our model selects normal nodes with high private trust to transmit data to a neighbor node having highest interaction trust with it.

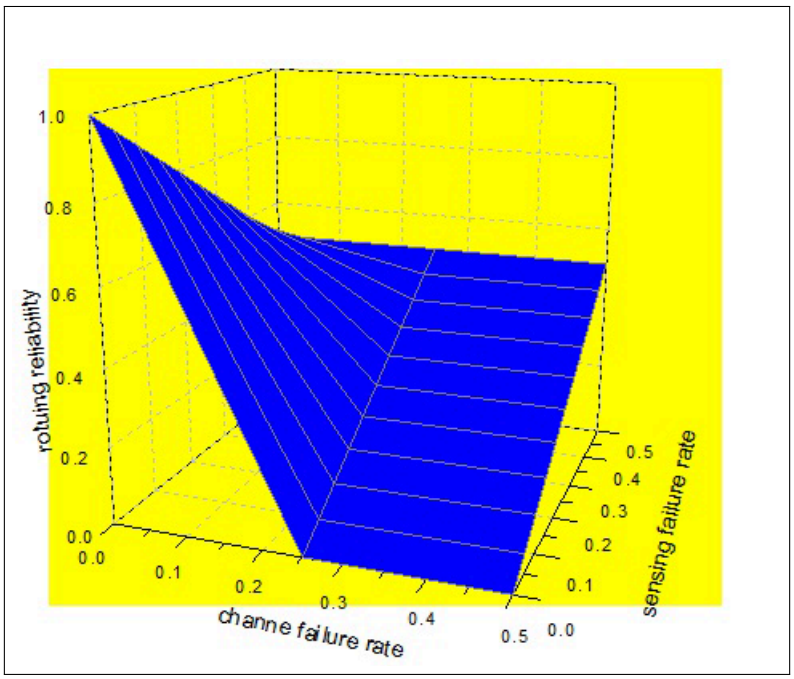


Figure 7: Relationship between reliability and failure rate

Table 4: Values in evaluation

Symbol	Description	values
N	Number of nodes	50
n	Number of CMs in a cluster	6-8
m	Number of CM's neighbors	4-6
θ_1	History trust coefficient	0.9
θ_2	Reward coefficient	0.04
θ_3	Penalty coefficient	0.05
θ_4	Energy coefficient	0.01
t	Threshold for E	0.1

Except for reliability, energy consumption is another important merit to measure the routing protocol based on the trust model. The structure of data is shown in Table 5. Simulation result shown in Figure 9 indicates that the energy consumption of ours models is higher than ECCR because our model keeps a larger size of data than ECCR since ECCR does not consider sensing data. It also indicates that the fault rate impacts a lot in our model because when a node is faulty, its private trust value is computed by the iterated part in formula (6).

Table 5: Structure of data in our model

<table>
<tr><td rowspan="2">Node ID</td><td colspan="2">The size of interaction</td><td colspan="2">The size of similar</td><td colspan="2">The size of trust</td></tr>
<tr><td>$S_{i,j}$</td><td>$F_{i,j}$</td><td>$c_{i,j}$</td><td>$d_{x,y}$</td><td>Interactive</td><td>Private</td></tr>
<tr><td>2 bytes</td><td>1 bytes</td><td>1 bytes</td><td>1 bytes</td><td>1 bytes</td><td rowspan="2">1 bytes</td><td rowspan="2">1 bytes</td></tr>
<tr><td colspan="5">The size of time | The size of sensing data | The size of energy</td></tr>
</table>

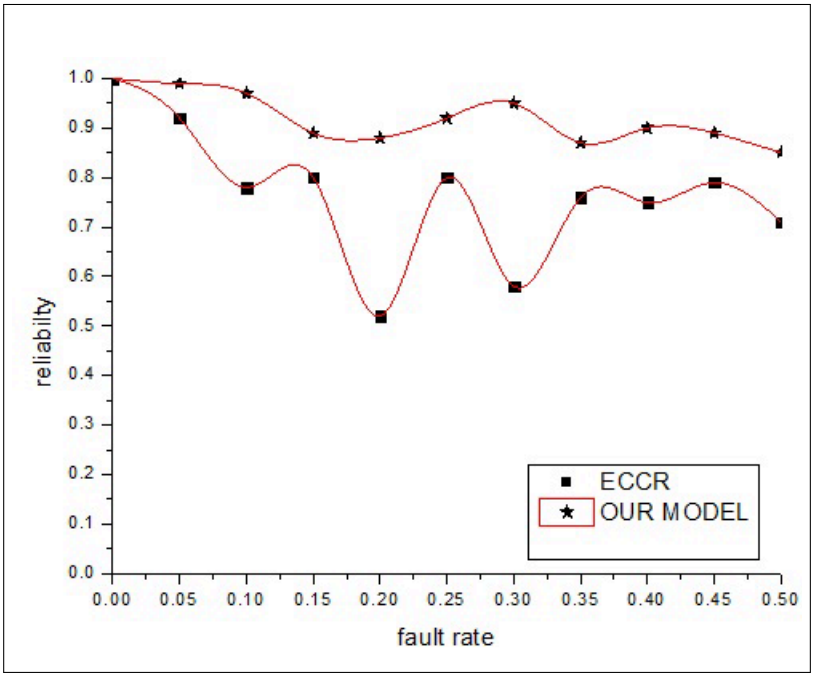


Figure 8: Reliability of different fault rate

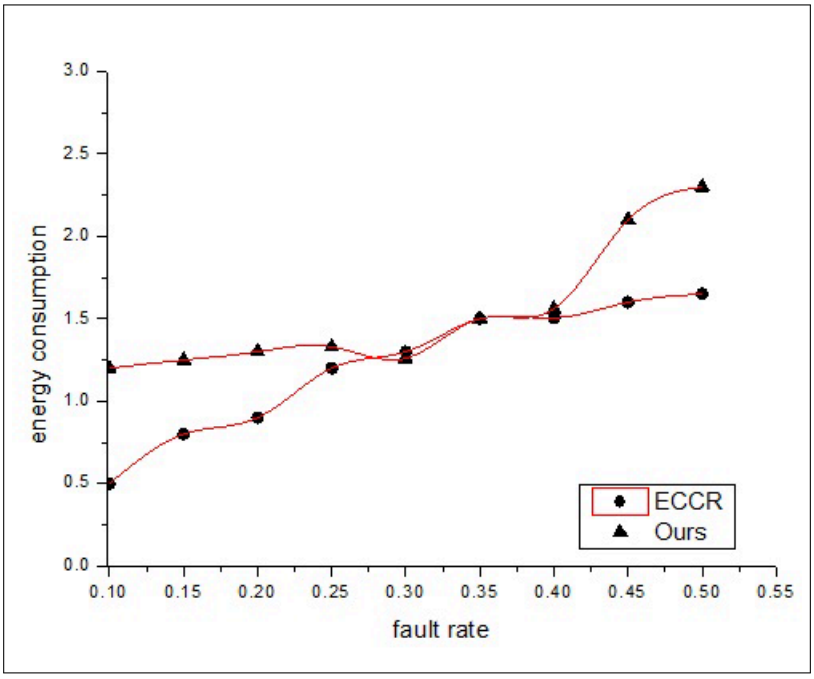


Figure 9: Energy cost of different fault rate

6.4 Evaluation of fault detection based on our model

In this subsection, we compare our model with TMS since TMS is an outstanding trust model considering different factors in WSN. The result is shown in Figure 10 which indicates that the detection of our model is higher than TMS during the running time because we introduce private trust to rapidly judge a fault node. But the fluctuation is larger than TMS due to the temporary malicious judge of event nodes.

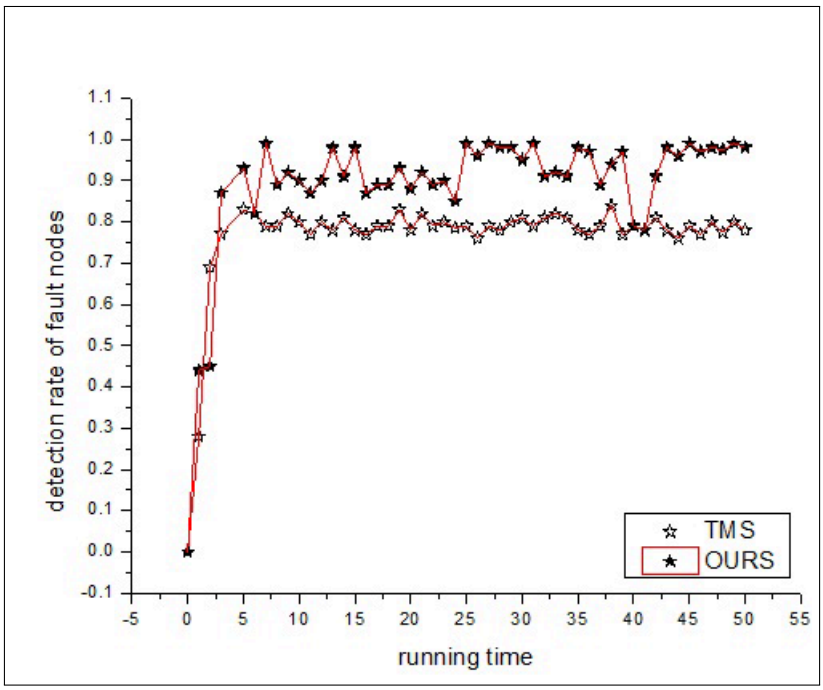


Figure 10: Detection rate within running time

7 Conclusion and future work

In this paper, a trust model based on multi-factors is proposed. It defines multi-factors in WSN to help build trust model including private trust and interactive trust. Private trust focuses on the past record, current record and remaining energy of a node and interactive trust focuses on the interaction of a node with its neighbors. Interactive factors include communication, data and time to keep nodes connective, consistent and synchronized. Private factors include data and energy to ensure a node consistent with itself and keeping active when working. The validation shows the increasing of factors' value can lead to the increasing of interactive trust value and the impacting of one factor is limited. Using the two types of trusts, a routing trust algorithm is proposed that is expressed as a two-weight diagram. With the routing model, a node can transmit its sensing data more accurately to the cluster head or sink than ECCR, but consume more energy. Furthermore, the trust model can be used in fault detection and the simulation results comparing with TMS show that the proposed model can rapidly detect fault and effectively raise fault detection rate with a fluctuation due to event nodes. In the future, we will pay more attention on the application of the model and algorithm this paper proposed to real wireless sensor network, for example, the environment detecting.

Acknowledgements

This work is supported by the National Basic Research Program of China (Grant No. 2011CB302802) and the Innovation Group Project of the National Natural Science Foundation(Grant No. 61321064). The National Natural Science Foundation of China (Grant No. 61370100). Shanghai Knowledge Service Platform Project (No. ZF1213). Wang acknowledges the supported of the key discipline of Shanghai second polytechnic University named software engineering (No.XXKZD1301). The authors would like to thank the referees for their invaluable comments and suggestions.

Bibliography

[1] Osman Khalid, Samee U.Khan(2013); Comparative study of trust and reputation systems for wireless sensor networks, *Secutity and Communication Networks*, 6(6):669-688.

[2] Xiang Gu, Jianlina Qiu, Jina Wang(2012); Research on Trust Model of Sensor Nodes in WSNs, *Procedia Engineering*, 29(2012):909-913.

[3] Renjian Feng, Shenyun Che, Xiao Wang, Ning Yu(2013); Trust Management Scheme Based on D-S Evidence Theory for Wireless Sensor Networks, *International Journal of Distributed Sensor Networks*, 2013(2013):9-18.

[4] K.Hoffman, D.Zage, C.Nita-Rotaru(2009); A survey of attack and defense techniques for reputation systems, *ACM Computing Surveys*, 42(1):1-31.

[5] C.Karlof, D.Wagner(2012); Secure routing in wireless sensor networks, *International Journal of Computer Science Issues*, 9(1): 187Â¨C191.

[6] J.Lopez, R. Roman, C.Alcaraz(2009); Analysis of security threats, requirements, technologies and standards in wireless sensor networks, *Foundations of Security Analysis and Design*, 25(2):289-338.

[7] B.Xiao, B.Yu(2006); Detecting selective forwarding attacks in wireless sensor networks, *In Proceedings of the 20th International Parallel and Distributed Processing Symposium*, 2006(2006):1-8.

[8] Y.Yu, K.Li, W.Zhou, P.Li(2012); Trust mechanisms in wireless sensor networks: attack analysis and countermeasures, *Journal of Network and Computer Applications*, 35(3): 867-880.

[9] E.Aivaloglou, S.Gritzalis, C.Skianis(2008); Trust establishment in sensor networks: behaviour-based, certificate-based and a combinational approach, *International Journal of Systems Engineering*, 16(5): 128-148.

[10] R Feng, X Xu, X Zhou, J Wan(2011); A trust evaluation algorithm for wireless sensor networks based on node behaviors and ds evidence theory, *Sensors*, 11(2):1345-1360.

[11] Riaz Ahmed, S.Jameel(2009); Group-based trust management scheme for clustered wireless sensor networks, *IEEE Transactions on Parallel and Distributed Systems*, 20(11): 1698-1712.

[12] Ganeriwal(2008); Reputation-Based Framework for High Integrity Sensor Networks, *In Proceedings of ACM workshop security of ad hoc and sensor networks*, 4(3): 66-72.

[13] V.R. Sarma Dhulipala, N.Karthik, RM.Chandrasekaran(2013); A Novel Heuristic Approach Based TrustWorthy Architecture for Wireless Sensor Networks, *Wireless Pers Commun*, 18(3):189-205.

[14] Na Wang, YiXiang Chen(2013); A Fault-Event Detection Model Using Trust Matrix in WSN, *Sensors & Transducers journal*, 158(11):190-194.

[15] Na Wang, Yanxia Pang(2014); An improved light-weight trust model in WSN, *Computer Modeling & New Technologies*, 18(4):57-61.

[16] Hongwei Tao, Yixiang Chen(2010); Another Metric Model for Trustworthiness of Softwares Based on Partition, *Advances in Intelligent and Soft Computing*, 82(2010):695-705.

[17] Yixiang Chen, TianMing Bu, Min Zhang, Hong Zhu(2010); Measurement of Trust Transitivity in Trustworthy Networks, *Journal of Emerging Technologies in Web Intelligence*, 2(4):319-325.

[18] Na Wang, YuePing Wu(2013); Data aggregation for failure tolerance in wireless sensor network, *Applied Mechanics and Materials*, 347(2013): 965-969.

[19] C.Intanagonwiwat, R.Govindan, D.Estrin(2000); A scalable and robust communication paradigm for sensor networks, *In Proc. Sixth Annual International Conference on Mobile Computing and Networks*, 28(2000): 238-249.

[20] L Ruan, H W Du, X H Jia, et al(2004); A greedy approximation for minimum connected dominating sets, *Theoretical Computer Science*, 329(1-3): 325-330.

[21] Y Tang, M T Zhou(2007); Maximal independent set based distributed algorithm for minimum connected dominating set, *Acta Electronica Sinica*, 35(5): 868-874.

[22] Machado Kassio, Rosario Denis(2013); A routing protocol based on energy and link quality for internet of things applications, *Sensors*, 13(2):1942-1964.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):263-279, April, 2015.

QEAM: An Approximate Algorithm Using P Systems with Active Membranes

G. Zhang, J. Cheng, M. Gheorghe, F. Ipate, X. Wang

Gexiang Zhang*, Jixiang Cheng
School of Electrical Engineering, Southwest Jiaotong University
Chengdu, 610031, P.R. China
*Corresponding author: zhgxdylan@126.com

Marian Gheorghe
Faculty of Engineering and Informatics, University of Bradford,
Bradford, West Yorkshire BD7 1DP, UK,
m.gheorghe@bradford.ac.uk

Florentin Ipate
Faculty of Mathematics and Computer Science, University of Bucharest
Academiei 14, Bucharest, Romania
florentin.ipate@ifsoft.ro

Xueyuan Wang
School of Information Engineering
Southwest University of Science and Technology
MianYang 621010, P.R.China

Abstract: This paper proposes an approximate optimization approach, called QEAM, which combines a P system with active membranes and a quantum-inspired evolutionary algorithm. QEAM uses the hierarchical arrangement of the compartments and developmental rules of a P system with active membranes, and the objects consisting of quantum-inspired bit individuals, a probabilistic observation and the evolutionary rules designed with quantum-inspired gates to specify the membrane algorithms. A large number of experiments carried out on benchmark instances of satisfiability problem show that QEAM outperforms QEPS (quantum-inspired evolutionary algorithm based on P systems) and its counterpart quantum-inspired evolutionary algorithm.
Keywords: Membrane computing, active membranes, approximate optimization approach, quantum-inspired evolutionary algorithm; satisfiability problem.

1 Introduction

In the last decades, natural computing has been intensively studied and a wide range of applications in computer science and many other areas have been produced. As a well established branch of natural computing, membrane computing, using models called P systems, has made a significant impact on the development of various disciplines [22], such as theoretical computer science, biology, linguistics, etc. The first variants of P systems were proposed in 1998 by G. Păun [19]. They represent a new distributed-parallel framework for designing cell-like or tissue-like computing models, handling multisets of abstract objects in a compartmentalized arrangement of membranes. The membrane structure delimits compartments in a hierarchical or network manner. Objects are arranged as multisets and dispersed across these compartments. Rules are usually associated to the regions enclosed by membranes and control the evolution of objects inside in a maximally parallel way. The main characteristics of P systems are the hierarchical or network architecture of membranes, type of rules (transformation, communication etc.) and intrinsic parallelism, which are all very effective from a computational point of view

and attractive and suitable for modelling various problems. Until now, P systems have been developed principally from a mathematical and computational point of view, building a great variety of computing models and studied for their computational power, complexity aspects and potential solutions to NP-complete problems, and have been utilized for modelling real-world problems in graphics, linguistics, biology. However, the issue of adapting P systems for solving practical problems remains a fundamental aspect of the research in this field, and fortunately a burgeoning interest in this respect for many researchers has been noticeable in the last years [7]. The application of P systems to such problems is still in a developmental phase [22], as compared to evolutionary computation.

Inspired by the evolution in natural selection and molecular genetics, evolutionary algorithms (EAs) have become the most successful metaheuristic search techniques [2]. The great success of EAs in various applications, such as evolutionary optimization and machine learning, can be attributed to two outstanding characteristics: practicability and robustness. EAs are regarded as blind search methodologies without domain specific knowledge [23], suitable for a variety of complex problems in real-world applications. As population-based search tools, EAs usually sample multiple points of the search space in a single step and consequently are quite robust with respect to the objective function landscapes containing many peaks. Developing potential efficient solutions for specific problems is a challenging and attractive topic for researchers from a wide range of areas. Quantum-inspired evolutionary algorithms (QIEAs), one of the three main research areas related to the complex interaction between quantum computing and evolutionary algorithms, are receiving renewed attention [28]. A QIEA is a new evolutionary algorithm for a classical computer rather than for quantum hardware. QIEAs use quantum-inspired bits (Q-bits), quantum-inspired gates (Q-gates) and observation processes to specify their structure and steps. More specifically, Q-bits are applied to represent genotype individuals; Q-gates are employed to operate on Q-bits to generate offspring; and the genotypes and phenotypes are linked by a probabilistic observation process.

Even though P systems and EAs use different rules and computational strategies to handle different objects, both of them are paradigms of natural computing and employed to solve complex problems such as NP-complete problems [27, 34]. P systems represent a suitable formal framework for parallel-distributed computation and EAs are very effective for implementing different algorithms to solve many problems. Thus, the possible interplay between P systems and EAs is very promising for further exploration and represents a fertile research field.

Being the successful instances of this interaction, membrane algorithms can be regarded as a class of hybrid optimization algorithms using the concepts and principles of metaheuristic search methodologies and the hierarchical or network structures of membranes and, to some extent, rules of P systems. When a P system is considered as a parallel-distributed framework for metaheuristic search techniques, it is investigated in terms of optimization results and computation framework, instead of computing power and efficiency. According to the investigations in the literature, there are two main types of membrane algorithms in terms of membrane structures: hierarchical and network. In [30], a tissue membrane system with a network structure was used to appropriately organize five representative variants of differential evolution algorithms. Three principal categories, nested membrane structure (NMS), one-level membrane structure (OLMS) and hybrid membrane structure, were reported with respect to the membrane algorithms with hierarchical membrane structures. In [16], a membrane algorithm with NMS was proposed by using a genetic algorithm and a local search method to solve travelling salesman problems. This kind of membrane algorithms was also applied to solve the min storage problem [13], DNA sequence design problem [24, 25] and the proton exchange membrane fuel cell model parameter estimation problems [26]. In [27], a membrane algorithm integrating OLMS with a QIEA, called QEPS, was proposed to solve knapsack problems and the experiment-based comparisons

between OLMS and NMS were drawn, implying that the choice of the membrane structure is very important for membrane algorithms. This membrane structure was also combined with a QIEA and tabu search [32], differential evolution [3], ant colony optimization [29], particle swarm optimization [33] and multiple QIEA components to solve radar emitter signal time-frequency atom decomposition, numerical optimization problems, travelling salesman problems, broadcasting problems in P systems and image processing, respectively. In [11], a dynamic multi-objective optimization algorithm using a membrane system with a hybrid structure was developed to design a controller for a time-varying unstable plant. The dynamic behavior analysis in [31] indicates that the membrane algorithm, QEPS, has a stronger capability to balance exploration and exploitation than its counterpart approach, QIEA. It is worth pointing out that Păun has made a clear claim that membrane algorithms represent a research directions with a well-defined practical use [22], and therefore further studies are very necessary to prove the use of P systems for solving real-world applications.

P systems with active membranes can produce an exponential growth of membranes and consequently can solve a class of NP-complete problems, such as the satisfiability (SAT) problem [1, 20] and the knapsack problem [18], in a linear or polynomial time. The two types of complete problems were discussed in [1, 18, 20] and in many other places from a mathematical perspective. To the best of our knowledge, no evolutionary algorithm using P system with active membranes has been devised to approximately solve the two aforementioned kinds of problems.

This paper proposes an approximate algorithm combining a P system with active membranes model and a QIEA, called QEAM. This approach is based on the hierarchical arrangement of the compartments and developmental rules (e.g., membrane separation, merging, transformation/communication-like rules) of a P system with active membranes model, and the objects consisting of Q-bit individuals, a probabilistic observation and the evolutionary rules designed with Q-gates to specify the membrane algorithms. In the experiments, the application of QEPS to SAT problems is first discussed, and then QEAM is tested on 65 benchmark SAT problems. Extensive experiments show that QEAM achieves much better results than QEPS and its counterpart QIEA. Also, the parametric and non-parametric tests show significant differences.

2 QEAM

In this section, we start by introducing some concepts related to P systems with active membranes and QIEAs and then describe in detail the proposed QEAM algorithm.

2.1 P systems with active membranes

In this subsection, we give a brief description of P systems with active membranes without polarizations due to [20] and [17], where more details can also be found.

A *membrane structure* is a rooted tree represented by a Venn diagram and is identified by a string of correctly matching parentheses, with a unique external pair of parentheses; this external pair of parentheses corresponds to the external membrane, called *the skin*. A membrane without any another membrane inside (the leaves of the tree) is said to be *elementary*. For example, the structure in Fig. 1 contains 8 membranes; membranes 3, 5, 6 and 8 are elementary. The string of parentheses identifying this structure is $\mu = [_1[_2[_5]_5[_6]_6]_2[_3]_3[_4[_7[_8]_8]_7]_4]_1$.

All membranes are labelled; here we have used the numbers from 1 to 8. We say that the number of membranes is the *degree* of the membrane structure, while the height of the tree associated in the usual way with the structure is its *depth*. In the example above we have a membrane structure of degree 8 and of depth 4.

The membranes delimit *regions*, precisely identified by the membranes (the region of a membrane is delimited by the membrane and all membranes placed immediately inside it, if any such a membrane exists). In these regions we place *objects*, which are represented by symbols of an alphabet. Several copies of the same object can be present in a region, so we work with *multisets* of objects. A multiset over an alphabet V is represented by a string over V, together with all its permutations: the number of occurrences of a symbol $a \in V$ in a string $x \in V^*$ (V^* is the set of all strings over V; the empty string is denoted by λ) is denoted by $|x|_a$ and it represents the multiplicity of the object a in the multiset represented by x.

A *polarizationless P system with active membranes* is a construct

$$\Pi = (V, T, H, \mu, w_1, \ldots, w_m, R),$$

where

1. $m \geq 1$ (the initial *degree* of the system);
2. V is an alphabet (the *working alphabet* of the system);
3. $T \subseteq V$ (the *terminal* alphabet);
4. H is a finite set of *labels* for membranes;
5. μ is a *membrane structure* consisting of m membranes, labelled (not necessarily in a one-to-one manner) with elements of H;
6. $w_1, \ldots, w_m$, are strings over V, describing the *multisets of objects* placed in the m regions of μ;
7. R is a finite set of *developmental rules*, of the following forms:
   (a) $[_h a \to v]_h$, for $h \in H, a \in V, v \in V^*$; (*object evolution rules*, associated with membranes and depending on the label, but not directly involving the membranes, in the sense that the membranes are neither taking part in the application of these rules nor are they modified by them);
   (b) $a[_h]_h \to [_h b]_h$, for $h \in H, a, b \in V$; (*communication rules*; an object is introduced in the membrane, possibly modified during this process);
   (c) $[_h a]_h \to [_h]_h b$, for $h \in H, a, b \in V$; (*communication rules*; an object is sent out of the membrane, possibly modified during this process);
   (d) $[_h]_h[_h]_h \to [_h]_h$, for $h \in H$; (*merging rules for elementary membranes*; in reaction of two membranes, they are merged into a single membrane; the objects of the former membranes are put together in the new membrane);
   (e) $[_h W]_h \to [_h U]_h[_h W - U]_h$, for $h \in H, U \subset W$; (*separation rules for elementary membranes*; the membrane is separated into two membranes with the same labels; the objects from U are placed in the first membrane, those from $W - U$ are placed in the other membrane);

For a detailed description on how to use these rules, refer to [17,20]. It is worth pointing out that these rules are used in the non-deterministic maximally parallel manner, i.e., in any given step, one or more rules of type (a), such that no unallocated object to rules can be allocated to any rule, and/or at most one rule of types (b)-(e) can be applied to each membrane. In this way, we get transition from a configuration of the system to the next configuration. A sequence of transitions forms a computation. A computation is halting if no other rules can be employed in its last configuration.

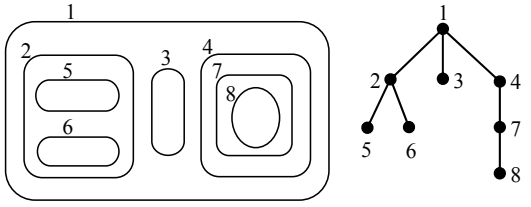


Figure 1: A membrane structure

```
Begin
    t ← 1
(i) Initialize Q(t)
    While (not termination condition) do
(ii)    Make P(t) by observing the states of Q(t)
(iii)   Evaluate P(t)
(iv)    Update Q(t) using Q-gates
(v)     Store the best solutions among P(t)
        t ← t + 1
    End
End
```

Figure 2: Pseudocode algorithm for QIEA [10]

2.2 Quantum-inspired evolutionary algorithms

The interaction of quantum computing and evolutionary algorithms has produced three research avenues: evolutionary-designed quantum algorithms using evolutionary algorithms to design new quantum algorithms, quantum evolutionary algorithms implementing evolutionary algorithms in a quantum computing environment and QIEAs [28]. QIEA is employed to describe the computational methods using concepts and principles of quantum computing for solving various problems in the context of a classical computer [14]. Based on the concepts and principles of quantum computing, such as quantum bit (qubit), quantum gate and superposition, a QIEA is developed as a novel evolutionary algorithm for a classical computer. Narayanan and Moore [15] introduced a preliminary idea of a QIEA and Han and Kim [10] proposed its practical algorithm. A QIEA is characterized by a Q-bit representation, a probabilistic observation and a Q-gate evolutionary rule. In recent years, QIEAs have become a promising and rapidly growing branch of evolutionary computation.

In QIEAs, a Q-bit is defined by a pair of complex numbers $(\alpha,\ \beta)$ as $[\alpha\ \beta]^T$, where $|\alpha|^2$ and $|\beta|^2$ are probabilities that the observation of the Q-bit will render a '0' or '1' state. Normalization requires that $|\alpha|^2 + |\beta|^2 = 1$. Note that QIEAs just need real numbers for amplitudes. Besides '0' and '1' states, a Q-bit can also be in a superposition of the two states. A Q-bit individual is represented as a string of l Q-bits

$$\begin{bmatrix} \alpha_1|\alpha_2|\cdots|\alpha_l \\ \beta_1|\beta_2|\cdots|\beta_l \end{bmatrix}, \tag{1}$$

where $|\alpha_i|^2 + |\beta_i|^2 = 1$ $(i = 1, 2, \cdots, l)$. A Q-gate in a QIEA is defined as a variation operator for updating the Q-bit individuals such as to guarantee that they also satisfy the normalization condition $|\alpha|^2 + |\beta|^2 = 1$ [10].

The basic pseudocode algorithm for a QIEA is shown in Fig. 2 and the description for each step is as follows.

1. In the "initialize $Q(t)$" step, a population $Q(1)$ with n Q-bit individuals is generated, $Q(t)$=$\{\boldsymbol{q}_1^t, \boldsymbol{q}_2^t, \cdots, \boldsymbol{q}_n^t\}$, at generation t, where $\boldsymbol{q}_i^t$ $(i = 1, 2, \cdots, n)$ is an arbitrary individual in $Q(t)$, which is represented as

$$\boldsymbol{q}_i^t = \begin{bmatrix} \alpha_{i1}^t|\alpha_{i2}^t|\cdots|\alpha_{il}^t \\ \beta_{i1}^t|\beta_{i2}^t|\cdots|\beta_{il}^t \end{bmatrix}, \tag{2}$$

where l is the number of Q-bits, i.e., the string length of the Q-bit individual. In the initial population, that is when $t = 1$, we have $\alpha_{ij}^t = \beta_{ij}^t = 1/\sqrt{2}$ for all $i = 1, 2, \cdots, n$

and $j = 1, 2, \cdots, l$. This means that all possible states are superposed with the same probability at the beginning.

2. By observing the states $Q(t)$, binary solutions in $P(t)$, where $P(t)$=$\{\boldsymbol{x}_1^t, \boldsymbol{x}_2^t, \cdots, \boldsymbol{x}_n^t\}$, are produced at step t. According to the current probability, either $|\alpha_{ij}^t|^2$ or $|\beta_{ij}^t|^2$ of $\boldsymbol{q}_i^t$, $i = 1, 2, \cdots, n$, $j = 1, 2, \cdots, l$, a classical bit 0 or 1 is generated. Thus, l classical bits can construct a binary solution $\boldsymbol{x}_i^t$ $(i = 1, 2, \cdots, n)$.

3. The fitness value for each binary solution $\boldsymbol{x}_i^t$ $(i = 1, 2, \cdots, n)$ is calculated by using an evaluation function.

4. In this step, all the Q-bit individuals in $Q(t)$ are updated by applying Q-gates. To be specific, the jth Q-bit in the ith Q-bit individual $\boldsymbol{q}_i^t$, $j = 1, 2, \cdots, l$, $i = 1, 2, \cdots, n$, is updated by applying the current Q-gate $\boldsymbol{G}_{ij}^t(\theta)$. As usual, QIEAs use a quantum rotation gate as a Q-gate; this is given by

$$\boldsymbol{G}_{ij}^t(\theta) = \begin{bmatrix} \cos\theta_{ij}^t & -\sin\theta_{ij}^t \\ \sin\theta_{ij}^t & \cos\theta_{ij}^t \end{bmatrix}, \tag{3}$$

   where θ_{ij}^t is an adjustable Q-gate rotation angle.

5. The best solutions among $P(t)$ are selected and stored into $b(t)$.

In QIEAs, the Q-bit representation, which can describe simultaneously multiple genotype states using a linear superposition of states in a probabilistic way, makes the algorithm rather good with respect to population diversity. Q-gate evolutionary rules are executed in the Q-bit probability space to avoid the selection pressure problem of conventional genetic algorithms with selection, crossover and mutation operators. As compared with local search methods and conventional genetic algorithms, a QIEA has good balance between exploration and exploitation so as to obtain stronger global search capability and better convergence. Furthermore, a QIEA is able to exploit the search space for a global solution with a small number of individuals, even with one individual; Q-gate evolutionary rules, which are only related to searching the best solution, are easy to implement in a parallel distributed structure because little information needs to be transmitted and exchanged.

2.3 QEAM

This section will introduce the membrane algorithm, QEAM, combining P systems with active membranes and QIEAs. QEAM uses a dynamic P systems-like framework, which is initially randomly produced and then may be changed in the process of evolution. This framework directly uses some of the elements of a P system with active membranes, whereas others are slightly adapted for this evolutionary algorithm. The objects employed will be organized in multisets of special strings built either over the set of Q-bits or $\{0, 1\}$. The rules will be responsible to make the system evolve and select the best fit Q-bit individuals.

More precisely, the dynamic P system-like framework will consist of:

1. a dynamic structure $[_0[_1]_1, [_2]_2, \cdots, [_m]_m]_0$ with m regions contained in the skin membrane, denoted by 0, where m is a number varied during the evolution process;

2. an alphabet that consists of all possible Q-bits and the set $\{0, 1\}$;

3. a set of terminal symbols, $T = \{0, 1\}$;

4. initial multisets
   $w_0 = \lambda$, $w_1 = q_1 q_2 \cdots q_{n_1}$, $w_2 = q_{n_1+1} q_{n_1+2} \cdots q_{n_2}$, $\cdots$, $w_m = q_{n_{(m-1)}+1} q_{n_{(m-1)}+2} \cdots q_{n_m}$, where q_i, $1 \leq i \leq n$, is a Q-bit individual; n_j, $1 \leq j \leq m$, is the number of individuals in w_j; $\sum_{j=1}^{m} n_j = n$, where n is the total number of individuals in this computation;

5. rules include the types of (a)-(e) in P systems with active membranes and their use will be given in the following description.

In what follows, we summarize the steps of QEAM by using a pseudocode notation, shown in Fig. 3, to help presenting the membrane algorithm.

```
    Begin
      Input n, g_max, t_max
      t ← 1
(i)   Initialize the membrane structure (m_t elementary membranes) and objects;
(ii)  While (not termination condition) do
(iii)     Produce g(t);
(iv)      Perform object evolution rule (a) in elementary membranes;
(v)       Perform communication rule (c);
(vi)      Perform object evolution rule (a) in the skin membranes;
          t ← t + 1
(vii)     Determine the number m_{t+1} of elementary membranes;
          If (m_{t+1} < m_t)
(viii)      Perform membrane merging rule (d);
          Else if (m_{t+1} > m_t)
(ix)        Perform membrane separation rule (e);
          End
(x)       Perform communication rule (b);
      End
      Output: the best individual
    End
```

Figure 3: Pseudocode algorithm for QEAM.

1. In the initialization of QEAM, a one level membrane structure $[_0[_1]_1, [_2]_2, \cdots, [_m]_m]_0$ consisting of a skin membrane denoted by 0 and m elementary membranes delimiting m regions inside the skin membrane is constructed as the membrane structure at iteration $t = 1$, where m is a random number ranged from 1 to n, where n is the number of Q-bit individuals. Each Q-bit individual forms an object. Thus, n objects are randomly scattered across the m elementary membranes in a non-deterministic way to make sure that each elementary membrane contains at least one object. So the number of objects in each elementary membrane varies from 1 to $n - m + 1$.

2. The termination condition for QEAM could be a prescribed number of maximal iterations or the algorithm searches the optimal or close-to-optimal solution.

3. This step determines the numbers $\boldsymbol{g}(t) = (g_1, g_2 \ldots, g_m)$, of evolutionary generations for independently performing object evolution rule (a) in the m elementary membranes, where g_i $(i = 1, 2, \ldots, m)$ for the ith elementary membrane is generated randomly between 1 and a certain integer number g_{max}.

4. The steps (ii) to (v) of the QIEA shown in Fig. 2 are performed independently in each elementary membrane to evolve the objects inside. The termination condition for the ith

```
Begin
  If random[0,1) < |α|^2
  Then x ← 0
  Else x ← 1
End
```

Figure 4: Observation process in QIEA

```
Begin
  t ← 0;
  Initialize tabu search;
  While (t < t_max) do
    t ← t + 1;
    Search the neighborhood;
    Evaluate candidate solutions;
    Update tabu list.
  End
End
```

Figure 5: Pseudocode algorithm for tabu search

elementary membrane is the maximal number $g_i (i = 1, 2, \cdots, m)$ of evolutionary generations. It is worth pointing out that the observation process in QIEAs, illustrated in Fig. 4, is applied to build a connection between a Q-bit $[\alpha\ \beta]^T$ and a classical bit and hence to build a link between Q-bit individuals and binary solutions. The Q-gate update procedure

$$\begin{bmatrix} \alpha\prime \\ \beta\prime \end{bmatrix} = G(\theta) \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \tag{4}$$

is used to transform a current Q-bit $[\alpha\ \beta]^T$ into the corresponding Q-bit $[\alpha\prime\ \beta\prime]^T$ at the next generation. The rotation angle θ in the Q-gate $G(\theta)$ in (4) is defined as $\theta = s(\alpha, \beta) \cdot \Delta\theta$, where $s(\alpha, \beta)$ and $\Delta\theta$ can be obtained from the lookup table in [10].

5. The communication rule is employed to send the best binary solution in each elementary membrane out to the skin membrane. This step is helpful to exchange information among the objects in the elementary membranes and the skin membrane because the QIEA employs Q-gates, which are related to only the best individual searched, to generate the offspring. After this step there are m binary solutions in total in the skin membrane.

6. In the skin membrane, a local search, tabu search [8, 12], is performed on the best binary solution selected from the m binary solutions, which are sent from the m elementary membranes (see Step (v)). The pseudocode algorithm for tabu search is shown in Fig. 5. In the "Initialize tabu search" step, an empty tabu list is constructed and tabu length is set to a value. At each iteration, the neighborhood of the best binary solution in the skin membrane is explored to obtain candidate solutions. Next, the candidate solutions are evaluated by using the fitness function and the best of them is selected to update the tabu list.

7. The number m_{t+1} of elementary membranes at iteration $t+1$ is produced randomly between 1 and n, which will directly determine the membrane structure at the next iteration.

8. If $m_{t+1} < m_t$, the $(m_t - m_{t+1})$ elementary membranes will be merged into the m_{t+1} elementary membranes. The merging process is shown in Fig. 6, where EM represents elementary membranes. In each merging operation, we first choose any two arbitary elementary membranes i and j from M elementary membranes, i.e., $1 \leq i, j \leq M$ and $i \neq j$; and then we merge the elementary membranes i and j into a single membrane and put the objects in the elementary membranes i and j into the merged membrane. The initial value of M is m_t. Thus, multiple membranes may be merged into a single membrane. So this rule is a multi-merging one.

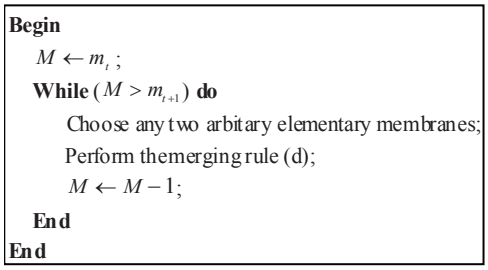


Figure 6: Merging process of EMs

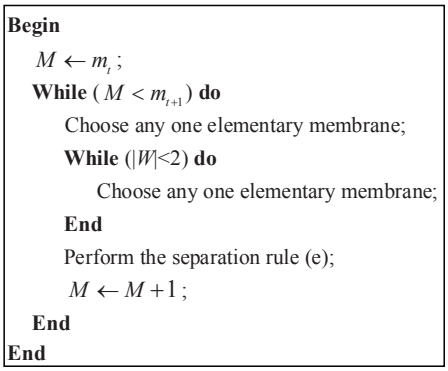


Figure 7: Separation process of EMs

9. If $m_{t+1} > m_t$, the $(m_{t+1} - m_t)$ elementary membranes will be separated into two membranes. The separation process is illustrated in Fig. 7, in which $|W|$ is the number of objects in the pre-separation membrane. We choose any one elementary membrane i which has at least two objects from M elementary membranes, i.e., $1 \leq i \leq M$. The initial value of M is m_t. When the separation rule is performed, $|U|$ ($|U| < |W|$) objects are placed in the first membrane and the $|W| - |U|$ objects are placed in the other membrane. Thus, a single membrane may be divided into several membranes. So, this rule is a multi-separation one.

10. By performing the communication rule (b) in the skin membrane, this step sends the fittest binary solution to each elementary membrane for the further evolution steps.

3 Experimental results

To test the performances of the presented algorithm, QEAM, we will use the satisfiability problem, which is a well-known NP-complete problem, to conduct the experiments. We start from the description of the satisfiability problem, and then turn to use QIEAs and QEPS as benchmark algorithms to solve 65 representative instances of the satisfiability problem. Finally, QEAM is tested on the same instance of the satisfiability problem to draw conclusions.

3.1 Satisfiability problem

The satisfiability problem (SAT) is a fundamentally paradigmatic problem in artificial intelligence applications, automated reasoning, mathematical logic, and related research areas [5]. SAT can be described as follows: given a Boolean formula in conjunctive normal form (CNF), determine whether or not it is satisfiable, that is, whether there exists an assignment to its variables on which it evaluates to true, i.e., a SAT instance is to search a variable assignment $\boldsymbol{x}$ so that a Boolean formula $f(\boldsymbol{x})$ becomes true, where $\boldsymbol{x}$ is a set of Boolean variables $x_1, x_2, \cdots, x_n$, i.e., $x_i \in \{0,1\}$, $i = 1, 2, \cdots, n$ and the propositional formula $f(\boldsymbol{x})$ is in a conjunctive normal form, i.e.,

$$f(\boldsymbol{x}) = c_1(\boldsymbol{x}) \wedge c_2(\boldsymbol{x}) \wedge \cdots \wedge c_m(\boldsymbol{x}), \tag{5}$$

where each clause $c_j(\boldsymbol{x})$, $j = 1, 2, \cdots, m$, is a disjunction of literals, and a literal is a variable or its negation [9]. A SAT instance is called *satisfiable* if such $\boldsymbol{x}$ exists, and *unsatisfiable* otherwise. In this paper only 3-SAT problems, in which each clause has exactly three literals, will be considered

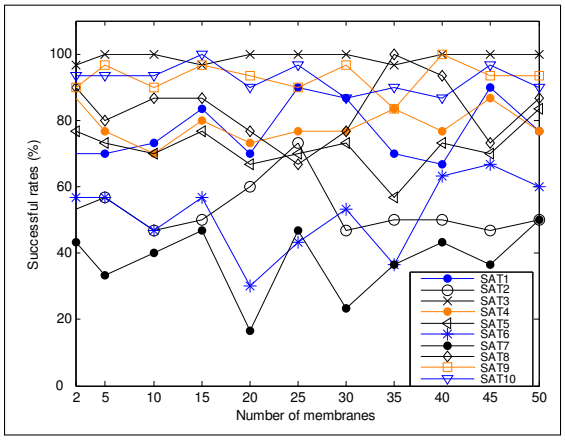


Figure 8: Successful rates

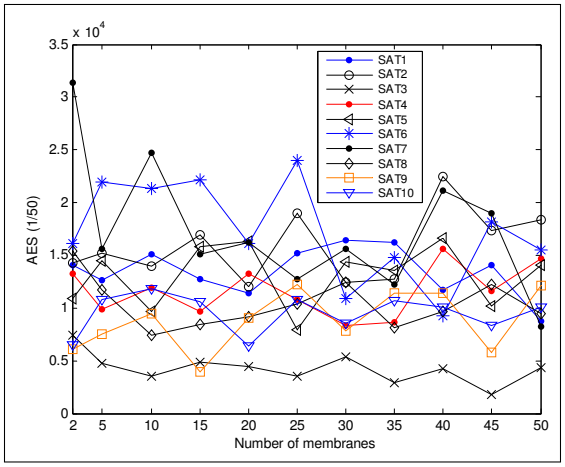


Figure 9: AES

because a number of other problems, such as the travelling salesman problem and the n-queens problem, can be reformulated with respect to 3-SAT problems. In [4] it is shown that 3-SAT problem is NP-complete.

In membrane computing, various types of P systems with membrane division are frequently investigated from a mathematical point of view to obtain an exponential working space in a linear time to solve the SAT problem [1, 20]. This paper will use an approximate algorithm to solve the SAT problem, in which the number of clauses that are not satisfied by the variable assignment $\boldsymbol{x}$ is considered as the evaluation function.

3.2 Results of QEPS and QIEA

According to the study in [27], the number of elementary membranes has a significant impact on the QEPS performances. So we first focus on how to set the number of elementary membranes in an empirical way. Ten benchmark 3-SAT problems[1], each of which has 20 Boolean variables and 91 clauses, are applied to conduct the experiments. The fitness function is the number of clauses that are not satisfied by the variable assignment. The population size n is set to 50. The values of 2, 5, 10, 15, 20, 25, 30, 35, 40, 45 and 50, for the number m of elementary membranes, are used in the experiments. According to previous investigations regarding the effect of the number $g_i (i = 1, 2, \ldots, m)$ of iterations on the QEPS performances [27], the parameter $g_i (i = 1, 2, \ldots, m)$ is set to a uniformly random integer ranged from 1 to 10. The algorithm stops when either 2.75×10^6 evaluation steps are made or the SAT problem solution is found, i.e., the minimal fitness value 0 is attained. The performances of the above 11 cases are evaluated by using the successful rate of 30 independent runs (the percentage of the runs making the SAT problem satisfiable) and the average number of evaluations to solutions (AES) over the successful runs. The experimental results are listed in Fig. 8 and Fig. 9, which illustrate that the successful rates and the AES vary with the number of elementary membranes.

As shown in Fig. 8 and Fig. 9, the successful rates and the AES show a broad range of variability with respect to the number of different elementary membranes; this indicates that the number of elementary membranes has a significant impact on the QEPS performances. In order to obtain a balance between the successful rates and the AES, the number of elementary membranes could be fixed at 15.

[1]SATLIB - The Satisfiability Library, http://www.satlib.org/

QIEA is also applied to conduct the experiments on the 10 SAT problems. In these experiments, QIEA employs the same population size and stopping criteria as the QEPS. The statistical results of 30 independent runs for each problem are listed in Table 1. The best experimental results of QEPS are also shown in Table 1, where each of which has 20 Boolean variables and 91 clauses; SR and AES represent successful rates and average number of evaluations to solutions, respectively.

To perform convincing comparisons between QIEA, QEPS and QEAM, additional fifty-five 3-SAT benchmark problems are employed to carry out experiments. Both QEPS and QIEA use 50 individuals as a population, the prescribed number of 2.75×10^6 evaluations to solutions as the stopping criterion and the number of clauses that are not satisfied by the variable assignment as the fitness function. In QEPS, the parameter $g_i (i = 1, 2, \ldots, m)$ is set to a uniformly random integer ranged from 1 to 10, and the number of elementary membranes is assigned to 15. The performances of the two algorithms are evaluated by using the following criteria: the mean of the solutions over 15 runs and their standard deviations. It is worth pointing out that the experiments are very time-consuming and therefore only 15 independent runs are performed for each SAT problem. The number of Boolean variables, the number of clauses in each Boolean formula and the experimental results are provided in Table 2.

Table 1: Comparisons of QIEA, QEPS and QEAM on 10 SAT problems

	QIEA		QEPS		QEAM	
Problems	SR(%)	AES	SR(%)	AES	SR(%)	AES
SAT1	77	572750	100	528850	100	220804
SAT2	70	496700	90	701200	100	300468
SAT3	53	638250	73	949400	97	279174
SAT4	100	218250	100	90300	100	59978
SAT5	87	575900	87	582300	100	179445
SAT6	57	469650	83	701700	100	336728
SAT7	53	1137750	67	907300	93	527795
SAT8	27	1120300	50	410300	100	354903
SAT9	87	684800	100	405450	100	128633
SAT10	93	281050	100	572750	100	115876

3.3 Results of QEAM

In the experiments for testing QEAM performance, the population size and the prescribed number of evaluations of solutions as the stopping criterion are set to 50 and 2.75×10^6, respectively, which are the same as those in QIEA and QEPS. QEAM applies the same g_{max} as QEPS. Additionally, the tabu length and t_{max} in QEAM are 5 and 100, respectively. For each of the first 10 benchmark 3-SAT problems shown in Table 2, we performed 30 independent runs and recorded the successful rate and the AES over successful runs. The experimental results are provided in Table 1. The QEAM performance is further investigated by using the remaining 55 3-SAT benchmark problems. We record the average solution and the standard deviations over 15 runs for each of them. The experimental results are listed in Table 2, where each of the first ten, the second ten, the third ten, the fourth ten, the fifth ten and the last five problems has 50, 75, 100, 125, 150 and 250 Boolean variables and 218, 325, 430, 538, 645 and 1065 clauses, respectively; Mean and Std represent the mean of the best solutions and the standard deviation of the best solutions, respectively; (+) represents significant difference.

Table 2: Comparisons of QIEA, QEPS and QEAM using 55 instances of the SAT problem (to be continued).

	QIEA		QEPS		QEAM		QEAMvs.QIEA		QEAMvs.QEPS	
SAT	Mean	Std	Mean	Std	Mean	Std	t-test	Imp.(%)	t-test	Imp.(%)
1	7.67	0.82	6.60	1.18	0.93	0.26	5.23e-23(+)	+87.87	5.30e-17(+)	+85.91
2	8.27	2.12	7.40	1.12	1.53	0.64	2.32e-12(+)	+81.50	1.13e-16(+)	+79.32
3	7.40	1.40	6.27	0.88	1.00	0.53	5.90e-16(+)	+86.49	5.64e-18(+)	+84.05
4	7.87	1.19	6.33	0.98	1.00	0.38	7.31e-19(+)	+87.29	5.74e-18(+)	+84.20
5	7.13	1.41	6.20	0.86	0.07	0.26	1.30e-17(+)	+99.02	2.50e-21(+)	+98.87
6	7.27	0.80	5.93	0.80	0.20	0.41	5.39e-23(+)	+97.25	1.53e-20(+)	+96.63
7	7.73	1.16	6.40	1.18	1.07	0.46	1.73e-18(+)	+86.16	8.26e-16(+)	+83.28
8	8.73	0.70	7.67	1.18	1.53	0.83	5.99e-21(+)	+82.47	6.01e-16(+)	+80.05
9	7.87	1.13	6.87	1.19	1.47	0.64	1.27e-17(+)	+81.32	2.84e-15(+)	+78.60
10	8.33	0.82	6.93	0.88	1.07	0.59	5.74e-22(+)	+87.15	7.33e-19(+)	+84.56
11	16.67	1.11	15.40	0.99	2.00	0.93	5.05e-26(+)	+88.00	9.32e-26(+)	+87.01
12	15.07	1.49	14.60	0.74	1.47	0.74	1.76e-23(+)	+90.25	1.38e-28(+)	+89.93
13	16.33	0.82	14.80	2.01	2.07	0.88	6.61e-28(+)	+87.32	1.84e-19(+)	+86.01
14	14.00	1.20	12.87	1.60	1.87	0.64	1.53e-24(+)	+86.64	1.41e-20(+)	+85.47
15	15.07	1.03	14.87	0.92	1.60	1.06	9.13e-25(+)	+89.38	3.01e-25(+)	+89.24
16	16.13	1.06	14.87	1.19	2.20	0.68	4.29e-27(+)	+86.36	5.80e-25(+)	+85.21
17	15.33	1.40	14.53	1.64	1.67	0.90	1.54e-23(+)	+89.11	2.00e-21(+)	+88.51
18	15.73	1.44	14.53	1.51	2.20	0.86	2.54e-23(+)	+86.01	8.03e-22(+)	+84.86
19	14.93	1.49	13.80	1.97	1.80	0.41	6.02e-24(+)	+87.94	9.25e-20(+)	+86.96
20	14.40	1.45	13.93	1.22	2.00	0.76	1.48e-22(+)	+86.11	1.19e-23(+)	+85.64
21	24.53	1.81	22.93	1.39	3.0	0.93	1.44e-26(+)	+87.77	5.29e-28(+)	+86.92
22	24.20	1.21	22.80	2.14	3.33	0.62	4.78e-31(+)	+86.24	3.08e-24(+)	+85.39
23	24.27	1.28	22.93	1.79	4.20	0.94	1.15e-28(+)	+82.69	6.05e-25(+)	+81.68
24	23.40	1.68	22.80	1.08	3.53	0.92	2.63e-26(+)	+84.91	1.51e-29(+)	+84.52

Table 2 Comparisons of QIEA, QEPS and QEAM (continued)

	QIEA		QEPS		QEAM		QEAM vs. QIEA		QEAM vs. QEPS	
SAT	Mean	Std	Mean	Std	Mean	Std	t-test	Imp.(%)	t-test	Imp.(%)
25	24.27	1.22	22.80	1.47	3.73	0.88	1.45e-29(+)	+84.63	4.13e-27(+)	+83.64
26	24.00	1.51	22.47	1.60	3.60	0.74	3.53e-28(+)	+85.00	1.06e-26(+)	+83.98
27	24.13	1.06	23.40	1.35	3.87	0.74	2.99e-31(+)	+83.96	1.08e-28(+)	+83.46
28	24.00	1.85	23.40	1.30	3.73	0.80	6.33e-26(+)	+84.46	6.40e-29(+)	+84.06
29	25.13	1.68	24.13	2.00	4.27	1.28	1.06e-25(+)	+83.01	9.18e-24(+)	+82.30
30	22.93	2.15	22.27	1.98	3.40	0.74	4.81e-24(+)	+85.17	1.64e-24(+)	+84.73
31	33.53	1.96	32.87	1.51	5.67	0.90	6.07e-29(+)	+83.09	3.87e-31(+)	+82.75
32	33.80	1.90	32.07	2.12	5.40	1.06	6.07e-29(+)	+84.02	2.76e-27(+)	+83.16
33	34.47	1.81	33.73	1.33	6.07	1.33	4.37e-28(+)	+82.39	1.85e-30(+)	+82.00
34	34.06	1.84	33.27	2.34	5.13	0.74	1.13e-30(+)	+84.94	1.78e-27(+)	+84.58
35	34.67	1.76	33.60	2.20	5.20	1.32	2.25e-29(+)	+85.00	4.32e-27(+)	+84.52
36	34.80	2.21	33.93	1.44	6.20	1.47	9.51e-27(+)	+82.18	1.93e-29(+)	+81.73
37	32.80	2.86	32.93	1.33	5.27	1.28	2.49e-24(+)	+83.93	1.05e-30(+)	+84.00
38	32.80	1.97	32.47	1.19	5.27	1.10	3.02e-28(+)	+83.93	4.14e-32(+)	+83.77
39	34.20	2.08	33.40	1.76	5.67	0.98	1.78e-28(+)	+83.42	1.09e-29(+)	+83.02
40	33.93	1.67	33.20	2.34	5.67	0.98	1.96e-30(+)	+83.29	7.19e-27(+)	+82.92
41	42.60	1.50	41.20	2.01	6.87	1.13	1.29e-33(+)	+83.87	1.13e-30(+)	+83.33
42	43.07	1.67	40.00	2.67	5.40	0.91	4.16e-34(+)	+87.46	2.66e-28(+)	+86.50
43	41.73	1.71	41.53	1.06	6.27	1.75	2.55e-30(+)	+84.97	2.08e-32(+)	+84.90
44	42.80	3.28	41.07	1.94	6.47	1.25	2.73e-26(+)	+84.88	1.01e-30(+)	+84.25
45	43.93	1.67	42.60	2.64	7.13	1.06	2.38e-33(+)	+83.77	1.66e-28(+)	+83.26
46	43.00	3.23	42.07	1.62	7.60	1.30	4.55e-26(+)	+82.33	6.09e-32(+)	+81.93
47	43.20	2.04	42.73	1.87	7.87	1.30	2.12e-30(+)	+81.78	5.60e-31(+)	+81.58
48	44.27	2.19	43.60	2.32	7.47	1.06	7.50e-31(+)	+83.13	4.97e-30(+)	+82.87
49	44.67	2.26	43.53	2.10	8.93	1.67	9.21e-29(+)	+80.01	6.37e-29(+)	+79.49

Table 2 Comparisons of QIEA, QEPS and QEAM (continued)

	QIEA		QEPS		QEAM		QEAM vs. QIEA		QEAM vs. QEPS	
SAT	Mean	Std	Mean	Std	Mean	Std	t-test	Imp.(%)	t-test	Imp.(%)
50	43.13	1.55	41.47	2.50	6.53	0.99	3.87e-34(+)	+84.86	5.45e-29(+)	+84.25
51	83.07	3.45	81.27	2.91	12.93	1.33	1.48e-33(+)	+84.43	5.51e-35(+)	+84.09
52	83.40	3.44	82.73	2.96	15.27	1.67	8.05e-33(+)	+81.69	4.07e-34(+)	+81.54
53	83.00	2.73	81.60	2.50	12.67	1.50	1.05e-35(+)	+84.73	3.04e-36(+)	+84.47
54	85.13	3.23	83.60	3.14	15.80	1.66	1.15e-33(+)	+81.44	1.14e-33(+)	+81.10
55	84.87	2.83	80.80	2.31	14.20	1.52	2.22e-35(+)	+83.27	1.76e-36(+)	+82.43

According to these experimental results, we employ statistical techniques to analyze the behaviour of the three algorithms over the 55 instances of the SAT problem. There are two statistical methods: parametric and non-parametric [6]. The former, also called single-problem analysis, uses a parametric statistical analysis t-test to analyse whether there is a significant difference between the two algorithms solving the optimization problem. The latter, also called multiple-problem analysis, applies non-parametric statistical tests such as Wilcoxon's and Friedman's tests, to compare different algorithms whose results represent average values for each problem, regardless of the inexistence of relationships among them. Therefore, a 95% confidence Student t-test is first applied to check whether the number of false clauses of the two pairs of algorithms, QEAM vs. QIEA and QEAM vs. QEPS, are significantly different or not. Furthermore, the percentage of improvement (%) in the average number of false clauses due to the QEPS algorithm over QIEA and QEPS is also listed in Table 2. Then two non-parametric tests, Wilcoxon's and Friedman's tests, are employed to check whether there are significant differences between the two pairs of algorithms, QEAM vs. QIEA and QEAM vs. QEPS. The level of significance considered is 0.05. The results of Wilcoxon's and Friedman's tests are shown in Table 3. The symbols + and - in Tables 2–3 represent significant difference and no significant difference, respectively.

In the experiments carried out on the QEAM, the QEPS and the QIEA, we also record the average elapsed time for the first ten SAT problems over 30 independent runs and for the remaining 55 instances of the SAT problem over 15 independent runs. The comparisons of the three algorithms are illustrated in Fig. 10. The x-axis and y-axis represent the number of SAT problems and the elapsed time, respectively.

As shown in Table 1, the QEAM greatly outperforms the QIEA and the QEPS in terms of the successful rates and the average number of evaluations. Also, Table 1 shows that QEPS obtains higher successful rates and smaller average number of evaluations than QIEA. It can be seen from the experimental results of 55 SAT bench problems in Table 2 that the QEAM achieves much better results than the QIEA and the QEPS. The QEPS obtains better results than the QIEA in 54 out of 55 cases. The t-test results demonstrate that there are 55 significant differences between the two pairs of algorithms, QEAM vs. QIEA and QEAM vs. QEPS. The p-values of the two non-parametric tests in Table 3 are far smaller than the level of significance 0.05, which indicates that the QEAM really outperforms the QIEA and the QEPS by introducing the framework and some rules of P systems with active membranes. It is worth noting that the study in [6] shows that the non-parametric statistical tests are more appropriate than parametric statistical tests in the analysis of the behaviour of the evolutionary algorithms over multiple optimization problems.

The QEPS uses the framework and some evolution rules of P systems. Each elementary membrane evolves for a certain number of generations in a non-deterministic way, and then all elementary membranes communicate in the skin membrane. Thus, the QEPS has better population diversity and the capability to balance exploration and exploitation. Consequently the QEPS obtains better results and smaller elapsed time, shown in Fig. 10, than the QIEA. The QEAM goes further and applies the framework and some evolution rules of P systems with active membranes. The good performance of the QEAM is

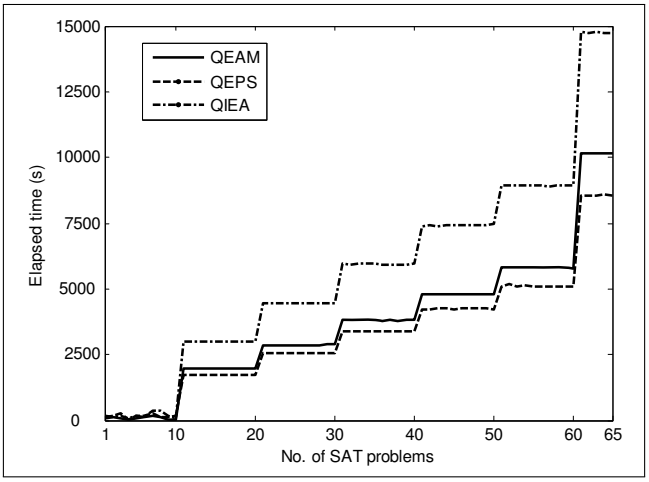


Figure 10: Comparisons of elapsed time.

due to the combination of independent evolution of each elementary membrane in a non-deterministic way, communication in the skin membrane, membrane separation and merging, and a local search in the skin membrane. Figure 10 shows that the QEAM and QEPS consumes less time than QIEA, which indicates that the use of evolution rules of P systems in the QEAM and QEPS has little effect on the overall computational load. Furthermore, the QEAM and QEPS may use a slightly smaller number of evaluations of the solutions than the QIEA because of the randomness of evolutionary generations for each elementary membrane. Additionally, as a result of the use of membrane separation and division, the QEAM consumes slightly more time than the QEPS, which is shown in Fig. 10.

Table 3: Results of non-parametric statistical tests for the two pairs of algorithms, QEAM vs. QEPS and QEAM vs. QIEA, in Table 2. The symbol + represents significant difference.

Tests	QEAM vs. QIEA	QEAM vs. QEPS
Wilcoxon test (p-value)	1.21e-13 (+)	1.21e-13 (+)
Friedman test (p-value)	1.11e-10 (+)	1.11e-10 (+)

4 Conclusions

Membrane algorithms, defined by carefully mixing selected ingredients of P systems and meta-heuristic search methodologies, and the interaction between P systems and quantum computing, are highly promising and give rise to challenging research issues, which are mentioned as open problems and research topics in [7, 21]. Benefiting from the cross-fertilization of ideas from P systems, evolutionary computation and quantum computing areas, this paper discussed a novel membrane algorithm combing P systems with active membranes and QIEA to solve satisfiability problem. A large number of experiments show that QEAM performs better than QEPS and QIEA. As further work, we aim to investigate other interactions between the three disciplines and their applications to specific problems.

Acknowledgments

The work of GZ is supported by the National Natural Science Foundation of China (61170016, 61373047), the Program for New Century Excellent Talents in University (NCET-11-0715) and SWJTU supported project (SWJTU12CX008). The work of MG and FI was partially supported by a grant of the Romanian National Authority for Scientific Research, CNCS-UEFISCDI, project number PN-II-ID-PCE-2011-3-0688.

Bibliography

[1] Alhazov, A.; Martin-Vide, C.; Pan, L.Q. (2003); Solving a PSPACE- complete problem by recognizing P systems with restricted active membranes, *Fund Inform*, ISSN 0169-2968, (2): 67-77.

[2] Bonissone, P.P.; Subbu, R.; Eklund, N.; Kiehl, T.R. (2006); Evolutionary algorithms + domain knowledge = real-world evolutionary computation, *IEEE T Evolut Comput*, ISSN 1089-778X, 10(3):256-280.

[3] Cheng, J.; Zhang, G.; Zeng, X.(2011); A novel membrane algorithm based on differential evolution for numerical optimization, *Int J Unconv Comput*, ISSN 1548-7199, 7(3):159-183.

[4] Cook, S. (1971); The complexity of theorem-proving procedures, *Proc. of STOC*, 151-158.

[5] Folino, G.; Pizzuti, C.; Spezzano, G. (2001); Parallel hybrid method for SAT that couples genetic algorithms and local search, *IEEE T Evolut Comput*, ISSN 1089-778X, 5(4):323-334.

[6] Garcia, S.; Molina, D.; Lozano, M.; Herrera, F.(2009); A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC 2005 special session on real parameter optimization, *J Heuristics*, ISSN 1381-1231, 15(6):617-644.

[7] Gheorghe, M.; Păun, Gh.; Prez-Jimenez, M.J.; Rozenberg, G. (2013); Frontiers of membrane computing: Open problems and research topics, *Int J Found Comput Sci*, ISSN 129-0541, 24(5):547-623.

[8] Glover, F.; Taillard, E.; Werra de, D. (1993); A users guide to tabu search, *Ann Oper Res*, ISSN 0254-5330, 41(1):3-28.

[9] Gottlieb, J.; Marchiori, E.; Rossi, C. (2002); Evolutionary algorithms for the satisfiability problem, *Evolut Comput*, ISSN 1063-6560, 10(1):35-50.

[10] Han, K.H.; Kim, J.H.(2002); Quantum-inspired evolutionary algorithm for a class of combinatorial optimization, *IEEE T Evolut Comput*, ISSN 1089-778X, 6(6):580-593.

[11] Huang, L.; Suh, I.H.; Abraham, A.(2011), Dynamic multi-objective optimization based on membrane computing for control of time-varying unstable plants, *Inform Sciences*, ISSN 0020-0255, 181(11): 2370-2391.

[12] Hwang, G.J.; Yin, P.Y.; Yeh, S.H.(2006); A tabu search approach to generating test sheets for multiple assessment criteria, *IEEE T Educ*, ISSN 0018-9359, 49(1):88-97.

[13] Leporati, A.; Pagani, D. (2006); A membrane algorithm for the min storage problem, *Lect Notes Comput Sci*, ISSN 0302-9743, 4361:443-462.

[14] Moore, M.; Narayanan, A. (1995); *Quantum-inspired computing. Tech. rep.*, Department of Computer Science, University Exeter, Exeter, U.K.

[15] Narayanan, A.; Moore, M. (1996); Quantum-inspired genetic algorithms, *Proc of IEEE CEC*, 61-66.

[16] Nishida, T.Y.(1996); Membrane algorithm with brownian subalgorithm and genetic subalgorithm. *Int J Found Comput Sci*, ISSN 129-0541, 18(6):1353-1360.

[17] Pan, L.Q., Alhazov, A., Ishdorj, T.O. (2005); Further remarks on P systems with active membranes, separation, merging, and release rules. *Soft Comput*, ISSN 1432-7643, 9(9):686-690.

[18] Pan, L.Q.; Martín-Vide, C. (2005); Solving multidimensional 0-1 knapsack problem by P systems with input and active membranes, *J Parallel Distr Comput*, ISSN 0743-7315, 65(12):1578-1584.

[19] Păun, Gh. (2000); Computing with membranes, *J Comput Syst Sci*, ISSN 0022-0000, 61(1):108-143.

[20] Păun, Gh. (2001); P systems with active membranes: attacking NP-complete problems. *J Automata Lang Comb*, ISSN 1430-189X, 6(1):75-90.

[21] Păun, Gh. (2007); Tracing some open problems in membrane computing. *Rom J Inf Sci Tech*, ISSN 1453-8245, 10(4):303-314.

[22] Păun, Gh.; Rozenberg, G.; Salomaa, A., eds. (2010); *The Oxford Handbook of Membrane Computing*. Oxford University Press.

[23] Whitley, D. (2001); An overview of evolutionary algorithms: practical issues and common pitfalls. *Inform Software Tech*, ISSN 0950-5849, 43(14):817-831.

[24] Xiao, J.H.; Zhang, X.Y.; Xu, J.(2012); A membrane evolutionary algorithm for DNA sequence design in DNA computing. *Chinese Sci Bull*, ISSN 1001-6538, 57(6):698-706.

[25] Xiao, J.H.; Jiang, Y.; He, J.J.; Cheng, Z.(2013); A dynamic membrane evolutionary algorithm for solving DNA sequences design with minimum free energy. *MATCH-Commun Math Ch*, ISSN 0340-6253, 70(3):971-986.

[26] Yang, S.; Wang, N. (2012): A novel P systems based optimization algorithm for parameter estimation of proton exchange membrane fuel cell model. *Int J Hydrogen Energ*, ISSN 0360-3199, 37(10): 8465-8476.

[27] Zhang, G.; Gheorghe, M.; Wu, C. (2008); A quantum-inspired evolutionary algorithm based on P systems for knapsack problem, *Fund Inform*, ISSN 0169-2968, 87(1):93-116.

[28] Zhang, G. (2011); Quantum-inspired evolutionary algorithms: a survey and empirical study. *J Heuristics*, ISSN 1381-1231, 17(3): 303-351.

[29] Zhang, G.; Cheng, J.; Gheorghe, M. (2011); A membrane-inspired approximate algorithm for traveling salesman problems, *Rom J Inf Sci Tech*, ISSN 1453-8245, 14(1):3-19.

[30] Zhang, G.; Cheng, J.; Gheorghe, M.; Meng, Q. (2013); A hybrid approach based on differential evolution and tissue membrane systems for solving constrained manufacturing parameter optimization problems, *Appl Soft Comput*, ISSN 1568-4946, 13(3):1528-1542.

[31] Zhang, G.X.; Cheng, J.X; Gheorghe, M. (2014); Dynamic behavior analysis of membrane-inspired evolutionary algorithms, *Int J Comput Commun Control*, ISSN 1841-9836, 9(2):227-242.

[32] Zhang, G.; Liu, C.; Rong, H. (2010); Analyzing radar emitter signals with membrane algorithms. *Math Comput Model*, ISSN 0895-7177, 52(11-12):1997-2010.

[33] Zhang, G.; Zhou, F.; Huang, X.; Cheng, J.; Gheorghe, M.; Ipate, F.; Lefticaru, R. (2012); A novel membrane algorithm based on particle swarm optimization for solving broadcasting problems, *J Univers Comput Sci*, ISSN 0948-695x, 18(13):1821-1841.

[34] Zhang, X.; Zeng, X.; Luo, B.; Zhang, Z.(2012); A uniform solution to the independent set problem through tissue P systems with cell separation, *Front Comput Sci*, ISSN 2095-2228, 6(4):477-488.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(2):280-290, April, 2015.

Human-Manipulator Interface Using Hybrid Sensors via CMAC for Dual Robots

P. Zhang, G. Du, B. Liang, X. Wang

Ping Zhang, Guanglong Du*
South China University of Technology, China
pzhang@scut.edu.cn, medgl@scut.edu.cn
*Corresponding author: medgl@scut.edu.cn

Bin Liang, Xueqian Wang
Graduate School at Shenzhen, Tsinghua University, China
bliang@tsinghua.edu.cn, wang.xq@sz.tsinghua.edu.cn

Abstract: This paper presents a novel method that allows a human operator to communicate his motion to dual robot manipulators by performing his double hand-arms movements, which would naturally carry out an object manipulation task. The proposed method uses hybrid sensors to obtain the position and orientation of the human hands. Although the position and the orientation of the human hands can be obtained from the sensors, the measurement errors increase over time due to the noise of the devices and the tracking error. A cerebellar model articulation controller (CMAC) is used to estimate the position and orientation of the human hands. Due to the limitations of the perceptive and the motor, human operator can not accomplish the high precise manipulation without any assistants. An adaptive multi-space transformation (AMT) is employed to assist the operator to improve the accuracy and reliability in determining the posture of the manipulator. With making full use of the human hand-arms motion, the operator would feel kind of immersive. Using this human-robot interface, the object manipulation task done in collaboration by dual robots could be carried out flexibly through preferring the double hand-arms motion by one operator.
Keywords: Cerebellar Model Articulation Controller (CMAC), robot teleoperation, human-robot interface.

1 Introduction

Human intelligence is required to make a decision and control the robot especially when it is in unstructured dynamic environments. Thus, robot teleoperation is necessary especially when the robot is in highly unstructured environments, where objects are unfamiliar or changing shape. There are some human-robot interfaces which have been commonly used [1], such as dials and robot replicas. However, for completing a teleoperation task, these contacting mechanical devices always require unnatural hand and arm motion.

There is another way to communicate complex motions to a remote robot and it is more natural that comparing with using contacting mechanical devices. This method is to track the operator hand-arm motion which is used to complete the required task using inertial sensors, contacting electromagnetic tracking sensors, gloves instrumented with angle sensors, and exoskeletal systems [2]. However, these contacting devices may hinder natural human-limb motion. Vision-based techniques are non-contacting and they are less hindering the hand-arm motion. Vision-based methods often use physical markers which are placed on the anatomical body part [3]. There are a lot of applications [4] based on marker-based tracking of the human motion. However, because body markers may hinder the motion for some highly dexterous tasks and operators may be occluded, the marker-based tracking is not always practical. Thus, a markerless approach seems better for many applications.

Compared with image-based tracking which uses markers, markerless is not only less invasive, but also can eliminate problems of marker occlusion and identification [5]. Thus, for remote robot teleoperation, markerless tracking may be a better approach. However, the existing markerless human-limb tracking techniques have many limitations so that they may be difficult to use in robot teleoperation applications. Many existing markerless-tracking techniques capture images and then compute the motion later [6, 7]. The robot manipulator could be controlled with the continuous robot motion by the markerless tracking. To allow the human operator to perform hand-arm motions for a task in a natural way without any interruption, the position and orientation of the hand and arm should be provided immediately. Many techniques can only provide 2D image information of the human motion [8], but the tracking methods cannot be extended for accurate 3D joint-position data. An end-effector of a remote robot would require the 3D position and orientation information of the operators limb-joint centers with respect to a fixed reference system. The problem how to identify human body parts in different orientations has always been a main challenge [9].

For robot teleoperation, some limited research towards markerless human-tracking has been done. Many techniques used a human-robot interface based on hand-gesture recognition to control the robot motion. Ionescu et al [10] developed markerless hand-gesture recognition methods which can be used for mobile robot control where only a few different commands are sufficient such as go, stop, left, right. However, for object manipulation in 3D space, it is not possible to achieve natural control and flexible robot motion by using gestures only. If a human operator wants to use gestures, he/she needs to think of those limited separate commands that the human-robot interface can understand like move up, down, forward. A better way of human-robot interaction would permit the operator to focus on the complex global task as a human operator naturally does when grasping and manipulating objects in 3D space instead of thinking about which type of hand motions are required. To achieve this goal, a method that allows the operator to complete the task using the hand-arm motions naturally provides the robot with information of the hand-arm motion in real time, that is the hand and arm anatomical position and orientation [11, 12]. However, method [11] is hard to obtain the accurate orientation of the human hand since the occlusion easily happens. To achieve the initialization, the human operator must assume a simple posture with an unclothed arm in front of a dark background, and the hand should be higher than the shoulder. Therefore, through the method [12], it is not possible to get precise results under a complex background. In addition, the human operator is hard to work in the chill weather with unclothed arm. And also it is limited because of lighting effect, which is difficult to work where the environment is too bright or too dark.

As the manipulation taskbeing more complex, multiple robot cooperation will be a trend. Vision-based methods mentioned above are hard to use in the multi-robot interface, as they can solve the occlusion problem. In addition, the contacting interfaces used in teleoperation for the multiple robot manipulators often require multiple operators [13].

This article presents a method of dual robot-manipulators interface using hybrid sensors to track the hand of the human operator (Fig. 1). Hand tracking based hybrid sensors is used to acquire 3D anatomical position and orientation and sent the data to the robot manipulator by a human-robot interface, which enables the robot end-effector to copy the operator hand motion in real time.

Fig. 2 shows the structure of our system. A position sensor provides the location of the human hand and an IMU provides the orientation of the human hand. We use Camshift to track the human hand. An IMU consisting of gyroscopes and magnetometer can measure the orientation and the acceleration of the hand. A CMAC estimation algorithm is used to estimate the positions and the accelerations of the hand. Another CMAC is applied to estimate the orientation and the angular velocities of the hand. In order to eliminate the influence of the noise and the

tracking failure, speed control method is used instead of the absolute location control method. The velocity value is transmitted to the manipulator and the manipulator moves at the received rate. The velocities of the position and the orientation are processed via AMT to improve the accuracy of manipulation. The natural way to communicate with the robots allows the operator to focus on the task instead of thinking the limited separate commands that the human-robot interface can understand like gesture- based approaches. There may be marker occlusion and identification when using vision-based approaches. In addition, this way let the operator feel kind of immersive in the multi-robot environment, which feels that his hands are in the robot site.

In Section 2, the CMAC method is presented. The human hand tracking system is then detailed in Section 3. Section 4 describes AMT method. Experiments and results are presented in Section 5. Discussions of this paper are detailed in Section 6, followed by concluding remarks in Section 7.

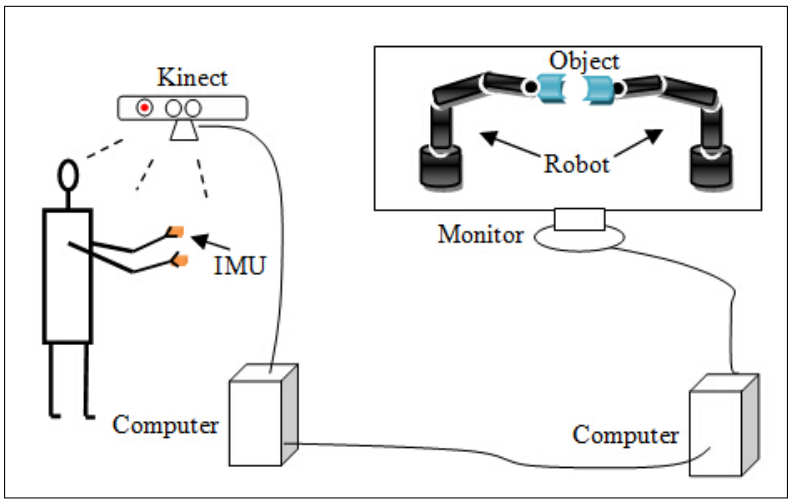


Figure 1: Non-invasive robot teleoperation system based on the Kinect

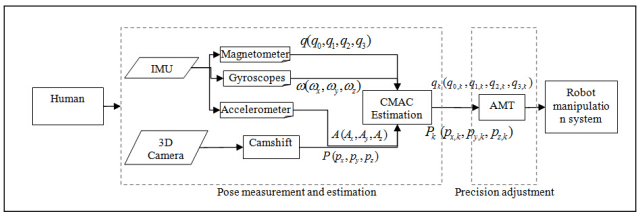


Figure 2: The structure of the system

2 Cerebellar Model Articulation Controller (CMAC)

Since the signals of the position and the orientation of human hand are time-varying and they are ill-defined when occlusion is encountered, an adaptive filter is required. In this paper, we use CMAC to process the pose signal (position and orientation). The convergence of CMACoutput error can be guaranteed because the stable conditions of the adaptive learning-rates are derived based on the Lyapunov function [14].

There are five components in CMAC: input space, association memory space, receptive-field space, weight memory space, and output space. The flow of CMAC is as follows:

(1) Input space C: The input vectoris $c = [c_1, c_2, ..., c_n]^T \in R$ according to a given control space.$c_i$is the input state variable and n is the input dimension. Each input state variable c_i must be quantized into discrete regions (element).
(2) Association memory space A: Several elements can be accumulated as a block B and each block performs a receptive- field basis function. The number of blocks n_B is greater than two and the number of components in the association memory space is $n_A = n \times n_B$. For given i^{th} input c_{ri} with the mean m_{ij} and variance σ_{ij}, we adopt Gaussian function as the receptive-field basis function ϕ_{ij} for j_{th} block, which can be defined as:

$$\phi_{ij} = exp[\frac{-(c_i - m_{ij})^2}{\sigma_{ij}^2}] \quad j = 1, 2, ..., n_B \tag{1}$$

where $\phi_{ij}(k-1)$ denotes the value of $\phi_{ij}(k)$ trough a time delay.
(3) Receptive-field space F: Areas formed by blocks are defined as receptive-fields. The number of receptive-fields is n_R, which equals n_B. The j_{th} multi-dimensional receptive-field function is represented as

$$b_j(c, m_j, \sigma_j) = \prod_{i=1}^{n} \phi_{ij} = exp[-(\sum_{i=1}^{n} \frac{(c_i - m_{ij})^2}{\sigma_{ij}^2})] \tag{2}$$

where $m_j = [m_{1j}, m_{2j}, ..., m_{nj}]^T \in R^n$ and $\sigma_j = [\sigma_{1j}, \sigma_{2j}, ..., \sigma_{nj}]^T \in R^n$. The multi-dimensional receptive-field functions can be expressed in a vector form as

$$\Gamma(c, m, \sigma) = [b_{1j}, b_{2j}, ..., b_{nj}]^T \tag{3}$$

(4) Weight memory space W: Each location of F to a particular adjustable value in the weight memory space can be defined as

$$\mathbf{w} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{nR}]^T = \begin{bmatrix} w_{11} & \cdots & w_{1p} \\ \vdots & \ddots & \vdots \\ w_{nR1} & \cdots & w_{nRp} \end{bmatrix} \tag{4}$$

where w_j denotes the connecting weight value of the output associated with the j_{th} multi-dimensional receptive-field.p is the number of the output value.
(5) Output space Y: The output of RCMAC is the algebraic sum of the activated weights in the weight memory,

$$\mathbf{y} = [y_1, y_2, ..., y_{n_p}] = w^T \Gamma(c, m, \sigma) \tag{5}$$

3 Hand tracking

3.1 Orientation measurement

The FQA, which is based on Earth gravity and magnetic field measurements, is used to estimate the orientation of a rigid body [15]. But this algorithm is applied for static or slow-moving rigid body only. To make it applicable for relatively large linear accelerations, CMAC fusion algorithm is used together with angular rate information to estimate orientation of dynamic body (either slow-moving or fast-moving) in the next section. For the estimation of the orientation, the quaternion states and the angular velocities (measured by IMU) are used as input elements,

$$\mathbf{c}_{ori} = [q_0, q_1, q_2, q_3, w_x, w_y, w_z] \tag{6}$$

where (w_x, w_y, w_z) is the vector of angular velocities. The input dimension n_{ori} is seven.

The output vector is

$$y_{ori} = [q_0, q_1, q_2, q_3] \tag{7}$$

3.2 Hand position tracking

Since the human operator holds the IMU to measure the orientation of his hand, the position of the IMU is equal to the position of the human hand. Moreover, the color of the IMU is special, so it is easier to be identified than the human hand. In the IMU position tracking system, we use a 3D camera to obtain two 2D images of the IMU: color image and depth image. The IMU positions are tracked through Camshift algorithm, which is more and more noticed by means of its favorable performance in reality and robust.
For the estimation of the position, we use the translation p_x, p_y, p_z (measured by Kinect) and the acceleration a_x, a_y, a_z(measured by IMU) as input elements,

$$\mathbf{c}_{pos} = [p_x, p_y, p_z, a_x, a_y, a_z] \tag{8}$$

The input dimension n_pos is seven. The output vector is

$$y_{pos} = [p_x, p_y, p_z] \tag{9}$$

4 Adaptive Multispace Transformation (AMT)

Robot manipulator has inherent perceptive limitations (such as perception of distance) and motor limitations (such as physiological tremor), which prevents the operator from operating precisely and smoothly enough for certain tasks [16]. For improving the visual and motor performance of teleoperation interface, we applied the modified version of Adaptive Multi-space Transformation (AMT) [16]. In the proposed method, an interface which introduces two scaling processes link the human operator working space to the robot working space (Fig. 3). The first change scales the movement produced by the human operator. Another change of scale is applied between the virtual unit vector K of the central axis of the robot EE and the robot movements. Such changes of scale modify the robot speed and, thus, improve performance.

The scaling vector S is used to relate the actions of the human hand in master space MS and the movement of the virtual unit vector K in the visual space VS. Another scaling variable u is used to relate the VS space to the robot working space WS. S and u are a function of the distance r between robot EE and the target. When $S < 1$, it decelerates the movement of K. Instead, it accelerates the movement of K while $S > 1$.

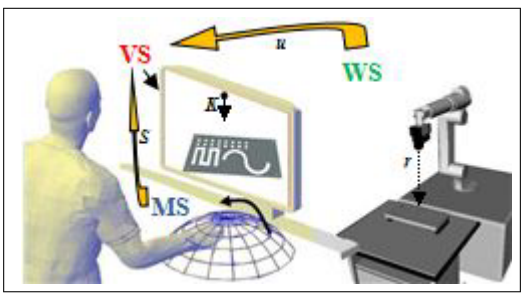


Figure 3: Representation of the human-interface-robot spaces

As the movement of the virtual position in VS is affected by vector S, let$S = [S_{pos}, S_{ori}]$, where S_{pos} is the scaling vector of the position and S_{ori} is the scaling variable of the orientation.

The Euler angular velocity in MS is $\dot{o_M}$ and $\dot{o_y}$ is the angular velocity in VS. Then

$$\dot{o_V} = \begin{cases} S_{ori}\dot{O_M} & \dot{O_M} \leq \delta_{ori} \\ 0 & \dot{O_M} > \delta_{ori} \end{cases} \tag{10}$$

where δ_{ori} is the threshold value. Assume that $\dot{P_M}$ is the velocity vector of hand movement in MS and $\dot{P_V}$ is the speed of the vector K in VS, the mapping vector of $\dot{P_M}$ in the VS is $\dot{P_M}$. Let $S_{pos} = [s_K s_{K_\perp}]$ and $K_\perp$ is a vector which is perpendicular to K. In addition,$K_\perp$,K and $\dot{P_M}$ are coplanar. So we have

$$\begin{aligned} p_K &= \begin{cases} S_K \bullet (|\dot{P_M\prime}| \bullet cos\theta) \bullet K = s_K \bullet K \bullet P_M\prime \bullet K & S_K \leq \delta_K \\ 0 & S_K > \delta_K \end{cases} \\ P_{K_\perp} &= \begin{cases} S_{K_\perp} \bullet (|\dot{P_M\prime}| \bullet sin\theta) \bullet K_\perp = s_{K_\perp} \bullet K_\perp \bullet P_M\prime \bullet K_\perp & S_K \leq \delta_K \\ 0 & S_{K_\perp} > \delta_{K_\perp} \end{cases} \end{aligned} \tag{11}$$

where δ_K and $\delta_{K_\perp}$ are the threshold value. Then the speed $\dot{P_V}$ of the vector K is given as

$$\dot{P_V} = p_K + p_{K_\perp} \tag{12}$$

When $s_K = s_{K_\perp}$, the speed $\dot{p_V}$ results in $\dot{P_V} = s_K P_M\prime$. While $s_K < s_{K_\perp}$, it requires greater precision in the direction of the central axis of EE. Instead, the direction which is perpendicular to the central axis requires greater precision with $s_K > s_{K_\perp}$. In this paper, we use $s_K > s_{K_\perp}$ so that the operator can easy to center the axis of the hole, as well as insert the peg into the hole quickly. Since S_{ori} and are functions of distance r, the function $S_{ori}(r)$ is defined as

$$\begin{cases} S_{ori}(r) = log(r) + C_1 \\ S_{K_\perp}(r) = log(r) + C_2 \\ S_K(r) = \sqrt{r} + C_3 \end{cases} \tag{13}$$

whereC_1,C_2 and C_3 are constants.

5 Experiments

5.1 Environment of experiment

A comparison experiment was designed to verify the proposed method. In the experiment, a series of screwing bolt tests were carried out to compare our method with methods [11, 12] in efficiency of the manipulation. The three methods controlled the position and orientation of robot EE directly to screw the bolt into the nut. Joint angles can be achieved by the solution of robot inverse kinematics. Given a posture of robot EE, the angle of each joint can be achieved by the robot inverse kinematics [17]. Fig. 4 shows the environment of the experiment.
Two GOOGOL GRB3016 robots were used in our experiments. Tab. 1 lists the nominal robot link parameters of the robot. Two robot manipulators were placed at the distance of 205.3 cm. The operator stood in the front of the Kinect and held an IMU. The leap motion sampled at 30 HZ. The diameter of the nut was 75.5 mm and the diameter of the bolt was 75 mm (Fig. 5). The length of the nut was 91 mm and that of bolt was 83 mm. In each task, the operator needed to control the robots and screwed the bolt into the nut.

Figure 4: Environment of experiment

Table 1: The DH parameters for the GOOGOL GRB3016 robot(a: link length.: link twist. d: link offset.: joint angle.)

JointDH	a(mm)	α (rad)	d (mm)	θ (rad)
1	150	$-\pi/2$	250	0
2	570	-π	0	$-\pi/2$
3	150	$\pi/2$	0	0
4	0	$-\pi/2$	650	0
5	0	$-\pi/2$	0	$-\pi/2$
6	0	0	-200	0

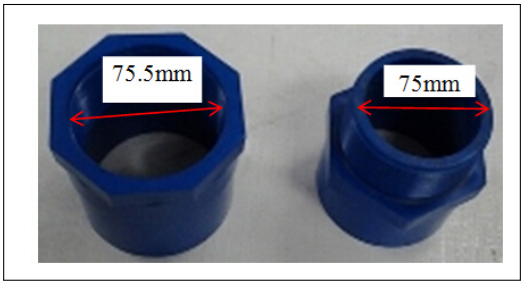


Figure 5: The screw nut and the screw bolt

5.2 Results of experiment

In the proposed algorithm, CMAC was used to estimate the orientation and the position of the human hand. Fig. 6 shows the 3-D path of two robot end-effectors during the screwing bolt experiment. The whole time of each test was about 240s. Fig. 7 shows the results of tracking the robot EE in the period of adjusting the position and orientation (about 35 s). The period of screwing blot was ignored since the position and the orientation did not change greatly. Fig. 7 (a, b, c) shows the position in X, Y, Z and Fig. 7 (d, e, f) shows the orientation in Yaw, Pitch and Roll. The dotted lines are the measured data (ML) received from the left hand. The dash dot lines are the measured data (MR) received from the right hand. The star lines are the estimated data of the left robot EE (EL). The plus lines are the estimated data of the right robot EE (ER). The estimation algorithms could reduce the noise and improve the operating accuracy. Since the gap between the blot and the nut was 0.5 mm, whether the operator could screw the blot into the nut was used to evaluate the accuracy of the three methods. Moreover, we compared the efficiency between our method and methods [11, 12], including the operation time and times of faults in each test.

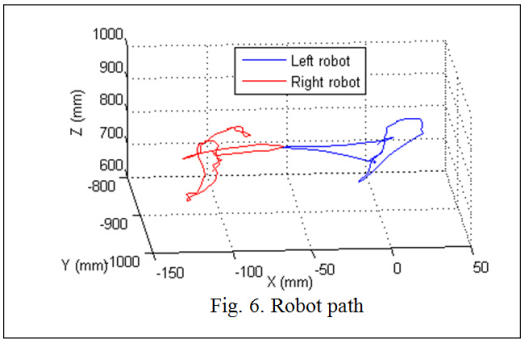


Figure 6: Robot path

The three methods were compared in terms of operation time and times of faults. The operation time includes closing the two robot end-effectors, adjusting the two robot end-effectors, screwing the bolt and separating the two robot end-effectors. The times of faults are the number of the operator tries to screw the bolt into the nut in each test. Tab. 2 shows the results of the operation time and times of faults for 5 tests, respectively. For test results of our method, the operation time for 5 tests ranges from 161 s to 169 s, with the mean time of 165.0 s. For methods [11, 12], the mean time is 193.4 s and 213.4 s. The mean times of faults are 3.6 and 5.6. Compared with methods [11, 12], the mean time of our method drops about 28.4 s and 48.4 s. Moreover, the times of faults of our method drops about 2.19 and 4.19.

Table 2: The DH parameters for the GOOGOL GRB3016 robot(a: link length.: link twist. d: link offset.: joint angle.)

		1	2	3	4	5	Mean
Our method	Time / s	165	161	166	169	164	165.0
	Times	2	2	1	1	1	1.41
Method [11]	Time / s	191	188	194	198	196	193.4
	Times	4	3	3	5	3	3.6
Method [12]	Time / s	211	208	214	218	216	213.4
	Times	6	5	5	7	5	5.6

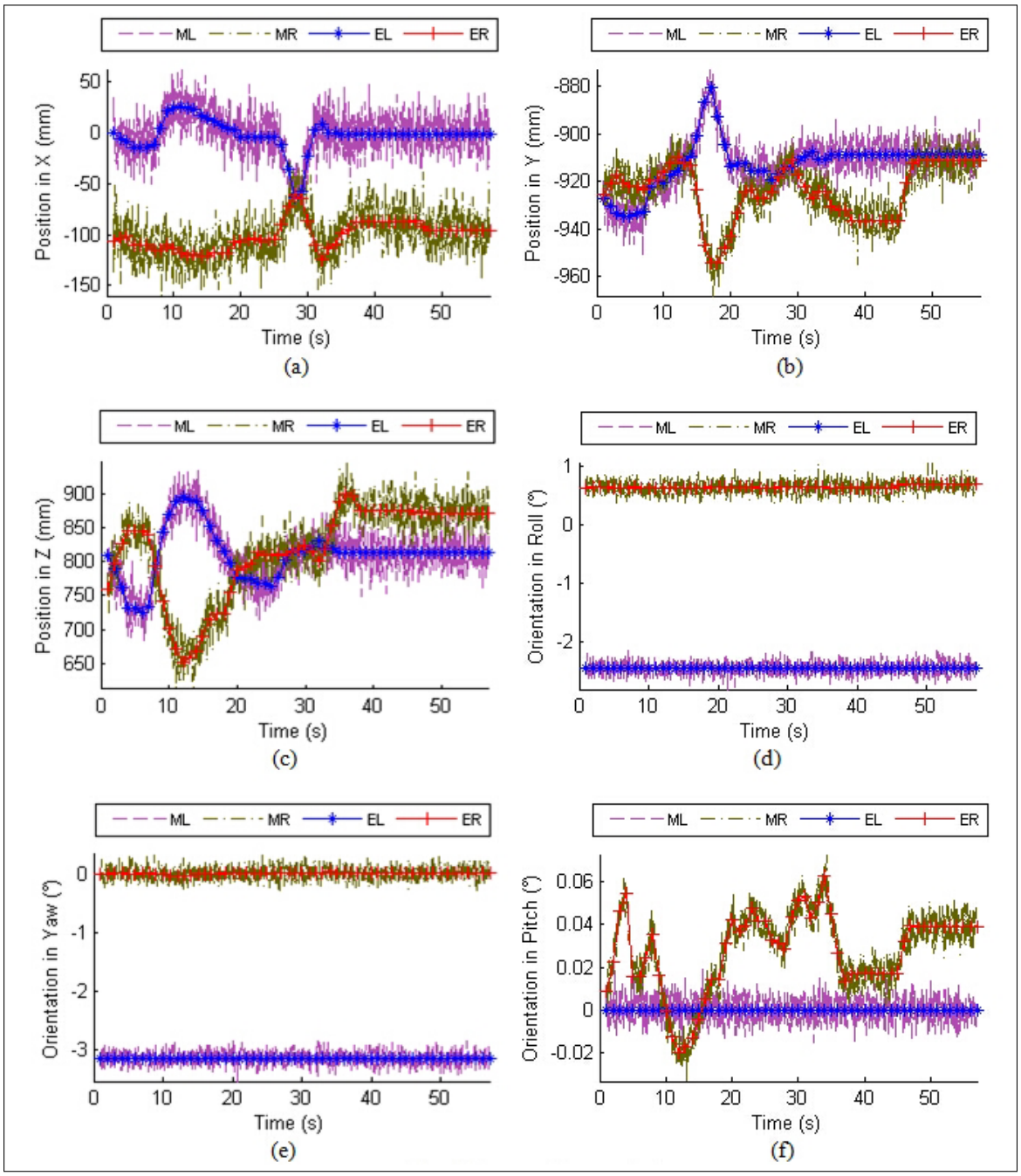


Figure 7: Peg tracking results

6 Discussions

This paper proposes a robot control method based on hybrid sensors. The motions of robot manipulators are controlled by imitating the motions of operators. For the robot teleoperation in the remote unstructured environment, a condition is assumed that all the remote robot site components can be installed on a mobile platform and enters those unstructured environment. And the remote unstructured environments involve robotic arm, robot controller, cameras on end-effectors, and some other cameras. The method shown here was testified by screwing a bolt into a nut. This system includes the operator into the decision control loop and that is a significant advantage. Firstly, it allows a robot to finish some tasks like moving without any prior knowledge like start location and even destination location. Secondly, there are some similar tasks requiring decision making. When picking objects or cleaning some objects which may contain some dangerous items, decision making is always required. Also it is expected that this system can be used to achieve the complex postures when the joints of the robot are limited. Compared with contacting electromagnetic devices, like hand joystick and data gloves, our method would not hinder most natural human-limb motion and allows the operator concentrate on his own task instead of decomposing commands into some simple operations. Compared with the non-contacting markerless method [11, 12], our method was proved more accurate and efficient. Moreover, as the manipulation task to be more complex, multiple robot cooperation will become a trend. However, the contacting interfaces used in teleoperation for the multiple robot manipulators often require multiple operators [18]. In this paper, it only needs one operator to control two robot-manipulators to complete complex manipulation.

In the future, gesture control will be combined with voice control. Simple commands like start, end, and pause can be completed by voice. Then the system will work more flexibly and advantageously.

7 Conclusions

This paper presents a human-robot interface system which uses a tracking technology based on hybrid sensors. To eliminate the effect of occlusion and failure of motion sensing, CMAC is developed to estimate the orientation and position information of the human hand. Since robot manipulator have inherent perceptive limitations and motor limitations, the high-precision tasks cannot be performed directly by the operator. Thus, AMT system is employed to assist the operator to complete the high-precision tasks.

In the experiment raised in this paper, screwing tasks have been carried out. We compared the efficiency between our method and method [11, 12]. Experimental results show that our method has better accuracy and efficiency. Further investigation of finger positioning will be considered to improve the performance of the human-robot interaction method.

Acknowledgments

Project funded by National Natural Science Foundation of China (Grant No:61403145) China Postdoctoral Science Foundation (NO:2014M550436) and the Fundamental Research Funds for the Central Universities (NO:2014ZM0039).

Bibliography

[1] T. Ando, R. Tsukahara, M. Seki (2012); A Hapic Interface Force Blinker 2 for Navigation of the Visually Impaired, *IEEE Trans. on Industrial Electronics*, 59(11): 4112-4119.

[2] K. Kiguchi, S. Kariya, K. Watanabe (2003); An exoskeletal robot for human elbow motion support-sensor fusion, adaptation, and control, *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, 31(3): 353-361.

[3] GL. Du, P. Zhang, LY. Yang, YB. Su (2010); Robot teleoperation using a vision-based manipulation method, *Audio Language and Image Processing*, (ICALIP 2010 Int. Conf., Shanghai, 945-949.

[4] A. Peer, H. Pongrac, M. Buss (2010); Influence of Varied Human Movement Control on Task Performance and Feeling of Telepresence, *Presence-Teleoperators and Virtual Environments*, 19(5):463-481.

[5] V. Siddharth (2004); *Vision-based markerless 3D human-arm tracking*, M.A.Sc. Thesis, Dept. of Mech. Eng., University of Ottawa, Canada.

[6] YH Ma, ZH Mao, W. Jia, C. Li, J. Yang, Magnetic Hand Tracking for Human-Computer Interface, *IEEE Trans. on Magnetics*, 47(5): 970-973, 2011.

[7] K.C.C. Peng, W. Singhose, D.H.Frakes (2012); Hand-Motion Crane Control Using Radio-Frequency Real-Time Location Systems, *IEEE Trans. on Mechatronics*, 17(3): 464-471.

[8] M. Khezri, M. Jahed (2011); A Neuro Fuzzy Inference System for sEMG-Based Identification of Hand Motion Commands, *IEEE Trans. on Industrial Electronics*, 58(5): 1952-1960.

[9] A.R.Varkonyi-Koczy, B.Tusor (2011); HumanComputer Interaction for Smart Environment Applications Using Fuzzy Hand Posture and Gesture Models, *IEEE Transactions on Instrumentation and Measurement*, 60(5):1505-1514.

[10] B. Ionescu, D. Coquin, P. Lambert, V. Buzuloiu (2005); Dynamichand gesture recognition using the skeleton of the hand, *Journal on Applied Signal Processing*, 2101-2109.

[11] GL Du, P Zhang (2014); Markerless Human-Robot Interface for Dual Robot Manipulators Using Kinect Sensor, *Robotics and Computer Integrated Manufacturing*, 30(2): 150-159.

[12] J. Kofman, S. Verma, X. Wu (2007); Robot-Manipulator Teleoperation by Markerless Vision-Based Hand-Arm Tracking, *Int. J. of Optomechatronics*, 1(3): 331-357.

[13] R. Marin, PJ. Sanz, R. Wirz (2005); A Multimodal Interface to Control a Robot Arm via the Web: A Case Study on Remote Programming, *IEEE Transactions on Industrial Electronics*, 52(6): 1506-1521.

[14] CM. Lin, LY. Chen, DS. Yeung (2010); Adaptive Filter Design Using Recurrent Cerebellar Model Articulation Controller, *IEEE Trans. on Neural Networks*, 19(7): 1149-1157.

[15] X. Yun, ER. Bachmann, RB. McGhee (2008): A Simplified Quaternion-Based Algorithm for Orientation Estimation From Earth Gravityand Magnetic Field Measurements, *IEEE Trans. on Instrumentation and Measurement*, 57(3): 638-650.

[16] LM. Munoz, A. Casals (2009); Improving the Human-Robot Interface Through Adaptive Multispace Transformation, *IEEE Trans. on Robotics*, 25(5): 1208-1213.

[17] G. Antonelli, S.Chiaverini, G. Fusco (2003); A new on-line algorithm for inverse kinematics of robot manipulators ensuring path tracking capability under joint limits, *IEEE Trans. on Robotics and Automation*, 19(1): 162-167.

[18] R. Marin, PJ. Sanz, R.Wirz (2005); A Multimodal Interface to Control a Robot Arm via the Web: A Case Study on Remote Programming, *IEEE Trans. on Industrial Electronics*, 52(6): 1506-1521.

Author index

Patriciu V.V., 222

Alic K., 154

Bădică C., 200
Baskaran R., 174
Benta D., 230
Budimac Z., 200

Chaoui A., 211
Chen Y., 248
Cheng J., 263
Costea C.R., 165
Cramariuc B., 230

Deivamani M., 174
Derpich I.S., 238
Dhavachelvan P., 174
Djouani K., 211
Du G., 280

Han B. R., 188

Ipate F., 263
Ivanović M., 200

Jerinić L., 200

Kahloul L., 211

Li J. B., 188
Li Y. J., 188
Liang B., 280

Mailat M., 230
Mitrović D., 200
Murugaiyan S.R., 174

Pătraşcu A., 222
Pertovt E., 154

Ravisankar V., 174
Rusu L., 230

Sepulveda J.M., 238
Silaghi H.M., 165
Silaghi M.A., 165
Svigelj A., 154

Victer Paul P., 174

Wang N., 248
Wang X., 263, 280

Zhang G., 263
Zhang P., 280
Zmaranda D., 165