Volume 26 Number 4 December 2002 ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics

The Slovene Society Informatika, Ljubljana, Slovenia

Informatica **26** (2002) Number 4, pp. 347–439

Informatica

An International Journal of Computing and Informatics

Archive of abstracts may be accessed at USA: http://, Europe: http://ai.ijs.si/informatica, Asia: http://www.comp.nus.edu.sg/ liuh/Informatica/index.html.

Subscription Information Informatica (ISSN 0350-5596) is published four times a year in Spring, Summer, Autumn, and Winter (4 issues per year) by the Slovene Society Informatika, Vožarski pot 12, 1000 Ljubljana, Slovenia.
The subscription rate for 2002 (Volume 26) is
- USD 80 for institutions,
- USD 40 for individuals, and
- USD 20 for students

Claims for missing issues will be honored free of charge within six months after the publication date of the issue.

LaTeX Tech. Support: Borut Žnidar, Kranj, Slovenia.
Lectorship: Fergus F. Smith, AMIDAS d.o.o., Cankarjevo nabrežje 11, Ljubljana, Slovenia.
Printed by Biro M, d.o.o., Žibertova 1, 1000 Ljubljana, Slovenia.

Orders for subscription may be placed by telephone or fax using any major credit card. Please call Mr. R. Murn, Jožef Stefan Institute: Tel (+386) 1 4773 900, Fax (+386) 1 219 385, or send checks or VISA or Eurocard card number or use the bank account number 900–27620–5159/4 Nova Ljubljanska Banka d.d. Slovenia (LB 50101-678-51841 for domestic subscribers only).

Informatica is published in cooperation with the following societies (and contact persons):
Robotics Society of Slovenia (Jadran Lenarčič)
Slovene Society for Pattern Recognition (Franjo Pernuš)
Slovenian Artificial Intelligence Society; Cognitive Science Society (Matjaž Gams)
Slovenian Society of Mathematicians, Physicists and Astronomers (Bojan Mohar)
Automatic Control Society of Slovenia (Borut Zupančič)
Slovenian Association of Technical and Natural Sciences / Engineering Academy of Slovenia (Igor Grabec)
ACM Slovenia (Dunja Mladenič)

Informatica is surveyed by: AI and Robotic Abstracts, AI References, ACM Computing Surveys, ACM Digital Library, Applied Science & Techn. Index, COMPENDEX*PLUS, Computer ASAP, Computer Literature Index, Cur. Cont. & Comp. & Math. Sear., Current Mathematical Publications, Cybernetica Newsletter, DBLP Computer Science Bibliography, Engineering Index, INSPEC, Linguistics and Language Behaviour Abstracts, Mathematical Reviews, MathSci, Sociological Abstracts, Uncover, Zentralblatt für Mathematik

The issuing of the Informatica journal is financially supported by the Ministry of Education, Science and Sport, Trg OF 13, 1000 Ljubljana, Slovenia.

Post tax payed at post 1102 Ljubljana. Slovenia taxe Percue.

INFORMATICA
AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS
INVITATION, COOPERATION

Submissions and Refereeing

Please submit three copies of the manuscript with good copies of the figures and photographs to one of the editors from the Editorial Board or to the Contact Person. At least two referees outside the author's country will examine it, and they are invited to make as many remarks as possible directly on the manuscript, from typing errors to global philosophical disagreements. The chosen editor will send the author copies with remarks. If the paper is accepted, the editor will also send copies to the Contact Person. The Executive Board will inform the author that the paper has been accepted, in which case it will be published within one year of receipt of e-mails with the text in Informatica LaTeX format and figures in .eps format. The original figures can also be sent on separate sheets. Style and examples of papers can be obtained by e-mail from the Contact Person or from FTP or WWW (see the last page of Informatica).

Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the Contact Person.

QUESTIONNAIRE

□ Send Informatica free of charge

□ Yes, we subscribe

Please, complete the order form and send it to Dr. Rudi Murn, Informatica, Institut Jožef Stefan, Jamova 39, 1111 Ljubljana, Slovenia.

Since 1977, Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automation and other related areas. In its 16th year (more than five years ago) it became truly international, although it still remains connected to Central Europe. The basic aim of Informatica is to impose intellectual values (science, engineering) in a distributed organisation.

Informatica is a journal primarily covering the European computer science and informatics community - scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board.

Informatica is free of charge for major scientific, educational and governmental institutions. Others should subscribe (see the last page of Informatica).

ORDER FORM – INFORMATICA

Name: ....

Title and Profession (optional): ....

....

Home Address and Telephone (optional): ....

....

Office Address and Telephone (optional): ....

....

E-mail Address (optional): ....

Signature and Date: ....

Informatica WWW:

http://ai.ijs.si/informatica/
http://orca.st.usm.edu/informatica/

Referees:

Witold Abramowicz, David Abramson, Adel Adi, Kenneth Aizawa, Suad Alagić, Mohamad Alam, Dia Ali, Alan Aliu, Richard Amoroso, John Anderson, Hans-Jurgen Appelrath, Iván Araujo, Vladimir Bajič, Michel Barbeau, Grzegorz Bartoszewicz, Catriel Beeri, Daniel Beech, Fevzi Belli, Simon Beloglavec, Sondes Bennasri, Francesco Bergadano, Istvan Berkeley, Azer Bestavros, Andraž Bežek, Balaji Bharadwaj, Ralph Bisland, Jacek Blazewicz, Laszlo Boeszoermenyi, Damjan Bojadžijev, Jeff Bone, Ivan Bratko, Pavel Brazdil, Bostjan Brumen, Jerzy Brzezinski, Marian Bubak, Davide Bugali, Troy Bull, Leslie Burkholder, Frada Burstein, Wojciech Buszkowski, Rajkumar Bvyya, Netiva Caftori, Particia Carando, Robert Cattral, Jason Ceddia, Ryszard Choras, Wojciech Cellary, Wojciech Chybowski, Andrzej Ciepielewski, Vic Ciesielski, Mel Ó Cinnéide, David Cliff, Maria Cobb, Jean-Pierre Corriveau, Travis Craig, Noel Craske, Matthew Crocker, Tadeusz Czachorski, Milan Češka, Honghua Dai, Bart de Decker, Deborah Dent, Andrej Dobnikar, Sait Dogru, Peter Dolog, Georg Dorfner, Ludoslaw Drelichowski, Matija Drobnič, Maciej Drozdowski, Marek Druzdzel, Marjan Družovec, Jozo Dujmović, Pavol Ďuriš, Amnon Eden, Johann Eder, Hesham El-Rewini, Darrell Ferguson, Warren Fergusson, David Flater, Pierre Flener, Wojciech Fliegner, Vladimir A. Fomichov, Terrence Forgarty, Hans Fraaije, Hugo de Garis, Eugeniusz Gatnar, Grant Gayed, James Geller, Michael Georgiopolus, Michael Gertz, Jan Goliński, Janusz Gorski, Georg Gottlob, David Green, Herbert Groiss, Jozsef Gyorkos, Marten Haglind, Abdelwahab Hamou-Lhadj, Inman Harvey, Jaak Henno, Marjan Hericko, Elke Hochmueller, Jack Hodges, Doug Howe, Rod Howell, Tomáš Hruška, Don Huch, Simone Fischer-Huebner, Alexey Ippa, Hannu Jaakkola, Sushil Jajodia, Ryszard Jakubowski, Piotr Jedrzejowicz, A. Milton Jenkins, Eric Johnson, Polina Jordanova, Djani Juričič, Marko Juvancic, Sabhash Kak, Li-Shan Kang, Ivan Kapustøk, Orlando Karam, Roland Kaschek, Jacek Kierzenka, Jan Kniat, Stavros Kokkotos, Fabio Kon, Kevin Korb, Gilad Koren, Andrej Krajnc, Henryk Krawczyk, Ben Kroese, Zbyszko Krolikowski, Benjamin Kuipers, Matjaž Kukar, Aarre Laakso, Les Labuschagne, Ivan Lah, Phil Laplante, Bud Lawson, Herbert Leitold, Ulrike Leopold-Wildburger, Timothy C. Lethbridge, Joseph Y-T. Leung, Barry Levine, Xuefeng Li, Alexander Linkevich, Raymond Lister, Doug Locke, Peter Lockeman, Matija Lokar, Jason Lowder, Kim Teng Lua, Ann Macintosh, Bernardo Magnini, Andrzej Małachowski, Peter Marcer, Andrzej Marciniak, Witold Marciszewski, Vladimir Marik, Jacek Martinek, Tomasz Maruszewski, Florian Matthes, Daniel Memmi, Timothy Menzies, Dieter Merkl, Zbigniew Michalewicz, Gautam Mitra, Roland Mittermeir, Madhav Moganti, Reinhard Moller, Tadeusz Morzy, Daniel Mossé, John Mueller, Jari Multisilta, Hari Narayanan, Jerzy Nawrocki, Rance Necaise, Elzbieta Niedzielska, Marian Niedq'zwiedziński, Jaroslav Nieplocha, Oscar Nierstrasz, Roumen Nikolov, Mark Nissen, Jerzy Nogieć, Stefano Nolfi, Franc Novak, Antoni Nowakowski, Adam Nowicki, Tadeusz Nowicki, Daniel Olejar, Hubert Österle, Wojciech Olejniczak, Jerzy Olszewski, Cherry Owen, Mieczyslaw Owoc, Tadeusz Pankowski, Jens Penberg, William C. Perkins, Warren Persons, Mitja Peruš, Stephen Pike, Niki Pissinou, Aleksander Pivk, Ullin Place, Gabika Polčicová, Gustav Pomberger, James Pomykalski, Dimithu Prasanna, Gary Preckshot, Dejan Rakovič, Cveta Razdevšek Pučko, Ke Qiu, Michael Quinn, Gerald Quirchmayer, Vojislav D. Radonjic, Luc de Raedt, Ewaryst Rafajlowicz, Sita Ramakrishnan, Kai Rannenberg, Wolf Rauch, Peter Rechenberg, Felix Redmill, James Edward Ries, David Robertson, Marko Robnik, Colette Rolland, Wilhelm Rossak, Ingrid Russel, A.S.M. Sajeev, Kimmo Salmenjoki, Pierangela Samarati, Bo Sanden, P. G. Sarang, Vivek Sarin, Iztok Savnik, Ichiro Satoh, Walter Schempp, Wolfgang Schreiner, Guenter Schmidt, Heinz Schmidt, Dennis Sewer, Zhongzhi Shi, Mária Smolárová, Carine Souveyet, William Spears, Hartmut Stadtler, Olivero Stock, Janusz Stokłosa, Przemysław Stpiczyński, Andrej Stritar, Maciej Stroinski, Leon Strous, Tomasz Szmuc, Zdzislaw Szyjewski, Jure Šilc, Metod Škarja, Jiři Šlechta, Chew Lim Tan, Zahir Tari, Jurij Tasič, Gheorge Tecuci, Piotr Teczynski, Stephanie Teufel, Ken Tindell, A Min Tjoa, Vladimir Tosic, Wieslaw Traczyk, Roman Trobec, Marek Tudruj, Andrej Ule, Amjad Umar, Andrzej Urbanski, Marko Uršič, Tadeusz Usowicz, Romana Vajde Horvat, Elisabeth Valentine, Kanonkluk Vanapipat, Alexander P. Vazhenin, Jan Verschuren, Zygmunt Vetulani, Olivier de Vel, Valentino Vranić, Jozef Vyskoc, Eugene Wallingford, Matthew Warren, John Weckert, Michael Weiss, Tatjana Welzer, Lee White, Gerhard Widmer, Stefan Wrobel, Stanislaw Wrycza, Janusz Zalewski, Damir Zazula, Yanchun Zhang, Ales Zivkovic, Zonling Zhou, Robert Zorc, Anton P. Železnikar

An Assessment of the Organization Virtuality with Three Different Reference Models

Cene Bavec
School of Management in Koper, Slovenia
Phone: +386 5 610 2000; fax: +386 5 610 2015
E-mail: cene.bavec@guest.arnes.si

Keywords: virtual organization, level of virtuality, modeling organization, colored Petri nets, virtual government

Received: May 17, 2002

The main objective of the research was to test a holistic view on virtual organizations with different perceptions of virtuality. Traditional and virtual organizations could be seen as two extremes of more general model of organization. To asses a transition from the traditional to the virtual organizations we have to grade organizational virtuality. In the paper we discuss three basically different reference models used to asses this transition. Two models are well known – the Mowshowitz's switching principle, and the Model of Business Networking (MBN) as a representative of models preferred by the ICT experts. They see the virtual organizations through implementation of the ICT, particularly the Internet. To express other characteristics of virtual organizations we also presented the model based on the Colored Petri nets and fuzzy logic that we originally used to study an organized anarchy. All three models were implemented to assess the case of the Custom Administration in Slovenia. An assessment confirms that organization of the Custom services clearly demonstrates an efficient utilization of the Internet and other features of virtual organizations.

1 Introduction

Many authors argue that the theory of virtual organizations leads to a generalization of the traditional organization theory. It is not yet a prevailing organizational concept (Klüber et al, 1999) but, Internet, networked and virtual organizations have already proved to be an efficient organizational paradigm that brought to the business world a higher level of flexibility, efficiency, resource utilization and better customer services.

An intuitive perception of virtual organizations is often inadequate and misleading so we are still searching for new managerial principles and practical tools for every day management that could replace still prevailing traditional organizational principles born in the industrial age. The theory of virtual organizations is presently very chaotic. We still haven't developed practically useful indicators to make an objective assessment of virtual organization and to distinguish them from the other forms of organization.

We have learned from practical experience that it is not realistic to classify organization in only two classes - virtual and traditional (Jansen at al, 1999). These should be seen as two extremes of a more general model of organizations. If we need to describe and to assess a transition from the traditional to the virtual organizations we have to grade their virtuality.

In this paper we discuss the indicators that could assess virtuality of specific business organization. In the case presented we studied a real business environment to underline practical issues of virtual organizations and to raise a general issue of their virtuality and efficiency. We used three very different reference models of virtual organizations. Two models are well known – the switching principle and metamanagement (Mowshowitz, 1999) and the Model of Business Networking (Klüber at al, 1999) that represents a class of models used mainly by the IT and Internet specialists.

To learn more about the structure and the internal nature of virtual organizations we also presented a formal definition of the organization, based on the Colored Petri nets and fuzzy sets logic. The definition was implemented in a computer model of non-hierarchical organizations and the organized anarchy (Bavec, 2001). The model was not initially designed to describe virtual organizations. Nevertheless, it predicted some features of non-traditional organizations as fuzziness of organizational rules and boundaries.

2 Reference Models Used

The main objective of the research was to test a holistic view on virtual organizations with different perceptions of virtuality:

- the Switching Principle – is mainly a managerial view with emphasis on organizational flexibility and manageability,
- the Model of Business Networking – it defines inter-organizational relations and predominantly an ICT view on organization with emphasis on modularity and business transparency,

- Model based on the Colored Petri nets formalism shows an internal view on "fuzzy" organizational structures and information flows.

We were well aware that models are not compatible and not even comparable but, we had intentionally selected such different perceptions of virtual organizations. The goal was to get some deeper understanding of potential indicators that could be used in their assessment. To overcome this methodological obstacle we used the models to separately assess seven features of virtual organizations proposed by Mertens et al. (1998):

- boundary crossing
- complementary core competencies
- sharing of knowledge
- geographical dispersion
- changing participants
- participant equality
- electronic communication

2.1 Switching Principle and Metamanagement

The first reference model we used was based on the switching principle and metamanagement introduced by Mowshowitz (1999). In the simplest way we could describe it as an ability of organization to dynamically select the best performer or executor (need-fulfillment) for a particular task (need). That means that an organization treats tasks separately from their potential performers. Switching would take place when replacing one performer would bring benefits that are greater than direct and indirect costs of replacement. Another concept introduced by Mowshowitz is metamanagement. It is basically management of virtually organized tasks and managerial implementation of the switching principle.

This principle may seem trivial at the first glance but it opens an entirely new view on the organization. We should notice that in the traditional organization theory and practice it was always a sign of serious miss-planning or "bad organization" when we had to change (switch) a performer in the production phase of the task when organization was already implemented.

Possibility of switching undoubtedly adds to organization and managerial flexibility, but the question that still remains is just how realistic it could be in every day business. Basic idea of virtuality is that switching could be done relatively fast. It would be difficult for the management to implement all traditional risk analysis, so the trust becomes an important decision and even an economic factor. We have to trust the new partner (Ishaya, Macaulay, 1999) and be reasonably confident that he will integrate into our operations and perform his role according to our expectations. If the mistrust is too high it could overwhelm other benefits.

From the Mowshowitz's model we could assume that the level of virtuality is correlated with the ability to implement the switching principle or metamangement. We could use it to assess changing participants and participant equality from the Mertens' list of virtual organization features.

2.2 ICT Oriented Models of Virtual Organization

Another, very different perception of virtual organization is seen in the Model of Business Networking (Klüber et al, 1999). It is a typical representative of models preferred by the ICT experts as they see virtual organizations through implementation of ICT, particularly the Internet. But, the model is more general and incorporates important features of virtual organizations that are highly relevant for the management.

The Model of Business Networking (MBN) has the following elements that were parts of our assessment of a real business environment:

- Customer processes (determine the design of a value chain),
- Integrators and Aggregators (third parties included into business relationships),
- Business Bus (logical space where complex services among business partners are flexibly and efficiently exchanged with the support of service providers),
- Business Ports (standardized interface to access the Business Bus).

Similar approaches are widely used, often under different names in design and implementation of information systems based on open networking like the Internet.

According to the MBN, integrators and aggregators are an essential element of networked and virtual organizations. They provide different business services: knowledge, coordination, process, information, and transaction services. They behave in the way to "soften" or even eliminate organizational boundaries between business partners. The Business Busses and the Business Ports describe inter-organizational relations and interfaces that define mainly an information structure of virtual organizations. But, more generally they describe complementary core competencies, sharing of knowledge, geographical dispersion and electronic communication.

3 Modeling Organizations with Colored Petri Nets

3.1 Rationale Behind the Model

In the year 1994 we developed a model of organization based on the extended Colored Petri nets and fuzzy logic to study organized anarchy and influence of information systems on the organization. Their superior semantic power makes possible a very rich representation of the organization and overcomes some limitations of classical representations. A general definition of the organization was based on abstract fuzzy sets with axiomatically assigned properties. The organization was defined as a set of rules that determined the chain of authority, description of working (organizational) places, and other organizational relations. It also determined conditions under which organizational processes could change their states. The quality of organization can be measured only through its impact on processes so it must be modeled together with them. The Petri nets proved to be an efficient way to combine organizational structure and the processes in the organization.

This methodology was used for modeling properties that reflect ambiguity or deviation from the traditional hierarchical organization. The study also exposes a paradigm that could be called an *informed anarchy paradigm* in analogy with the organized anarchy. The informed anarchy paradigm is based on empirically founded facts that unclear technology of allocation, dissemination, and also faulty understanding of information are prevailing properties of organizations.

An object-oriented model was developed that interlinks a formal organization with decision and information processes into an integral model. It was based on three classes of objects: organization, decision processes and information processes. An important feature of the model was its ability to model conditions on micro level, that means both on the level of working (organizational) place and individual process.

Further development confirmed that the same model represent also some relevant features of virtual organizations.

3.2 Model of Organization Based on Colored Petri Nets

Petri nets are well proven tools for systems modeling that can describe dynamic and static properties of the system. Similar situations, only much more complex, can be met in business organizations, so we experimented with the possibility to model them with Colored Petri nets (CPN). We published some results of the computer simulation (Bavec, 2001) - relation between the level of organization anarchy, load of problems, formal and informal information systems, and efficiency of decision-making.

We defined an organization *ORG* as a 12-tuple or an extended Colored Petri net (Bavec, 1995):

$ORG = (B,P,T,D,C,R,I,O,\delta,\eta,\rho,\sigma)$

$B = (b_1, b_2, \dots b_j)$	a finite set of colors
$P = (p_1,p_2, \dots p_n)$	a finite set of places
$T = (t_1,t_2, \dots t_m)$	a finite set of transitions
$D = (d_1, d_2, \dots d_n)$	a finite set of organizational places (working places, divisions, etc.)
$C = (c_1, c_2, \dots c_k)$	a finite set of concepts or objects
$R = (r_1, r_2, \dots r_m)$	a finite set of organizational relations $R = \{R \mid R \subseteq D \times D\}$

$B \cap P \cap T \cap D \cap C \cap R = 0$

$I\colon T \rightarrow P^{\infty}$	an input function that maps a set of transitions t_i into places p_i
$O\colon T \rightarrow P^{\infty}$	an output function that maps a set of transitions t_i into places $\vec{p}_i$
$\delta\colon D \rightarrow P$	a function that maps organizational places $d_i \in D$ into places $p_i \in P$
$\eta\colon C \rightarrow P$	a function that maps concepts or objects $c_i \in C$ into places $p_i \in P$
$\rho\colon R \rightarrow T$	a function that maps organizational relations $r_i \in R$ into transitions $t_i \in T$
$\sigma\colon Z \rightarrow P$	a function that maps multi-set of tokens $z(p_i) \in Z$ into places $p_i \in P$
$\lambda \in [0,1]$	a threshold, an additional condition for firing transitions.

We are aware that business and human organizations, particularly as complex ones as the virtual organizations, can't be highly formalized (structured). Consequently, there is a question how far can we go with formal definitions. But, the model confirms that we could model some features of virtual organizations with the CPN and its derivations (Deng et al. 1990).

In the model we implemented fuzzy logic, mainly through the threshold $\lambda \in [0,1]$ which additionally controls firing of tokens. We also proved that the introduction of concepts or objects (also Bastide, 1996) $c_i \in C$ assigned to organizational places $d_i \in D$ (they could be everything from working places to organizational units) provided us with the tool to model complex relations between organization as the set of rules, and processes that are running in accordance with the organizational rules.

With the controlled firing of tokens in the CPN and fuzzy logic we could describe and study features of virtual organizations like the switching principle, the ambiguity of organizational relations and particularly the boundary crossing.

4 The Case Study - Assessment of the Government Agency

4.1 Beyond the Business Partnership

Emerging experience and the theory of virtual organization is based on the present business practice with very few examples from the government administration. But, there is widely spread belief among researchers that governments and public administrations are one of the most promising grounds for an introduction of virtual organizations. The development of Internet and innovative customer and citizen oriented services in governments in developed countries more than justify this assumption. It is encouraging to notice a similar development also in the middle developed countries, such as Slovenia.

In the case presented we studied the collaboration of the Custom Administration in Slovenia (CA) with different private companies. The case reveals development steps from the traditional government agencies towards highly efficient and technologically advanced organizations that clearly articulate elements of the new organizational paradigm – the virtual organizations. The CA has developed a sophisticated and efficient Custom information system in which 98% of all custom declarations are lodged electronically up to the highest security standards.

The development of the CA information system and computer applications were outsourced to the private company ZZI from Ljubljana. The application was linked to the operation environment of users. Jointly with its partners, ZZI developed an interface to enable its software to be integrated into larger operation information systems based on the BAAN, SAP, NAVISION, etc. It is important to notice that the CA has been equally opened to other potential partners that could develop software and services to enable different users to link their systems to the Custom information system, enabling them to submit the customs declarations and other documents by the Internet.

Beside the software development ZZI and nearly 30 other providers also offer transmission services for the electronic data interchange within different environments and among different partners. The server solutions and programs used by clients facilitate the automatic data interchange within the different environments and applications. Such service for example is the transmission of the messages from the Internet environment of partners to the X.400 environment of the Custom Administration.

4.2 Implementation of the Switching Principle and Metamanagement

At the beginning of the research we were concerned with the notorious rigidity of the government organization. It seemed quite unrealistic to notice any sign of the switching principle in the government agencies. But, what we really found was a clear presence of elements of metamanagement and implementation of the switching principle.

Services such as the transmission of the messages from the Internet environment of partners to the X.400 environment of the Custom Administration and other forms of electronic data interchange services performed by private companies are without any doubt "switchable". The official policy of the CA is in favor of tighter inclusion of trustworthy participants into the Custom Information System. This just confirms this development. Transferring elements of the CA authority to partners express another important feature of virtual organizations – the trust. From the government point of view it is a major breakthrough to realize that it is more convenient and even cheaper to trust partners and to control them more "softly" in indirect and off-line mode.

Looking into the project and official documents of the CA reveals that the switching principles as well as the role of trust are introduced entirely intuitively, without any reference to virtual organizations. It shows that evolution of virtual organization could be entirely spontaneous and a natural development in the competitive and technologically advanced environments.

4.3 Implementation of the MBN

According to the Model of Business Networking (MBN) integrators and aggregators are essential elements of networked and virtual organizations. They are the third parties included into business relationship. In the case of the Custom information system they are not government agencies, nor the users of the system. Electronic data interchange services performed by the ZZI and other companies are "infomediary" (Österle, 1999). The companies around the CA are playing the role of integrators and aggregators with standardized procedures and with standardized interfaces and can be interchanged and replaced. It gives the CA a very high flexibility, accompanying with noticeable cost reduction and better customer services.

Again, the CA implemented the most visible feature of virtual organization – a very high flexibility.

The overall architecture of the CA information system is modular with surprisingly similar topology as the MBN. Terminology used is different, but its structure could be described with the MBN features. Modular system design and application of the Business Bus (the way to exchange of business services) and the Business Ports (standard business and technological interfaces) offers a tool for optimal organizational design in the CA and their partners.

The Business Bus in this case is a virtual world of custom declaration processing that is separated from the physical world of goods, importers and custom houses. Locations of custom warehouses are the matter of convenience and agreement between the CA and an importer. It indicates the presence of vital elements of virtual organization.

4.4 "Fuzzy" Organizational Bonds

With the description of the virtual organization with the CPN we could identify and formalize some internal features like the strength of managerial and organizational bonds. We could study the mechanisms that make some positions in organization logically members of two or even more different organizations. Many positions or tasks in the ZZI are so strongly linked to the CA, and also opposite, that employees often don't know who their boss really is and to whom to report to in some cases.

It means that some organizational relations are not just fuzzy but they could also extend out of the organization. That is a relatively simple explanation for ambiguous boundary limits and boundary crossing.

The most interesting feature that the CPN modeling offers is its ability to model fuzzy information and decision processes that are the characteristics of the organized anarchy and also of virtual organizations. We were quite intrigued by the fact that we can implement so many ideas of organized anarchy directly on the virtual organizations.

4.5 Results

The reference models used confirm that the organization of the Custom Administration clearly demonstrates features of contemporary organizations with an efficient utilization of the Internet and even more hidden elements of virtual organization. In the absence of proven methodologies and indicators for assessment of virtual organizations we assessed the features of virtual organizations proposed by Mertens et al. (1998).

We transformed the whole problem into seven separated and independed assessments. Each feature was ranked on the scale from 1 to 100 and plotted on the radar chart (Figure 1). The picture reveals an uneven development of virtual organizations futures – some of them are very pronounced, others are still very close to the traditional organizations.

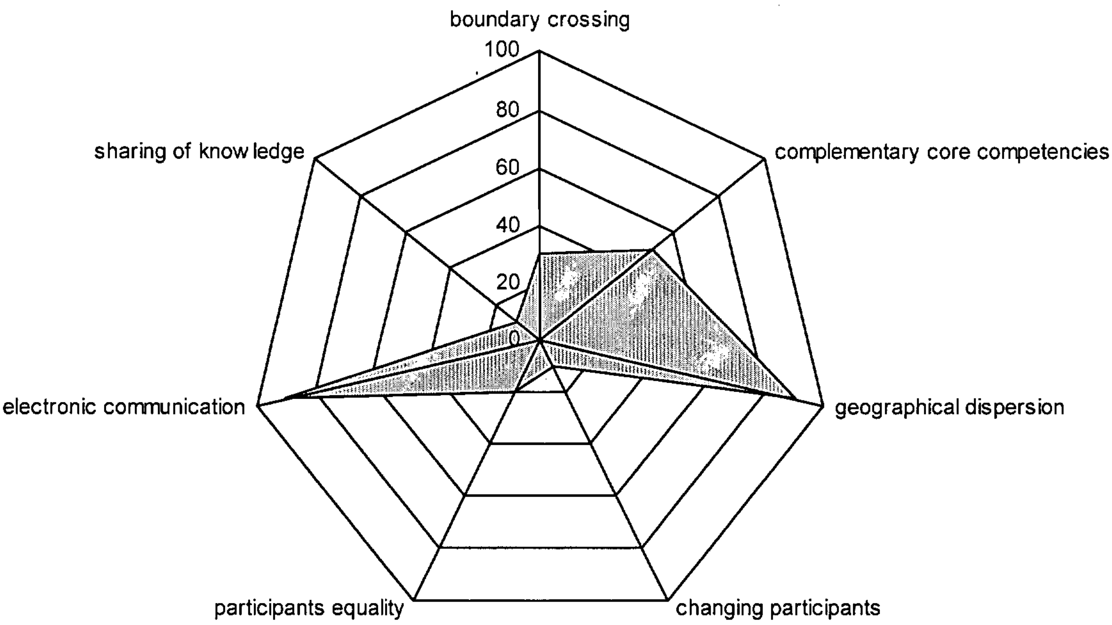


Figure 1: An assessment of the basic features of the virtual organizations – the case of the Custom Administration in Slovenia

An electronic communication and the geographical dispersion are very developed but, in every day business they are the easiest goals to achieve. They represent more technical than managerial challenge. Obviously, it is much more difficult to achieve managerial goals like sharing of knowledge, participation equality and particularly changing participants. They are prerequisites for introduction of the switching principle and metamanagement. Results also confirm what we could intuitively expect - boundary crossing and complementary core competencies are somewhere in the middle on the scale of managerial problems. They could be achieved in the next step, after introducing electronic communication and geographical dispersion on a broad scale.

Conclusion

The research has revealed that the CA had progressively developed towards virtual organization entirely intuitively. At the beginning, their main goal was to outsource development of their information system and to utilize the ICT, as much as possible. It confirms a well known fact noticed in the business community that we presently face an evolution rather than revolution towards virtual organizations. This evolution is spontaneous in technologically advanced environments. But, even if we accept the fact that the emergence of virtual organizations could be spontaneous the management still needs deeper insight into challenges of the new organizational paradigm. It will soon turn out to be one of the most important expertise of contemporary managers.

The case of the Custom Administration also presents a fine example of virtuality - the virtual world of custom declaration processing is separated from the physical world of goods, importers and custom houses. It provides such a high flexibility that Slovenian accession to the European Union won't require any changes in organization of the CA – their virtual world will be simply extended from Slovenian borders to the whole EU.

An assessment of the degree of virtuality proved to be a real challenge. We are still short of any useful methodology or a set of relevant indicators. Nevertheless, a simple case we investigated showed that we could combine different models in search for more holistic view on virtual organizations.

We were able to detect weaknesses and obstacles in managerial strategies and also to grade their goals from the easiest to the more complex. Technical issues like extensive introduction of the Internet are relatively easy to achieve and to manage. One of the most pronounced features of virtual organizations like boundary crossings is also quite common, even in the early phase of the development of virtual organization.

The ability to change participants has received the lowest grade in our research. It seems that the real managerial challenge is hidden in the switching principle and metamanagement. It could lead us to the conclusion that fully developed virtual organizations are still difficult to achieve. For that reason management needs much deeper understanding of challenges and obstacles in the transition from traditional to virtual organizations. Researchers could contribute with models and tools that would enable managers to set relevant goals and to asses their efforts.

References

[1] Bastide R. (1996): Approaches in unifying Petri nets and the Object-Oriented Approach, Working paper, L.I.S., Universitě Toulouse

[2] Bavec C. (1995) Object Oriented Modelling of Organization, Ph.D.Dissertation (in Slovenian), University of Ljubljana, Faculty of Economics

[3] Bavec C. (2001): "Modeling of Management Decision-making Processes in Organized Anarchy", Informatica, 25 (2001) 375–379

[4] Bavec C., Zorko Z. (2002): Evolution of Networked and Virtual Government Agencies – The Case from Slovenia, Procedings of 3rd European Conference E-Comm-Line 2002, September 26-27, 2002, Bucharest, Romania

[5] Cohen M. D., March J. G., Olsen J. P. (1972) A Garbage Can Model of Organizational Choice. Administrative Science Quarterly, 17 1-25

[6] Davidow W.H., Malone M.S. (1992) The Virtual Corporation, Harper Collins, New York

[7] Deng Y., Chang S.K. (1990): A G-net Model for Knowledge Representation and Reasoning, IEE trans. On Knowledge and data Engeneering, Vol. 3, No. 3

[8] Drucker P.F. (1999): Management Challenges for the 21st Century, Butterworth-Heineman

[9] Hesselbein F., Goldsmith M., Beckhard R. (1997): The Organization of the Future, The Drucker Foundation, Jossey Bas, San Francisco

[10] Ishaya T., Macaulay L.(1999): The Role of Trust in Virtual Teams, Proceedings of the 2nd International VoNet Workshop, September 23-24, 1999, Simowa Verlag Bern

[11] Jansen W., Steenbakkers W., Jägers H. (1999): Electronic Commerceand Virtual Organizations, Proceedings of the 2nd International VoNet Workshop, September 23-24, 1999, Simowa Verlag Bern

[12] Jensen K. (1992): Coloured Petri Nets, Basic Concepts, Analysis Methods and Practical Use, Vol. 1, Springer-Verlag, Berlin Heidelberg

[13] Klüber R., Alt R., Ősterle ZH. (1999): Emerging Electronic Services for Virtual Organizations – Concept Framework, Proceedings of the 2nd International VoNet Workshop, September 23-24, 1999, Simowa Verlag Bern

[14] Mertens, P., Griese J., Ehrenberg D. (1998): Virtuelle Unternehmen und Informationsverabeiting, Springer, Berlin

[15] Morabito J., Sack I., Bhate A. (1999): Organization Modeling – Innovative Architectures for 21st Century, Prentice Hall

[16] Mowshowitz A. (1997): Virtual Organization: Avision of Management in the Information Age, The Information Society, Vol. 10

[17] Mowshowitz A. (1999): The Switching Principle in Virtual Organization, Proceedings of the 2nd International VoNet Workshop, September 23-24, 1999, Simowa Verlag Bern

[18] Strausak N. (1998): "Résumé of VoTalk", VoNet Workshop, April 27-28,1998, Simowa Verlag Bern

Using Image Segmentation as a Basis for Categorization

Janez Brank
Department of Intelligent Systems
Jožef Stefan Institute, Ljubljana, Slovenia
janez.brank@ijs.si

Keywords: image categorization, segmentation, generalized kernels

Received: June 26, 2002

Image categorization is the problem of classifying images into one or more of several possible categories or classes, which are defined in advance. Classifiers can be trained using machine learning algorithms, but existing machine learning algorithms cannot work with images directly. This leads to the need for a suitable way of representing or describing images such that learning algorithms can work with them. We consider a representation based on texture segmentation and a similarity measure between segmented images which has been used successfully in the related area of image retrieval. A generalized kernel for use with the support vector machine (SVM) algorithm can be built from such a similarity measure. We compare this approach with a more straightforward representation based on autocorrelograms, and we show that these two representations can be combined to obtain classifiers with higher categorization accuracy.

1 Introduction

Besides textual and relational data, people increasingly have to deal with pictorial data, or data in the form of images. Large *pictorial databases* are being produced as archives digitize their collections, and additionally the World Wide Web contains a huge number of images. Apart from purely technical problems of storing and processing such large amounts of data, the emergence of large collections of images opens the problems of enabling the users to make sense of this data and find what they need. *Image categorization* deals with one aspect of this problem: given a set of images and a set of predefined categories or classes, we assume that each image should belong to one or possibly several of these categories. For a large collection it would be impractical to have a human observer categorize all the images, so we want to be able to classify the images automatically after a small number of images has been classified manually to be used for training the automatical classifiers.

However, this view of image categorization as a machine learning task immediately opens up a new problem: existing machine learning algorithms generally cannot work with images directly. Instead, they often assume they will be dealing with instances described by vectors or tuples. We need to be able to represent images using structures of this kind to make use of existing machine learning algorithms.

1.1 Related work in image retrieval

We can build on existing work in *image retrieval*, which is a related area where the problem of representation has already been encountered. In image retrieval, the user poses a query to the system and the system should find images that are somehow relevant to the query. Thus a way of representing the query, a way of representing images, and a way of comparing a query and an image (to determine if the image is relevant with regard to this query) are needed. One approach that is both technically feasible and useful enough to be commonly used in practice (e.g. in web image search engines such as Google) is to describe each image using a few keywords, and the user's query can then request images whose description includes or excludes particular keywords. However, this approach is only feasible if textual descriptions of images can be obtained automatically (e.g. from the HTML file that linked to an image); it is usually too costly to have a human maintainer prepare such descriptions manually for a larger database. In addition, this textual approach suffers from problems of polysemy: different people would use different words to describe an image, and the same words may mean different things to different people. Therefore it is often desirable to rely solely on what can be automatically extracted from the images themselves. The user's query is then often simply a request to look for images similar to a given query image or sketch (this approach is known as "querying by content", or "content-based image retrieval").

There are several close parallels between image retrieval and image categorization. In categorization, if a new image is similar to training images from a particular category, it should probably itself belong to that category; in content-based image retrieval, if an image from the database is similar to the query image, it should probably be shown to the user. Thus we see that both areas need a way of representing images and assessing similarity between them. Many image representations and similarity measures have been proposed in image retrieval, and we would like to examine some of them from the point of view of image categorization as well.

One popular class of image representations is based on simplifying the image by approximating the color of each pixel by the nearest color from a predefined and fixed color palette; this can also be seen as partitioning (or *quantizing*) the space of all possible colors. Some information is then recorded about the presence of each color on the image. When simply the proportion of the image covered by (the pixels of) that color is stored, the resulting description is called a *histogram* [11]. However, this disregards all spatial information (how the color is distributed around the image): for example, a large patch of red would affect the histogram of an image in the same way as a large number of red pixels scattered all over the image, which is surely undesirable.

Several improved histogram-like representations of images have been proposed. For example, an *autocorrelogram* [4] records, for each color *c* and for a few small integers *d*, the probability that a pixel, chosen randomly at distance *d* from a randomly chosen pixel of color *c*, will itself be of the color *c*. This retains information about the amount of a color present on the image, but also records something about the spatial arrangement of each color. Still, all "global" representations of this type can be seen as somewhat rigid as they record a strictly fixed amount of data for each image. They cannot take into account the fact that some images are more complex than others, that an image may contain several objects, or that it may be helpful to distinguish between an (interesting) object and (uninteresting) background.

1.2 Image segmentation

Another, more sophisticated, class of image representations is based on *segmentation*, or dividing an image into a set of *regions* such that each region is roughly homogeneous in color and/or texture. Each image is then represented by a set of regions; each region is typically described by a short vector that is a by-product of the segmentation procedure (containing e.g. the average color of the region, information about texture, and so on). Additionally, the location of each region on the image (i.e. which parts of the image are covered by that region) is often recorded as well. In general, regions might overlap, and each region might itself be composed of several disjoint parts; this is not necessarily problematic as they need not be shown to the user, and image similarity measures usually permit the regions to be disconnected, and sometimes work with overlapping regions as well. Representations based on segmentation can adapt well to differences in complexity between images, and have been used successfully in image retrieval [NRS99, WLW00].

Various segmentation algorithms have been proposed in the context of image retrieval [NRS99, WLW00]. These approaches are usually based on dividing the image into a grid of small "windows" (e.g. 4×4 pixels); each window is described by a short vector (containing e.g. the average color and possibly a few coefficients from the higher-frequency bands of a wavelet transform, in order to capture the presence of edges or texture), and these vectors are then clustered. Each of the resulting clusters contains vectors that lie close together in their vector space, and such vectors hopefully correspond to windows that are similar in appearance; therefore it makes sense to form a region from such a group of windows. The region thus obtained can be described by the centroid of the cluster, i.e. by the average of the vectors that describe the windows from which the region was formed.

To use segmentation for image retrieval, it is also necessary to introduce a measure of similarity between segmented images. Such measures usually examine pairs of individual regions (one region from each image) and combine the measures of similarity or difference between regions into a single similarity measure between entire images. For example, the *integrated region matching* (IRM) measure [8] defines the distance between two images as a weighted sum of distances between regions, in which the weights are chosen so as to allow larger regions to have a larger influence on the similarity between images.

To use the representations described above for image categorization, one could use global representations (e.g. autocorrelograms) in combination with any of several machine learning algorithms (such as support vector machines, SVM); or use a segmentation-based similarity measure with an algorithm that allows an arbitrary similarity measure to be plugged into it (e.g. the nearest-neighbor method). However, our earlier work [1] has shown that the nearest neighbor method, in combination with segmentation-based image similarity measures, results in rather unimpressive performance in comparison to SVM and global representations. It is therefore our goal to try using segmentation together with support vector machines. The main challenge here is that the SVM in its original formulation assumes all training and test examples to be described by vectors with the same number of components, while in the case of segmentation the description of each image has more structure than that, and the number of regions can also vary from image to image.

2 Support vector machines

Support Vector Machines (SVMs) [3] are a relatively recent family of machine learning algorithms that have been used successfully in many application domains. In the most elementary form of this method, we assume that each training example is a vector from some *d*-dimensional real space, and that there are exactly two classes, called positive and negative. Several extensions to multiclass problems are possible [5], usually by converting one multiclass learning problem into several two-class problems (e.g. training one classifier for each pair of classes to separate members of one class from those of the other class).

In SVM learning, we want to separate the positive vectors from the negative ones using a hyperplane such that the positive training vectors lie on one side of the plane and the negative ones lie on the other side.

Additionally, to help make the classifier more robust and more reliable for use on unseen test vectors, we want the training vectors to lie as far from the separating hyperplane as possible. Maximizing this distance (known as the *margin*) from the plane to the nearest training example can be cast as an optimization problem in the following way.

Let x_i be the *i*th training vector, and y_i its label (which equals +1 for positive examples and–1 for negative training examples. A hyperplane can be described by the equation $w^Tx + b = 0$, where w is the "normal", i.e. a vector perpendicular to the plane, and b is a threshold that determines the actual location of the plane in space. w^Tx denotes the dot product of the vectors w and x. Given a particular vector x, we can determine what side of the plane it lies on by examining whether $w^Tx + b$ is positive or negative. However, to ensure that the training examples do not lie too close to the plane, we must also insist that $w^Tx + b$ has a large enough absolute value. We can describe this using the following conditions:

$$y_i = 1 \Rightarrow w^Tx_i + b \geq 1 \quad \text{and} \quad y_i = -1 \Rightarrow w^Tx_i + b \leq -1,$$

or, more concisely: $y_i(w^Tx_i + b) \geq 1$ for all training instances i. If all training examples satisfy these conditions, the space between the hyperplanes $w^Tx + b = 1$ and $w^Tx + b = -1$ is empty; to maximize the breadth of this margin space, we need to maximize the distance between these two planes, which equals $2/||w||$. Maximizing the margin is thus equivalent to minimizing $||w||^2$ subject to the above conditions.

This optimization problem is usually also extended to allow some training instances to be misclassified (or at least lie within the margin, though perhaps on the correct side of the separating plane) if this leads to a wider margin on the other training instances (the *soft margin* formulation of SVM).

Solving the optimization problem gives us the values of w and b, and the resulting classifier simply works according to the formula $\mathit{prediction}(x) = \mathrm{sgn}[w^Tx + b]$.

Using standard techniques from optimization theory, this optimization problem can be transformed into a "dual" form. It turns out that the dual form, as well as the resulting classification rule, can be expressed so that the training vectors need never be accessed directly, as long as we are able to compute the dot product of any two vectors. In particular, the normal w can be written as $w = \Sigma_i\, \alpha_i y_i x_i$, where the α_i coefficients are obtained by solving the dual optimization problem. The classifier can then be described as $\mathit{prediction}(x) = \mathrm{sgn}[\mathrm{b} + \Sigma_i\, \alpha_i y_i x_i^T x]$.

Now suppose we used some mapping φ to map our original instances x_i into some other (possibly higher-dimensional) vector space F. Let $K(x_i, x_j) := \langle\varphi(x_i), \varphi(x_j)\rangle_F$ be a function that, given two instances x_i and x_j, computes the dot product $\langle\cdot,\cdot\rangle_F$ (in the new space F) of their images $\varphi(x_i)$ and $\varphi(x_j)$ under the mapping φ. It follows from the above that we could train a hyperplane in F without ever working with the mapped vectors $\varphi(x_i)$ explicitly, as long as we are able to compute $K(x_i, x_j)$ for any two vectors x_i and x_j. The function K defined in this way is known as a *kernel*. The importance of kernels arises from the fact that the mapping φ need not be linear, and for a nonlinear φ a hyperplane in F could correspond to some highly nonlinear separation surface in the original space. In this way, kernels allow the SVM algorithm to induce nonlinear models while preserving the optimization framework essentially intact. The appeal of kernels stems from the fact that a wisely chosen function K can be simple to compute and yet correspond to a complex nonlinear mapping into some very high-dimensional space F.

A kernel corresponds to a dot product in some vector space and can therefore in some sense be seen as a sort of similarity measure: the dot product of two vectors (if their length is fixed) is greatest when they point in the same direction, and then decreases as the angle between them increases, eventually becoming 0 (for orthogonal vectors) and even negative, reaching the minimum if the two vectors point in exactly the opposite direction.

However, the converse is not true: that is, not every similarity measure corresponds to a scalar product in some vector space. If we used a non-kernel similarity measure as if it were an actual kernel, we would no longer have the mathematical guarantees that the SVM training algorithm would converge, and even if it converged there would be no theoretical grounds to expect the resulting classifier to have good performance.

3 Generalized kernels

Generalized SVMs have been proposed by Mangasarian [9] to allow an arbitrary similarity function to be used in a way analogous to a kernel. In the previous section we have seen that SVM can learn nonlinear models of the form

$$\mathit{prediction}(x) = \mathrm{sgn}[\mathrm{b} + \Sigma_i\, \alpha_i y_i K(x_i, x)]$$

where $K(x_i, x) = \langle\varphi(x_i), \varphi(x)\rangle_F$ for some mapping φ to some space F and some dot product$\langle\cdot,\cdot\rangle_F$ in F.

Now if some arbitrary function K were used instead of a proper kernel function, again giving us a classifier of the form $\mathrm{sgn}[b + \Sigma_i\, \alpha_i y_i K(x_i, x)]$, this might still be a perfectly reasonable and useful classifier, but it wouldn't necessarily correspond to some hyperplane in some vector space F to which the instances x_i and x might have been mapped. Thus we couldn't obtain the α_i values using the criterion of maximizing the margin, because there wouldn't even be a hyperplane whose margin to maximize. Instead, [9] proposes to minimize the value $\alpha^T H\alpha$ (subject to the same constraints as before, i.e. that our training instances should lie on the correct side of the separation surface) for some positive definite matrix H. (This problem has a very similar structure to the dual form of the original SVM optimization problem, and is in fact equivalent to it if K really corresponds to a dot product and a suitable matrix H is chosen.)

In the simplest case of the generalized SVM, we would take $H = I$ (the identity matrix) and thus minimize $\Sigma_i\, \alpha_i^2$. This can be interpreted intuitively as looking for a separation surface that can be expressed in the simplest possible way, possibly with many α_i equal to 0 (i.e. without really using the training example x_i in the description of the separating surface).

It can be shown that the formulation for $H = I$ is equivalent to mapping each instance x into the vector $(K(x, x_1), \ldots, K(x, x_n))$ of its "similarities" (as measured by K) to all the training instances $x_1, \ldots, x_n$, and then using an ordinary linear support vector machine over this new representation. For the problem of image categorization, this amounts to the intuitively appealing suggestion that two images should be treated as similar if they exhibit a similar pattern of similarities to known training images.

4 Region clustering

In this section we consider another approach to using segmentation-based representations for image categorization. Each image has its own set of regions and regions belonging to different sets are in a sense quite independent of each other. This leads to the need for special similarity measures that compare two images by considering all pairs of regions and aggregating the similarities of regions into a measure of similarity between the images.

As an alternative, we propose to bring the region-based representations of images to a "common denominator" by clustering the descriptions of all the regions of all the training images. The hope here is that each cluster would correspond to a group of similar regions from several images, while regions from separate clusters would be quite different in appearance. Thus, when comparing two images, if a region of one image belongs to a different cluster than some region of the other image, there would be no need to compare these two regions in any particular way, because knowing that they belong to different clusters already indicates that they are different in appearance and cannot really contribute towards the similarity of the two images under consideration.

Therefore, an image would then be described by recording, for each cluster of regions, what proportion of the area of this image is covered by regions of this cluster. If there are d region clusters, each image would now be represented by a d-dimensional real vector (with possibly many zero-value components, as there would probably be much more clusters than an average image has regions). With all images represented in this same d-dimensional space, we can then use the ordinary linear support vector machine to train classifiers.

5 Experimental evaluation

To compare the approaches described in the previous sections, we conducted experiments on the *misc* database, which is publicly available (http://www-db.stanford.edu/IMAGE/) and has already been used in image retrieval literature [13, 10], as well as in our earlier work on image categorization [1]. This database contains approximately 10000 small photographic images (of sizes around 128 by 96 pixels). It is thematically very diverse.

We selected 1172 images from the database and manually assigned each of them to one of 14 categories (butterflies, US flag, sunsets, autumn, flowers, planets, satellite images of Earth, cars, mountains, clouds, sea, surfboards, sailboats, prairie animals). The intention of this selection was to have categories of varying size and difficulty. The smallest category (flags) contains 32 images, and the largest (sunsets) contains 224 images. Some of the categories, such as sunsets or flowers, have characteristic and easily recognizable color distributions, while some categories are quite similar in this respect and would therefore be more difficult to distinguish (e.g. sea and clouds, both of which have a lot of blue and white pixels).

To train the SVM classifiers, we used the LibSvm [2] program, which has the advantage of natively supporting multiclass problems. It uses the all-pairs approach to convert a multiclass problem to several two-class problems: for each pair of classes, a classifier is trained to distinguish members of one class from members of the other class. To classify a new example, it is shown to all the classifiers, each of which then votes for either one or the other of the two classes which it has been trained to separate. The class with the greatest number of votes is then adopted as the final prediction.

We compared the following approaches to image categorization:

1. Images are represented in the HSV (hue, saturation, value) color space, which is quantized into 256 colors (the H axis is split into 16 equal ranges and the S and V axes into 4 equal ranges). Each image is then described by an autocorrelogram in the resulting quantized color space. The autocorrelograms are 1024-dimensional vectors and are used as input for linear SVM.

2. Images are segmented into regions using the segmentation algorithm from WALRUS [10]. The IRM similarity metric [8] is then used to construct a generalized kernel as described in Section 3 above. In other words, each image is represented by a vector of its IRM similarities to all the training images; these vectors are then used as input for linear SVM.

3. Images are segmented as in the previous paragraph. Each region is described by a short (12-dimensional) vector, which is a by-product of the segmentation algorithm. The vectors resulting from all the regions of all the training images are then clustered (here we use the same algorithm, BIRCH [14], that is also used by WALRUS during segmentation). An image is then described by a sparse vector specifying what proportion of the area of the image is covered by regions from each region cluster. Depending on the parameters of the segmentation, the average number of regions per image might vary from less than ten to more than a hundred; then, depending on the parameters of the clustering, the number of region clusters (and hence the dimensionality of the space in which our images are now represented) is usually on the order of a few hundred.

Once images are represented in this way, linear SVM can be used to train classifiers for them.

For the sake of comparison, we also report the performance of the nearest neighbor method with the IRM similarity metric (that is, each image is predicted to belong to the same class as the most similar training image). All performance values reported here are averages (and standard errors) based on tenfold stratified cross-validation.

Method	Classification accuracy
Autocorrelograms	80.2 % ± 1.3 %
Generalized kernels	79.0 % ± 1.3 %
Region clustering	70.0 % ± 1.6 %
Nearest neighbors + IRM	69.1 % ± 1.3 %

As expected, the nearest-neighbor method is in general less successful than the approaches based on SVM. However, it turns out that the two segmentation-based approaches do not outperform the representation based on autocorrelograms. The performance of the generalized kernel method is not significantly different (using a paired t-test) from that of autocorrelograms, and the generalized kernel method has the additional disadvantage of much greater computational cost.

In addition, the performance of the region clustering approach is remarkably poor. A closer examination suggests that the partitioning of regions into region clusters is problematic and unstable. For example, if the centroid of each cluster is recorded and then all regions are distributed to the cluster with the nearest centroid, most of the regions will tend to move to a different cluster than they were originally attached to. This means that two otherwise similar regions might fall into different clusters by pure chance, and the similarity between their images would thus go unnoticed. The authors of the BIRCH clustering algorithm were aware of the possibility of such problems, and proposed several redistribution passes where the regions are redistributed to the nearest centroids, but in our experiments this did not lead to a really stable partition even after five or ten such passes.

An alternative way of making use of the region clustering approach might be to include the test images in the region clustering phase. This really amounts to a form of transduction, i.e. using test data as if it was simply additional unlabeled training data. It ensures that both the training images and the test images are really being represented in a space that treats both groups of images equally. In this setting, the performance of the region clustering increases considerably, and it achieves an accuracy of 86.4 % ± 1.0 %. However, for the comparison with other methods to be fair, transduction should also be included in the SVM learning process. Since LibSvm does not support tranduction, we used the SvmLight program [6] for these experiments; it implements Joachims' transductive SVM algorithm [7]. With transductive SVM, region clustering achieves an average accuracy of 91.9 % ± 1.0 %, while autocorrelograms achieve an accuracy of 90.7 % ± 1.1 %. Although this difference is not really significant from a practical point of view (a t-test shows that it is statistically significant at a confidence level of 0.945, slightly below the usual 0.95), it suggests that the region clustering approach does have at least some potential to be useful.

Finally, we also considered combining several representations. An analysis of classification errors shows that classifiers based on different representations often make mistakes on different test images; that is, a it often happens that a test image is classified correctly by one classifier but incorrectly by another. For example, consider the classifiers based on autocorrelograms and on generalized kernels (with the IRM measure). Of the 1172 images, 828 are classified correctly by both; 120 only by the former; 100 only by the latter; and 128 are misclassified by both. (To obtain these numbers, each image was classified by a model obtained from that 90% of the dataset which does not contain the image under consideration.)

Thus it seems that some advantage could be gained by combining the features of both of these representations. Many approaches exist for combining several classifiers, but with SVM, this can be done in a particularly simple way. If we have two representations, ϕ_1: $X \rightarrow F_1$ and ϕ_2,: $X \rightarrow F_2$, combining their features (or attributes) would be equivalent to a new representation ϕ: $X \rightarrow F_1 \times F_2$ defined by the formula $\phi(x) = (\phi_1(x), \phi_2(x))$. Now if the kernels $K_1(x_i, x_j)$ and $K_2(x_i, x_j)$ correspond to some dot product on F_1 and F_2, respectively, the function $K(x_i, x_j) := K_1(x_i, x_j) + K_2(x_i, x_j)$ is a dot product on $F_1 \times F_2$. Thus we can obtain the equivalent of a combined representation simply by computing the sum of two kernels.

In our experiments, the combination of the autocorrelogram representation and the generalized kernel using the IRM similarity measure achieved a categorization accuracy of 83.7 % ± 1.4 %. A t-test shows that this performance level is significantly better than that of either of these two representations individually.

6 Conclusions and future work

Our experiments show that it is difficult to use segmentation-based image representation methods in image categorization. Relatively complex ways of using information obtained from segmentation, such as the generalized kernel approach and (to a lesser extent) the region clustering approach, have been found able to compete with a simpler and more straightforward approach such as autocorrelograms but not to significantly outperform it. In the presence of unlabeled test images, the region clustering approach performs really well (relative to other representations) if a transductive SVM learner is not available. We have shown that it is possible to use segmentation-based representation in combination with another representation to achieve a small but significant increase of categorization accuracy.

We nonetheless believe that there must be ways of using segmentation more profitably for image categorization, just as it is used in image retrieval, and that this is still an interesting topic for future work. In particular, it would be interesting to further explore the influence of the clustering algorithm used in the region clustering approach, and to look for more stable clustering algorithms that would allow the region clustering approach to perform better in the inductive in additional to the transductive setting.

In addition, as segmentation is a relatively complex task, and segmentation algorithms usually depend on several parameters, it would be interesting to explore the influence of these various parameters on the segmentation (and consequently on image categorization) in a more systematic way.

The region clustering approach could also be augmented by taking the similarity between different clusters into account. Currently, regions that belong to different clusters contribute to different components of the sparse vectors that describe our images, and therefore whatever similarity might exist between two regions from different clusters cannot contribute anything towards our algorithm's perceived similarity between their two images. Acknowledging that regions can be at least somewhat similar even if they belong to different clusters might lead to an improved representation, but would (if taken to the extreme case) again require us to do the equivalent of comparing every region of one image with every region of the other image, which is what the region clustering approach was designed to avoid in the first place. Perhaps one could determine (from the region clustering process itself), for each region cluster, just a few most similar clusters and then compare pairs of regions from the closely similar clusters but ignore pairs of regions from entirely unrelated clusters.

Region clustering could also be integrated with segmentation. Currently, segmentation is being performed separately on each image, by clustering the descriptions of its 4×4 pixel windows; then, the region descriptions of all the images in the training set are clustered to form region clusters. These two steps could be merged by considering the descriptions of all windows from all the images as a single large set and performing clustering on this. Each image would then be represented by a vector of values showing what proportion of the image is covered by windows belonging to a particular cluster.

Combination of several kernels could also be pursued further, particularly in the direction of combining more than two classifiers and using weighted sums of kernels.

Additionally, the methods considered here should be tested on other datasets, as (given that widely different methods achieve highly similar categorization accuracy values on the present dataset) it is perhaps simply unrealistic to expect better performance on the current dataset, as the categories have an essentially "semantic" motivation that the current image representation methods simply cannot capture.

References

[1] J. Brank: *Machine learning on images* (in Slovenian). Proc. IS 2001, Ljubljana, 2001, pp. 152–55.

[2] C.-C. Chang, C.-J. Lin: *LibSVM: a library for support vector machines* (version 2.3). Dept. of Comp. Sci. and Inf. Eng., Nat'l. Taiwan University, April 2001.

[3] C. Cortes, V. Vapnik: *Support-vector networks.* Machine Learning, 20(3):273–297, September 1995.

[4] J. Huang, S. R. Kumar, M. Mitra: *Combining supervised learning with color correlograms for content-based image retrieval.* Proc. 5th ACM Multimedia Conf., Seattle, USA, 1997, pp. 325–334.

[5] C.-W. Hsu, C.-J. Lin: *A comparison of methods for multi-class support vector machines.* Dept. of Comp. Sci. and Inf. Eng., Nat'l Taiwan University, April 2001.

[6] T. Joachims: *Making large-scale SVM learning practical.* In: B. Schölkopf et al. (eds.), Advances in Kernel Methods. MIT Press, 1999, pp. 169–184.

[7] T. Joachims: *Transductive inference for text classification using support vector machines.* Proc. 16th ICML, Bled, Slovenia, 1999, pp. 200–209.

[8] J. Li, J. Z. Wang, G. Wiederhold: *IRM: Integrated region matching for image retrieval.* Proc. 8th ACM Multimedia Conf., Los Angeles, 2000, pp. 147–156.

[9] O. L. Mangasarian: *Generalized support vector machines.* In: A. J. Smola et al. (eds.), Advances in Large Margin Classifiers, MIT Press, 2000, pp. 135–146.

[10] A. Natsev, R. Rastogi, K. Shim: *WALRUS: a similarity retrieval algorithm for large databases.* Proc. ACM SIGMOD, 1999, pp. 395–406.

[11] M. J. Swain, D. H. Ballard: Color *indexing.* Int. Journal of Computer Vision, 7(1):11–32, Nov. 1991

[12] J. Z. Wang, J. Li, G. Wiederhold: *SIMPLIcity: Semantics-sensitive integrated matching for picture libraries.* Advances in Visual Inf. Systems, 4th Int. Conf., 2000, pp. 360–371.

[13] J. Z. Wang, G. Wiederhold, O. Firschein, S. X. Wei: *Content-based image indexing using Daubechies' wavelets.* Int. Journal of Dig. Lib., 1(4):311–328, December 1997.

[14] T. Zhang, R. Ramakrishnan, M. Livny: *BIRCH: An efficient data clustering method for very large databases.* Proc. ACM SIGMOD, 1996. pp. 103–114.

Recognition of Image Authenticity Using Significant DCT Coefficients Quantization

Chin-Chen Chang*, Jun-Chou Chuang* and Tung-Shou Chen**
* Department of Computer Science and Information Engineering
National Chung Cheng University Chiayi, Taiwan 62107, R.O.C.
Phone:886-5-2720411 ext. 33100 FAX: 886-5-2720859
E-mail: {ccc, lzchung}@cs.ccu.edu.tw

** Department of Information Management
National Taichung Institute of Technology Taichung, Taiwan 404, R.O.C.
Phone: 886-4-22211181 ext. 2213 FAX: 886-4-22233545
E-mail: tschen@ntit.edu.wt

Keywords: JPEG, DCT, image authentication, signature, watermarking

Received: May 11, 2002

Traditional image authentication methods cannot preserve the authenticity after the processing of the JPEG lossy compression. This is because the JPEG lossy compression destroys the secret embedded in the image. However, the JPEG lossy compression has been so widely used that it simply exists everywhere. Thus, this kind of modification should be taken into consideration. To improve traditional image authentication methods, we shall propose a new method that can not only prevent images from being tampered with but also allow reasonable JPEG lossy compression. Our method works by extracting some significant discrete cosine transform (DCT) coefficients and setting a compression tolerant range. The extracted DCT coefficients will be able to survive when the image is not further modified or lossily compressed. The experimental results show that our method can tolerate JPEG lossy compression while keeping the image from illegal modifications.

1 Introduction

Digital images have been widely used in the computer world. However, without due protection, they can be easily modified with image-processing tools. As a result, image authentication has become an important research issue. If someone has maliciously manipulated an image, the image authentication system has to have the ability to point out the exact places that have been modified.

Image authentication methods can be classified into two categories. They are digital-signature-based methods (Bhattacharjee & Kutter 1998, Friedman 1993, Lu & Liao 2000) and watermarking-based methods (Kundur & Hatzinakos 1999, Schneider & Chang 1996, Lin et al. 2000, Hung et al. 2001). In a digital-signature-based method, the original image is hashed and then encrypted via the public key encryption (Rivest et al. 1978). The encryption result is called the "signature" of the image. On the other hand, in a watermarking-based method, watermarks are first embedded into an image and later extracted from it to verify the authenticity. An image is said to have gone through malicious manipulations if the retrieved watermarks are not identical to the corresponding original watermarks.

In 2001, Hung et al. proposed an image authentication method based on the DCT coefficients. Their method performs the vector quantization (VQ) technique on the original image. The extracted VQ indices are called features. These features are embedded into the DCT coefficients located in the middle-frequency part of each DCT block of the original image. The embedded features can be used for image tamper detection and recovery. Unfortunately, this method cannot tolerate JPEG lossy compression (Pennebaker & Mitchell 1993). The image after JPEG lossy compression may still be considered acceptable.

Hence, we shall propose a new method here in this paper that can prevent images from being tampered with even when they have gone through JPEG lossy compression. Our method takes advantage of two distinct properties of the image and extracts some features from these two properties for further verification.

The first property is that the DCT coefficients located in the upper left positions contain most of the information in an image block. To make use of the property, we extract some features from those significant DCT coefficients and modify them within a pre-defined compression-tolerant range. The extracted features can tolerate JPEG lossy compression if the significant DCT coefficients do not go beyond the compression-tolerant range. Conversely, if the image has been maliciously manipulated, the modified DCT coefficients will surely exceed the compression-tolerant range, and that is how we still detect illegal modifications.

The second property of the JPEG lossy compression is that, after the encoding and decoding processes, the high frequency part of each block will be lost. This is because most of the DCT coefficients located in the lower right positions will become zero after JPEG lossy compression. Thus the 64 pixel values of each smooth image block will move toward their mean value. Based on this property, the proposed method calculates the maximum difference between the mean value and each pixel value for each block. The maximum difference should be decreased after JPEG lossy compression. In other words, if the maximum difference gets increased, then we know that the image is tampered with.

This paper is organized as follows. In Section 2, we shall review DCT property and the image authentication method by Hung et al. After that, our proposed image authentication method will be described in Section 3. In Section 4, some experiments and security analyses will be discussed. Finally, the conclusions will be presented in Section 5.

2 Related Work

2.1 Discrete Cosine Transformation

DCT is an image transformation method that can transform each pixel in the spatial domain into the frequency domain. It is very popular in applications such as image compression, image processing, watermarking, etc. The two-dimension transformation of DCT and inverse DCT (IDCT) are defined as follows respectively.

$$F(i,j)=\frac{4}{N^2}c(i)c(j)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\cos(\frac{(2x+1)i\pi}{2N})\cos(\frac{(2y+1)j\pi}{2N}). \tag{1}$$

$$f(x,y)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}c(i)c(j)F(i,j)\cos(\frac{(2x+1)i\pi}{2N})\cos(\frac{(2y+1)j\pi}{2N}). \tag{2}$$

Here $c(i)$, $c(j)=1/\sqrt{2}$ for i, j=0, otherwise $c(i)$, c(j)=1. Besides, $f(x, y)$ is the pixel value in the spatial domain and $F(i, j)$ is the DCT coefficient in the frequency domain. Generally speaking, 2-D DCT is often used to process blocks of 8*8 pixels each. Therefore, the parameter N is set to be 4.

An important property of DCT is that after DCT transforming, for an image block, the DCT coefficients located in the upper left positions contain most of the energy. That is to say, even if we only use those DCT coefficients to reconstruct a block by IDCT transformation, the image features of the reconstructed block will still be the same as the original block. That is why we can extract some significant DCT coefficients located in the upper left positions as our significant features.

2.2 The Hung et al.'s Image Authentication

2.2.1 DCT-Based Embedding Procedure

Hung et al.' s image authentication can be divided into two stages: feature extraction and feature embedding. In the feature extraction stage, they employ the VQ compression technique to process the original image. Note that the encoding block of VQ is 4×4 pixels. Therefore, they can obtain VQ indices as features, or called watermarks. Two 16-dimensional codebooks Ω_d and Ω_r are used to encode the original image O respectively. The encoding results are W_d and W_r. W_d is the detection feature and W_r is the recovery feature. These two distinct features are used for image tampering detection and recovery.

In feature embedding, W_d and W_r are embedded into the DCT coefficients located in the middle-frequency part of each block of the original image. Here the DCT block is 8×8 pixels. The detection features of an image block are embedded into its block itself, but the recovery features are not because the recovery features will be destroyed if its block gets tampered with. To solve this problem, they use pseudorandom permutation operation to embed the recovery features of one block in another block.

Consider an image block B_i. Assume the embedded features are $W_i=(w_1, w_2,\ldots, w_s)$ and the middle-frequency coefficients are $M_i=(m_1, m_2,\ldots, m_s)$. Here s is the total bit length of the features w_i^d and w_i^r. They use the hiding function and the pseudorandom number with a seed s_k to embed the features into the DCT coefficients located in the middle frequency part. The hiding function is defined as follows.

$$\mathrm{H}(m_j,w_j)=\begin{cases}\frac{m_j}{\lfloor 4\alpha\rfloor}\times 4\alpha+2\alpha & \text{if} \quad w_j=1\\ \frac{m_j+2\alpha}{\lfloor 4\alpha\rfloor}\times 4\alpha & \text{if} \quad w_j=0\end{cases}. \tag{3}$$

Here $\alpha\geq 1$ is the adjusting magnitude. A large α value will make an image become more distorted. On the other hand, a small α value cannot endure an error. After the embedding procedure is finished, IDCT is performed upon the above DCT block to obtain the embedded image. The embedded image can be published when the verification information is embedded into the original image.

2.2.2 Tamper Detection and Recovery Procedure

Given a test image O'. They use the embedded detection features to decide if the test image O' has been tampered with or not. If the answer is yes then they use the recovery procedure to recover the modified places. First, they use DCT transformation on the test image O', and then they retrieve the detection features $\tilde{W}_i^d$ and the recovery features $\tilde{W}_i^r$ in each DCT block. They use the extraction function and the pseudorandom number with a seed s_k to extract the features in each DCT block. The extraction function is defined as follows.

$$E(m_j)=\begin{cases}0 & \text{if} \quad [(m_j+\alpha)\bmod(4\alpha)]<2\alpha \\ 1 & \text{otherwise}\end{cases} \quad . \qquad (4)$$

Here m_j is the coefficients in the middle-frequency part. As for the recovery features, because the pseudorandom permutation operation was previously used to permute them, the inverse pseudorandom permutation should be performed here. They compute $\tilde{W}_i^{\prime\prime r}=P^{-1}(\tilde{W}_i^{\prime r})$, where P^{-1} is an inverse permutation function.

Next, they perform VQ on the test image O'. The 16-dimensional codebook Ω_d is used to encode the test image O'. Consider a DCT block $\widetilde{B}_i$ of the test image. An 8×8 DCT block $\widetilde{B}_i$ can be divided into four encoding VQ-blocks (b_{i1}, b_{i2}, b_{i3}, b_{i4}) with 4x4 pixels each. Let D_i denote (d_{i1}, d_{i2}, d_{i3}, d_{i4}) and d_{ij} be the VQ index of the closest codeword of the subblock b_{ij}. A block $\widetilde{B}_i$ is said to have been tampered with if $D_i \neq \tilde{W}_i^d$ and RMSE($B_i^{\prime}, \widetilde{B}_i$)>$t$, where RMSE is the root mean square error (RMSE) between two blocks, and t is a threshold . Here $\tilde{W}_i^d$ is the detection feature of the block $\widetilde{B}_i$, and $B_i^{\prime}$ is the reconstructed block generated following the side-match VQ method (Chang & chen 1993) as follows.

When the method detects that a DCT block $\widetilde{B}_i$ has been tampered with, then the recovery procedure is performed. Assume $\widetilde{B}_i$=(b_{i1}, b_{i2}, b_{i3}, b_{i4}) has been illegally modified. They use the recovery features $W_{i1}^{\prime\prime r}$ and $W_{i4}^{\prime\prime r}$ by looking up the codebook Ω_r to recover subblocks b_{i1} and b_{i4}. As for subblocks b_{i2} and b_{i3}, they use the side-match VQ method to reconstruct them. Assume the four neighbors of subblock b_{ij} are b_l, b_r, b_u, and b_d. Then the side-match method can use these four neighbors to reconstruct subblock b_{ij}. When the side-match method is adopted, the bit length of the recovery features can be reduced. Let the reconstructed block of $\widetilde{B}_i$ be $B_i^{\prime}$. Finally, we can obtain a recovered image $\tilde{O}$.

The main drawback of this method is that it cannot tolerate inevitable innocent modification such as JPEG lossy compression. Therefore, we intend to propose a new method to improve it as follows.

3 The Proposed Method

Our proposed method is based on two properties of the JPEG lossy compression. We shall define the compression-tolerant range for the significant DCT coefficients and calculate the maximum difference for each block. The compression-tolerant range and the maximum difference are employed respectively to withstand JPEG lossy compression and to prevent the image from being tampered with. The signing and verification procedures of our proposed method will be stated as follows.

3.1 The Signing Procedure

Given a gray-level image O. First, we partition it into nonoverlapping blocks, and then we use DCT to transform each block into a DCT coefficient matrix $C(i, j)$, where $0 \leqq i, j \leqq 7$. The DCT coefficients located in the upper left positions contain most of the information of the image block, even if the image has been compressed by lossy JPEG. This is the reason why we often call them the "significant DCT coefficients". We extract some features from the significant DCT coefficients of each block for further verification. If the block is modified, then the significant DCT coefficients will also be changed. The extracted features will not be identical with the original ones.

To withstand JPEG lossy compression, we set the compression-tolerant range for those significant DCT coefficients. The detailed procedure is described as follows. Consider an image block B_i with its DCT coefficient matrix C. The proposed method chooses ten significant DCT coefficinets C(0,0), C(0,1), C(1,0), C(2,0), C(1,1), C(0,2), C(0,3), C(1,2), C(2,1), and C(3,0) in zig-zag scan order (like JPEG). We apply these DCT coefficinets to represent the features of the block. Besides, we use a scale function to quantize and adjust these DCT coefficients. The scale function is defined as below.

$$C''(i,j)=\begin{cases}\lfloor C(i,j)/\alpha \rfloor\times\alpha+\dfrac{\alpha}{2} & \text{if } C(i,j)\geq 0 \\ \lfloor C(i,j)/\alpha \rfloor\times\alpha-\dfrac{\alpha}{2} & \text{otherwise}\end{cases} , \qquad (5)$$

where $0 \leqq i, j \leqq 7$, and α is the position number. The parameter α is used to quantize the significant DCT coefficients, and the value (α / 2) is used to indicate the compression-tolerant range. A large α value will make the values of the DCT coefficients change drastically; as a result, the reconstructed image will become very distorted. On the other hand, if the value of α is too small, then the image can hardly tolerate JPEG lossy compression. We can obtain a new reconstructed DCT coefficient matrix $C''(i, j)$ by adopting the above scale fucntion. Finally, we use IDCT transformation to transform each adjusted DCT block into 8x8 pixels and thus obtain a signed image O'.

We could record the ten adjusted DCT coefficients directly and use them as the features of the corresponding block. However, this would waste a lot of storage space. The proposed method conducts the following steps to reduce the storage of the ten coefficients. First, we create a one-dimension array A whose size is 4096. Each element of A is a bit value. The maximum values of the DCT coefficients in a DCT block will not exceed the range between +2048 and -2048 after the DCT transformation. The proposed method assigns the contents of A by means of a pseudo random number generator (PRNG) with a seed S_k. Notice that we can use different seeds in different images for security's sake.

We will use the contents of A to record the ten adjusted DCT coefficients. If the value of an adjusted DCT coefficient is p, then the proposed method retreives the corresponding bit A[2048+p] to represent p. This bit is called the "feature bit." Hence, we collect the ten feature bits as a variable f for each block. Note that f is the final result of the proposed method according to the ten significant DCT coefficients.

How is the extraction of the features related to security? The features extracted by the above procedure do not contain any information about the DCT coefficients located in the lower right positions. What if someone maliciously manipulates the DCT coefficients located in the lower right positions, then? For example, assume that a DCT coefficient located in a lower right position is 0. If we modify this coefficient to be 1000, then the reconstructed block will be different from the original block. However, depending on features extraction only, we cannot solve this kind of problem. Because the above procedures only consider the significant DCT coefficients located in the upper left positions, we think it is necessary to add some processes to assist the features extraction to solve this kind of problem.

The lower right DCT coefficients of a block processed by the quantization table of JPEG will get close to zero or become zero. These DCT coefficients indicate the high frequency part of the block. Their zero values represent that the pixel values of the corresponding block usually have lower variance. That is, the pixel values of the block will be close to their mean value in most cases. Let m denote the mean value of a block B. The proposed method employs m to subtract each pixel value in B and picks out the maximum absolute different value as the maximum difference T of B. According to the results of our experiments, after JPEG lossy compression, for the same block B, its maximum difference T' shall not be bigger than two and a half times the original difference T.

Finally, we disclose the signed image O'. The proposed method stores a quantization value α, the seed S_k of PRNG, the maximum difference T of each block, and the feature f of each block in the certification authority (CA). The maximum difference and the features of each block will be used to verify the signed image O'.

[Signing Procedure]

Input: A gray-level image O, a seed S_k of PRNG, and a quantization value α.

Output: The signed image O', the maximun difference T of each block, and the feature f of each block.

Step 1: Partition an image O into nonoverlapping blocks and use DCT to transform each block into DCT coefficient matrix $C(i, j)$, where $0 \leqq i,j \leqq 7$.

Step 2: Use a scale function to quantize and adjust ten signification DCT coefficients located in the upper left positions.

Step 3: Create a one-dimension array A whose size is 4096. The content of A is assigned by PRNG with a seed S_k.

Step 4: If the adjusted DCT coefficient is p, then retrieve the corresponding bit A[2048+p]. Collect the ten featue bits and genertate the feature f for each block.

Step 5: Use IDCT trasnformation to transform each adjusted DCT block into 8×8 pixels and then obtain a signed image O'.

Step 6: Partition the signed image O' into nonoverlapping blocks and calculate the maximum difference T for each block.

Step 7: Disclose the signed image O' and store the quantization value α, a seed S_k of PRNG, the features f of each block, and the maximum different T of each block in the certification authority (CA).

An example of the signing procedure is illustrated in Figure 1. Figure 1(a) shows a block of 8×8 pixels. We use DCT to transform this block into DCT coefficients and then list them in Figure 1(b). After that, we use the scale function to adjust ten of the significant DCT coefficients. The quantization value α in the scale function of this example is 8. The adjusted DCT coefficients are shown in Figure 1(c). The ten adjusted significant DCT coefficients are 892, -132, 84, -12, -164, 44, -4, 84, 148, and 132. Next, we create a one-dimension array A. The content of A, listed in Figure 1(d), is assigned by PRNG with the seed S_k. The extracted ten feature bits are 1, 1, 1, 0, 1, 0, 0, 0, 0, and 0 since A[2048+892]=1, A[2048+(-132)]=1, ..., A[2048+(132)]=0. After the above processes, the proposed method uses IDCT to transform the adjusted DCT coefficients block into 8×8 pixels. The signed block is shown in Figure 1(e). The mean value and maximun difference in this example are 111 and 98 respectively. Finally, we store the quantization value α (i.e., 8), the seed S_k of the PRNG, the maximum difference (i.e., 98), and the feature bits (1, 1, 0, 1, 0, 0, 0, 0, 0, 0) in CA.

3.2 The Verification Procedure

The basic idea of the verification procedure is that we will extract the maximum differences and features of the signed image and compare them with the corresponding maximum differences and features stored in CA. The image is tampered with if they are not identical.

At the beginning, we input a signed image O'. The proposed method requests the quantization value α, the seed S_k of the PRNG, the feature f of each block, and the maximum difference T of each block from CA. Next, the signed image O' is partitioned into nonoverlapping blocks where the block size is 8×8 pixels. Consider a block B_i. We extract the maximum difference T' of B_i. The maximum difference calculation method here is identical with that in the signing procedure. The block will be proven to have been tampered with if the maximum difference T' is larger than 2.5×T. Here T is the maximum differnece of B stored in CA.

Suppose $T' \leq 2.5 \times T$. We cannot say for sure that this block has not been tampered with for now. The proposed method has to do the following checking. The proposed method uses DCT to transform B_i into DCT coefficients. The feature f' in B_i will be extracted. The feature extraction procedure in the verification procedure is the same as that in the signing procedure. The block B_i can be proven to have not been tampered with if the extracted feature f' of the signing image is identical with the corresponding feature f stored in CA. After the checking is all done, the blocks tampered with are marked "Yes" and the blocks not tampered with are marked "No".

[Verification Procedure]

Input: The signed image O'. The quantization value α, the seed S_k of the PRNG, the feature f of each block,and the maximum different T of each block from CA.

Output: 'Yes' or 'No'. 'Yes' means the block has been tampered with. 'No' means the block has not been tampered with.

Step 1: Partition an image O' into nonoverlapping blocks.

Step 2: Extract the maximum difference T' for each block.

Step 3: Compare T with T' in the same block. The block is proven to have been tampered with if T' is larger than 2.5×T. If so, stop verifying the block and mark "Yes" for this block. Otherwise, we cannot yet say that this block has not been tampered with for now. The proposed method has to do the following checking.

Step 4: Extract the feature f' in the remaining blocks.

Step 5: Compare f' with f in the same block. If they are identical, the block is marked "No". Otherwise, the block is marked "Yes".

An example of the verification procedure is illustrated in Figure 2. Figure 2(a) shows a violence block of 8×8 pixels to be verified. The mean value and maximum difference in this example are 68 and 129, respectively. The maximum differences in Figures 1 and 2 are 98 and 129, respectively, and the maximum value 129 is bigger than 98. The distortion here is acceptable in terms of the proposed method. Thus we need to verify this block again in the following. The proposed method applies DCT on this block. The DCT coefficients of this block are listed in Figure 2(b). After that, we use the same scale function as was used in the previous example to adjust the ten significant DCT coefficients. The quantization value α is 8 in the scale function. The quantized results are 548, -44, 28, 12, -284, 44, -20, 356, 20, and 220. Next, we create a one-dimension array A by PRNG with the seed S_k and list it in Figure 1(d). The extracted ten feature bits are 1, 1, 1, 0, 1, 0, 0, 1, 0, and 1 since A[2048+548]=1, A[2048+(-44)]=1,..., and A[2048+220]=1. The proposed method picks up the original feature from the previous example and compares it with the above results. Finally, we find that they are not identical since $(1, 1, 0, 1, 0, 0, 0, 0, 0, 0) \neq (1, 1, 1, 0, 1, 0, 0, 1, 0, 1)$. This means this block has been tamperd with.

4 Discussions and Analyses

The number of the significant DCT coefficients in our experiment is set to be 10. We select ten significant DCT coefficients to represent the feature of a block because they are enough to keep most of the information of a block according to our experiments. If we use more feature bits, then the security of the proposed method can be improved significantly, but the image quality of the signed image will be decreased at the same time. This is because most of the significant DCT coefficients have been modified and adjusted by the scale function. In addition, we need to store more feature bits of each block in CA. On the other hand, if we use less significant DCT coefficients in the proposed method, then the image quality of the signed image will be improved. However, the coefficients will be not enough to represent the feature of a block and thus will increase the probability of undetected tampering.

The scale function uses quantization value α to quantize and adjust the significant DCT coefficients. If we increase the value of α, the extracted features will be able to survive under high JPEG lossy compression. That is, the compression-tolerant range will be increased. However, the image quality of the signed image will be decreased at the same time. This is because the increase of quantization value α will enlarge the variety of the DCT coefficients. On the other hand, if we decrease quantization value α, then the image quality of the signed image will be increased, but the compression-tolerant range will be decreased.

The feature bits and the maximum differences in this paper are used to verify the signed image. The number of the feature bits is the same as that of the significant DCT coefficients. If we extract more feature bits, then the security of the image can be improved, but we need to store more features. Besides, the selection of the maximum difference of each block is also very important. A large difference will increase the pixel's compression-tolerant range, but a small difference can decrease the probability of undetected tampering.

Consider the following cases of malicious manipulations. First, if someone wants to use image-processing tools to slightly modify the image directly, we can consider this kind of modification acceptable like reasonable JPEG lossy compression. Otherwise, we can use the features to point out the modified areas. Second, if someone tries to remove the features, it will be very difficult. The security of our method is based on PRNG and the features stored in certificate authority (CA) for further verification. Without the seed S_k of PRNG, nobody can get the features. Third, if someone wants to maliciously manipulate the non-significant DCT coefficients like setting a large number on those non-significant DCT coefficients, we can use the maximum difference to point this modification out.

Our experiments were executed on IBM personal computer with a Pentium 133 CPU. The image-processing tool employed in the experiments was Photoshop 5.0. Two test images 'Lena' and 'F16' were used. Their original images are shown in Figures 3 and 4. Their image size is 256×256 pixels each. Table 1 lists the PSNR values versus α under some quantization values. Table 2 shows the error block numbers of two signed images under some JPEG compression quality. In Photoshop, the parameter q represents compersssion quality, and its value is between zero to ten. A large q value means better image quality but also means low compression rate, and vice versa.

The peak signal to noise rate (PSNR) is used to evalute the image quality, and it is defined as follows:

$$\text{PSNR}=10\times\log_{10}\frac{255^2}{MSE}\ \text{dB}. \tag{6}$$

The mean square error (MSE) for an $N\times N$ gray-level image is

$$MSE=\left(\frac{1}{N}\right)^2\sum_{i=1}^{N}\sum_{j=1}^{N}(x_{ij}-\bar{x}_{ij})^2. \tag{7}$$

Here x_{ij} denotes an original pixel value, and $\bar{x}_{ij}$ denotes the corresponding decoded pixel value. Besides, we define the compression rate (CR). The compression rate is defined as follows:

$$\text{CR}=\frac{X}{\tilde{X}}. \tag{8}$$

Here X means the original image size while $\tilde{X}$ means the compressed image size.

Two extra experiments were conducted to illustrate the performance of our method except the above experiments. The quantization vlaue α was set to be 8 in these two extra experiments. In the first experiemt, we input the original image 'Lena' shown in Figure 3 and output the signed gray-level image 'Lena' in Figure 5. After that, we used JPEG lossy compression to compress the signed 'Lena', where the compression rate was 3 (CR=3), and, moreover, we modified the compressed 'Lena' in the eyes. The decompressed 'Lena' after compression and tampering is shown in Figure 6. The result of the verification procedure is listed in Figure 7. According to our experiment, the decompression result is "acceptable"; and the eyes tampered with can also be pointed out by our method. Such phenomenon also exists in 'F16'. The experimental results are shown in Figures 8-10.

5 Conclusions

In this paper, we have proposed two important properities of image authentication. Taking advantage of those two properities, our method can both prevent images from being tampered with and allow acceptable JPEG lossy compression. First, we set the compression-tolerant range for significant DCT coefficients. The block can withstand JPEG lossy compression if the significant DCT coefficients after JPEG lossy compression do not fall off the compression tolerant-range. Second, we set the maximum difference for each block to detect malicious manipulations done to the non-significant DCT coefficients. According to our experimental results, our method can indeed withstand JPEG lossy compression while keeping the image from being tampered with.

6 References

[1] Bhattacharjee S. & Kutter M. (1998) Compression Tolerant Image Authentication. *IEEE International Conference on Image Processing*, 1, p. 435-439.

[2] Friedman, G. L. (1993) The Trustworthy Digital Camera: Restoring Credibility to the Photographic Image. *IEEE Transactions on Consumer Electronics*, 39, 4, p. 905-910.

[3] Lu C. S. & Liao H. Y. (2000) Structural Digital Signature for Image Authentication: An Incidental Distortion Resistant Scheme. *Proceedings of the Workshops on ACM multimedia*, p. 115 – 118.

[4] Kundur D. & Hatzinakos D. (1999) Digital Watermarking for Telltale Tamper Proofing and Authentication. *Proceedings of the IEEE*, 87, 7, p. 1167-1180.

[5] Schneider M. & Chang S. F. (1996) A Robust Image Content Based Digital Signature for Image Authentication. *International Conference on Image Processing,* 3, p. 227-230.

[6] Lin E. T., Podilchuk C. I. & Delp, E. J. (2000) Detection of Image Alternations Using Semi-Fragile Watermarks, *Security and Watermarking of Multimedia Contents*, 3971, p.152-163.

[7] Hung K. L., Chang C. C. & Chen T. S. (2001) Secure Discrete Cosine Transform Based Technique for Recoverable Tamper Proofing. *Optical Engineering*, 40, 9, p. 1950-1958.

[8] Rivest R., Shamir A. & Adleman L. (1978) A Method for Obtaining Digital Signatures and Public-key Cryptosystems. *Communications of the ACM*, 21, 2, p. 120-126.

[9] Pennebaker W. B. & Mitchell J. L. (1993) JPEG: Still Image Data Compression Standard, New York, Van Nostrand Reinhold, 1993.

[10] Chang R. F. & Chen W. T. (1993) Side-match vector quantization for reconstruction of lost block. *Journal of Visual Communications and Image Representation*, 4, 2, P. 171-177.

Table 1: The image quality (PSNR) of the signed images under different quantization vlaues α=4, 6, 8, 10, and 12

	α=4	α=6	α=8	α=10	α=12
F-16	50.04dB	49.77dB	47.72dB	45.93dB	44.36dB
Lena	52.07dB	49.87dB	47.97dB	46.16dB	44.62dB

Table 2: Numbers of Error blocks of the signed images pointed out after different JPEG compression qualities *q*= 8, 5, and 1

Signed images	α=4	α=6	α=8	α=10	α=12
F-16 (*q*=8)	0	0	0	0	0
F-16 (*q*=5)	942	586	230	70	0
F-16 (*q*=1)	1024	981	853	820	640
Lena (*q*=8)	0	0	0	0	0
Lena (*q*=5)	960	597	225	74	0
Lena (*q*=1)	1024	1018	921	873	650

$$\begin{bmatrix} 157 & 121 & 130 & 136 & 122 & 128 & 147 & 130 \\ 85 & 79 & 69 & 69 & 54 & 55 & 57 & 54 \\ 58 & 45 & 46 & 46 & 41 & 67 & 105 & 67 \\ 52 & 72 & 61 & 61 & 56 & 92 & 197 & 177 \\ 54 & 96 & 98 & 98 & 68 & 130 & 210 & 208 \\ 98 & 92 & 116 & 116 & 136 & 186 & 198 & 188 \\ 142 & 128 & 115 & 115 & 127 & 136 & 134 & 154 \\ 156 & 151 & 144 & 144 & 139 & 141 & 144 & 162 \end{bmatrix}$$

(a) A block of 8×8 pixels

$$\begin{bmatrix} 895 & -132 & 85 & -12 & -27 & 38 & -14 & -1 \\ -167 & 41 & -4 & 4 & -2 & 24 & -11 & 5 \\ 83 & 148 & -46 & 31 & 45 & -19 & 24 & 0 \\ 134 & -41 & -9 & 2 & -2 & 2 & 15 & -3 \\ 100 & -40 & 32 & -35 & -7 & 12 & -18 & -1 \\ 29 & -10 & 14 & 16 & 8 & 4 & -4 & 11 \\ 40 & -24 & -18 & 21 & -6 & 14 & 9 & 6 \\ -14 & 15 & 2 & -6 & 2 & 3 & 0 & -6 \end{bmatrix}$$

(b) The DCT coefficients

$$\begin{bmatrix} 892 & -132 & 84 & -12 & -27 & 38 & -14 & -1 \\ -164 & 44 & -4 & 4 & -2 & 24 & -11 & 5 \\ 84 & 148 & -46 & 31 & 45 & -19 & 24 & 0 \\ 132 & -41 & -9 & 2 & -2 & 2 & 15 & -3 \\ 100 & -40 & 32 & -35 & -7 & 12 & -18 & -1 \\ 29 & -10 & 14 & 16 & 8 & 4 & -4 & 11 \\ 40 & -24 & -18 & 21 & -6 & 14 & 9 & 6 \\ -14 & 15 & 2 & -6 & 2 & 3 & 0 & -6 \end{bmatrix}$$

(c) The adjusted DCT coefficients

Coefficient	0	...	1764	...	1884	...	1916	...	2004
PRNG	1	...	1	...	0	...	1	...	1
Coefficient	2028	...	2036	...	2044	...	2060	...	2068
PRNG	0	...	1	...	0	...	0	...	0
Coefficient	2076	...	2092	...	2096	...	2132	...	2140
PRNG	1	...	0	...	1	...	0	...	0
Coefficient	2180	...	2196	...	2212	...	2268	...	2404
PRNG	0	...	0	...	1	...	1	...	1
Coefficient	2596	...	2940	...	3000	...	3500	...	4096
PRNG	1	...	1	...	0	...	0	...	1

(d) The content of an array A

$$\begin{bmatrix} 156 & 120 & 130 & 136 & 122 & 127 & 145 & 130 \\ 85 & 78 & 69 & 68 & 54 & 55 & 58 & 54 \\ 57 & 45 & 45 & 46 & 41 & 67 & 105 & 67 \\ 52 & 71 & 61 & 61 & 55 & 92 & 195 & 176 \\ 55 & 95 & 97 & 97 & 69 & 130 & 209 & 207 \\ 98 & 92 & 115 & 115 & 135 & 185 & 197 & 187 \\ 141 & 127 & 116 & 114 & 127 & 135 & 133 & 153 \\ 156 & 150 & 143 & 143 & 139 & 141 & 144 & 161 \end{bmatrix}$$

(e) The signed block

Figure 1: An example of the signing procedure

$$\begin{bmatrix} 157 & 121 & 130 & 136 & 122 & 128 & 147 & 130 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 54 & 96 & 98 & 98 & 68 & 130 & 210 & 208 \\ 98 & 92 & 116 & 116 & 136 & 186 & 198 & 188 \\ 142 & 128 & 115 & 115 & 127 & 136 & 134 & 154 \\ 156 & 151 & 144 & 144 & 139 & 141 & 144 & 162 \end{bmatrix}$$

(a) A violence block of 8×8 pixels

$$\begin{bmatrix} 550 & -42 & 26 & 10 & 0 & 8 & 7 & 0 \\ -281 & 40 & -16 & -2 & 3 & 14 & -6 & 2 \\ 359 & 20 & 18 & 2 & 11 & 6 & 0 & 0 \\ 216 & -52 & 7 & 12 & -7 & 17 & 5 & 2 \\ 8 & 46 & -1 & -5 & 8 & 6 & -4 & 1 \\ 132 & 4 & 10 & 7 & 6 & 3 & 0 & 0 \\ 164 & -41 & 0 & 2 & -9 & 13 & 3 & 1 \\ -66 & 41 & 1 & -1 & 9 & -4 & -3 & 0 \end{bmatrix}$$

(b) The DCT coefficients

$$\begin{bmatrix} 548 & -44 & 28 & 12 & 0 & 8 & 7 & 0 \\ -284 & 44 & -20 & -2 & 3 & 14 & -6 & 2 \\ 356 & 20 & 18 & 2 & 11 & 6 & 0 & 0 \\ 220 & -52 & 7 & 12 & -7 & 17 & 5 & 2 \\ 8 & 46 & -1 & -5 & 8 & 6 & -4 & 1 \\ 132 & 4 & 10 & 7 & 6 & 3 & 0 & 0 \\ 164 & -41 & 0 & 2 & -9 & 13 & 3 & 1 \\ -66 & 41 & 1 & -1 & 9 & -4 & -3 & 0 \end{bmatrix}$$

(c) The adjusted DCT coefficients

Figure 2: An example of the verification procedure

Figure 3: The original image 'Lena'

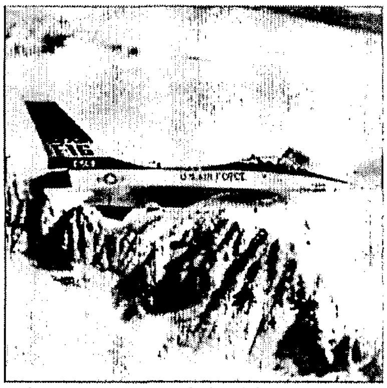


Figure 4: The original image 'F16'

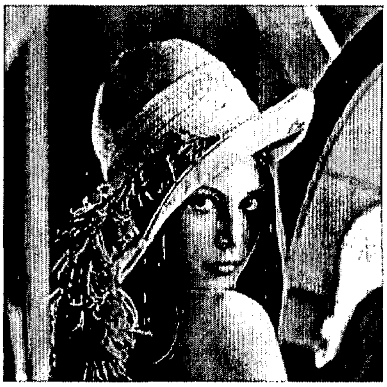


Figure 5: The signed image 'Lena' where the PSNR=47.97 dB

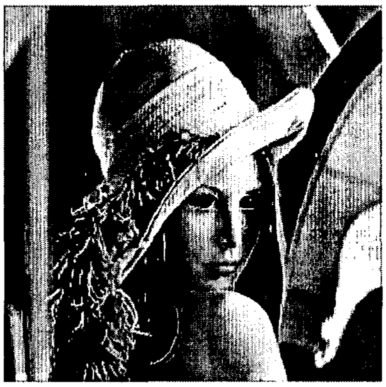


Figure 6: The compressed 'Lena' with the eyes tampered with

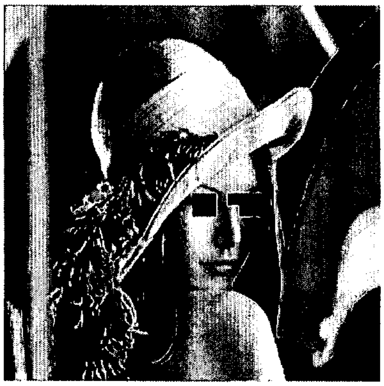


Figure 7: The verification result

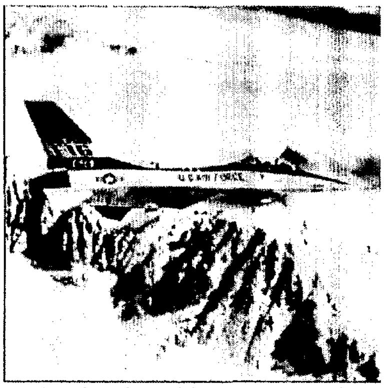


Figure 8: The signed image 'F16' where the PSNR=47.72 dB

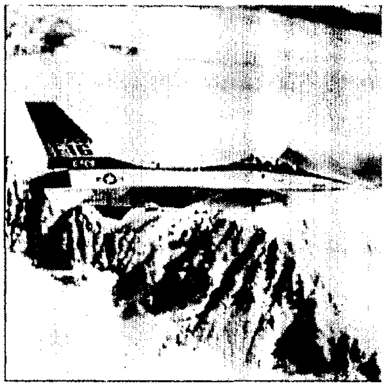


Figure 9: The 'F16' with airframe tampered with

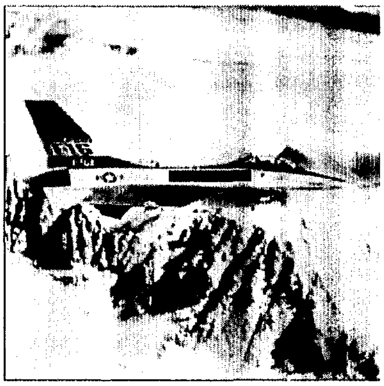


Figure 10: The verification result

Protecting the Data State of Mobile Agents by Using Bitmaps and XOR Operators

Jesús Arturo Pérez Díaz,
ITEMS, Campus Cuervanava
Av. Paseo de la Reforma 182-A, 62589, Temixco, Morelos, Mexico.
jesus.arturo.perez@itesm.mx
AND
Darío Álvarez Gutiérrez,
University of Oviedo
Calvo Sotelo s/n, 33007 Oviedo, Spain.
darioa@pinon.ccu.uniovi.es

Keywords: security, mobile agents, data state protection, encryption

Received: June 20, 2001

Mobile agents have been considered a promising technology to develop e-commerce applications, however the security concerns about the technology have stopped their widespread use.

The identified security areas comprise protecting hosts against malicious agents, protecting the agent's transmission and protecting agents against malicious hosts. The first two security issues and the protection of the agent's code state can be solved by applying traditional security techniques. Even though there are some works that manage the privacy of execution, their implementation is almost unfeasible in terms of performance and complexity.

This paper describes a fast and easy to implement algorithm that a mobile agent can use to encrypt its data during its itinerary. The algorithm only makes use of a bitmap and XOR operations. The algorithm consist of applying XOR operations to the data to be ciphered and a random bitmap, while the map is repeatedly shifted to the right or to the left in order to compute a CRC field for validation against malicious tampering.

The method only uses basic bit operations so that its implementation is very easy to develop. Besides, since it does not use any computationally expensive cryptographic technique (i.e. digital signatures) it is very fast. In this way we manage to have a secure, simple, fast and feasible protection algorithm to protect data while mobile agents are roaming, where simplicity and performance are its better advantages.

1 Introduction

Mobile Agent Systems are expected to make e-commerce transactions inside virtual supermarkets. In this application area security is crucial since we can consider that any application will not be useful without doing secure transactions.

Mobile agents consist of code state, data state, and execution state. Mobile agent systems are platforms that allow agents to migrate from one node (a mobile agent system) to another, keeping its three states. While agents migrate there are several security aspects involved. We can point out different mechanisms that must be implemented by the mobile agent system to ensure the security of the mobile agent applications. Mobile agent systems basically must provide:

- Protection of the agent system against attacks from mobile agents.
- Protection of the agent against other agents.
- Protection of information transmission between agent servers against unauthorized third parties.
- Protection of the agent against malicious agent systems (malicious hosts), which includes protection of the data state of the agent.

Different security architectures for mobile agents [1] and mobile agent systems [2] [3] have used standard cryptographic techniques like public key cryptography, or digital signatures to authenticate authorities and solve the problem of protecting the host against malicious agents. Also, they have implemented secure channels for the transmission of the agents by using SSL or TLS.

Nevertheless, the protection of mobile agents from malicious hosts is only partially solved.

The code state of the agent can be signed since it will not be modifiable. In this way, we can protect the static part of the agent. However, to protect the data state (that changes dynamically) becomes a more difficult task to tackle.

There are some works in this area, described in the related work section. However, there has not been found any solution having a feasible implementation.

Consider that most of mobile agent e-commerce applications do not need to protect all the data state, but only some important values where agents filter information and compile their results. The algorithm presented in this paper protects all the data that the agent decides to encrypt by calling a cipher function. When the agent returns, only the source server is able to decrypt the sensible information stored by the agent.

The algorithm describes an easy way to protect sensible data that must be gathered and are carried by mobile agents alongside their itinerary. Two principal advantages are highlighted: the algorithm is simple and feasible to implement, and computationally inexpensive.

2 Related Work

Wilhelm presented a technique for protecting the itinerary of the mobile agents by using hardware mechanisms [4]. He considered that software algorithms were not enough to ensure complete security during the mobile agent's itinerary. Even though the technique managed to achieve the protection of the itinerary, its implementation in real applications becomes difficult, since special hardware is required.

One interesting approach to avoid the malicious host attacks was proposed by Fritz Hohl [5]. This approach, which is called *Code Mess Up*, consists of a combination of two mechanisms: the first one generates a new and far less understandable version of the agent's code. The second mechanism restricts the lifetime of the agent's code and data. In this way, when the code of the agent is messed up, the malicious server would take some more time in order to understand the code and then attack it, but since the agent's lifetime is restricted, the malicious server will not have enough time to attack the agent. In this way the agent remains untouched.

Another solution for this problem was proposed by Tomas Sander and Christian Tschudin [6][7]. They presented techniques on how to achieve "non-interactive computing with encrypted programs" in certain cases and give a complete solution for this problem in important instances. They further show how an agent might securely perform a cryptographic primitive, digital signing, in an untrusted execution environment. Their results are based on the use of homomorphic encryption schemes and function composition techniques.

The last two solutions were designed to offer privacy on the agent's execution, but not to give privacy and integrity to the agent's data. Beside, both of them have two main problems: a quite difficult implementation and a considerable performance hit in case of implementation. Perhaps these disadvantages are the reason why these techniques have not been implemented by any mobile agent system, as far as we know, and the problem in current mobile agent systems is still unsolved.

Considering these related works, the goal of our research is to offer a simple and feasible to implement algorithm that can be used by mobile agents just for encrypting the data that they gather while they are roaming in untrusted execution environments, and without a perceivable performance hit.

3 Data Encryption Using Bitmaps and the XOR Operation

The design of this data protection technique takes into account the fact that, in most applications, it is not vital to protect the whole data state of the mobile agent but some variables holding sensible data gathered by the agent, which is the main goal of the agent's travel and need to be protected.

Typical examples of these applications are e-commerce applications in which an agent travels alongside an itinerary looking for prices or particular services. The vital information that must be protected is the price or service offered in each visited server.

This technique requires that the agent travels holding data generated by the source server that will be used by the agent to encrypt the sensible data gathered, using fast XOR operations.

3.1 Usefulness of the XOR Operator

The main encryption idea is to apply the XOR operation between data and a random number (expressed as a bitmap in a row of a matrix and known only by the source server) to encrypt information. Once the agent returns to the source server, the XOR operation is applied again to the encrypted data, using the same random number, and the information is restored.

The agent in the source server will generate two matrixes with a number of rows equal to the number of data items it expects to encrypt. Initially, both matrixes will be filled with the same random numbers (forming a random background bitmap). One of the matrixes will be stored in the source server and the agent will carry the other.

For example, let's assume a 10 (binary 1010) is generated as a random number and put it in a row of the matrix. This number goes with the agent and a copy is also stored in the source server. During the itinerary, the agent gets a 3 (binary 0011) that the agent wishes to encrypt. The XOR operation will be then applied between the random number and the datum to be protected (1010 ⊕ 0011), giving 1001 as a result. This datum is stored in the same row of the matrix, overwriting the initial random mask, in order to avoid the next server seeing the random number used to encrypt the datum. Also, the next server is not able to know the real datum, since it ignores the random number used to apply the XOR operation.

A given server will use the next free row available in the matrix to store new data, as the occupied rows contain data encrypted in previous servers. In this way, the current server will never be able to know the previously encrypted data since it does know neither the datum nor the random number.

The source server had stored in a duplicate matrix a copy of the random numbers, in order to retrieve when the agent returns, the data encrypted by the agent while roaming from server to server. Thus, to retrieve the datum, the encrypted number 1001 will get a XOR applied with the corresponding random number generated in the source server (1001 XOR 1010), giving 0011 as a result (3, the datum the agent had encrypted).

To assure that the information restored upon return to the source server has not been tampered with, and is the same information that the agent encrypted in each server, a CRC field is computed in order to perform an integrity test.

The complete encryption algorithm is described in the next section.

3.2 Detailed Description of the Encryption Algorithm

A matrix with several fields is defined (table 1), which is used to encrypt the agent's data and for validation of the data later. The matrix is initially filled with random numbers, creating a background bitmap used to encrypt data gathered by the agent alongside its itinerary (table 2). The matrix is duplicated. One copy travels with the agent and the other is kept in the source server. The source server's matrix is used to recover the data upon agent's return.

Every datum to be protected by the agent needs a row of the matrix, so the agent must know beforehand the approximate amount of data it is going to use.

The structure of the matrix is as follows:

Data Area of the Mobile Agent						
Host ID	**Data to be protected**				**CW**	**CRC**
128 bits	128 bits	128 bits	....	128 bits	128 bits	128 bits
f1	*f2*	*f3*		*fn*		

Table 1. Fields composing the rows of the matrix

The first field is the identifier of the server, as we need to know, for each row, the place where data was encrypted. The second field represents the space needed to store the data to be gathered by the agent, in 128 bit blocks. The third field is the "codeword", which is a random number to be generated in the remote server. The codeword is used to rotate data before applying the encryption function. The last field is a CRC, which is computed applying a XOR operation using all the 128 bit blocks in the data area. This CRC is used upon agent return to verify that the data area has not been altered.

Data Area of the Mobile Agent						
ID Host	**Data to be protected**				**CW**	**CRC**
128 bits	128 bits	128 bits	....	128 bits	128 bits	128 bits
0101	1101	0011	010	0101	0100	1101
1100	0001	1101	010	0001	0100	0100
1011	0011	0101	001	1101	1001	0100

Table 2. Matrix filled with a random generated bitmap

When the agent departs from the source server, the agent carries the matrix filled with random numbers, creating a background bitmap that is used to hide information, as shown in the next table:

Alongside the itinerary, the following algorithm is applied for each datum to be encrypted:

1. The remote server creates a record with the same fields than a row of the matrix that the agent has.
2. The host ID, data to be encrypted in 128-bit blocks form, and a generated random codeword (CW) are put into the record.
3. Each 128 bit block fi is rotated to the left as many times as indicated by the 7 less-significant bits of the CW. That is $fi \leftarrow fi << li$, where $li \leftarrow CW \, \& \, 07Fh$.
4. Before applying the third step on $fi+1$, the CW is rotated to the right as many times as indicated by the 7 most-significant bits of the CW. Thus, the number of times that each fi is rotated is not always the same. CW will then be $CW \leftarrow CW >> mi$ where $mi \leftarrow (CW << 7) \, \& \, 07Fh$. Once the CW is rotated step 3 is repeated. These tasks will continue until no more 128-bit blocks are left.
5. The original CW is restored into the corresponding field of the register in order to retrieve the original information using the inverse algorithm in the source server.
6. The CRC field is computed as follows. The initial value is filled with binary 0's, and then it is XOR'ed in sequence (from left to right) with all the 128 bit blocks in the data area, giving the final CRC value.

7. Lastly, the corresponding row in the matrix holding the original bitmap is XOR'ed with the generated register (data) to be protected, thus encrypting the data.
8. The counter indicating the number of lines used in the matrix is incremented so that the next row available of the matrix can be used.

It is worth to note that, once the mobile agents arrive at a new server, the new server can not access the information stored in the previous server, as the background bitmap held in the previous row before the XOR operation was applied can not be guessed[1]. Only the source server, who has a copy of the original matrix, is able to apply the inverse algorithm to retrieve the encrypted data.

The CRC field is computed in order to detect any alteration made by a malicious server on the encrypted data. Without any validation action, a malicious server could modify just one bit on the encrypted data field and when the agent returns to its source server it would not be able to detect that alteration. So, the agent would recover a wrong modified value since the source server just applies and XOR operation with the stored copy of the matrix in order to get the encrypted value.

The CRC field does not prevent a malicious server from making any alteration, but it ensures that if an alteration were made it would be detected since the CRC field will be invalid.

The bit rotations made in step 3 may appear unnecessary. However, if the blocks are not rotated, a malicious server could alter the encrypted information in just one bit in a specific position, and the CRC may not change since each block is XORred with the next block. Then, upon return to the source server, this alteration would not be detected.

On the other hand, once the random rotations are applied, a maliciously-altered bit in one block would be detected, as this bit affects many positions in the inverse decryption algorithm (since the position of that bit will change after rotations are applied), rendering always an invalid CRC that will detect the alteration.

To retrieve the information encrypted by the agent alongside the itinerary, the source server just applies the XOR operation to each row of the matrix that was used by the agent, with the corresponding row of the copy of the original matrix holding the initial background random bitmap.

Then, using the random CW, inverse rotations are applied to retrieve the real data that was encrypted by the agent in a given intermediate server.

The main advantage of this technique, encrypting data using bitmaps and XOR operations is that is very easy to implement, compared with other methods, which use very complex mathematical algorithms [8] Besides, it is computationally inexpensive, as only very fast bit operations are used, avoiding effectively the performance impact of other techniques such as digital signatures, keys, or any other means that hurt performance.

4 Feasibility of Implementation and Incorporation to Current Mobile Agent Systems

A great advantage of our protection scheme is the feasibility of implementation. Besides, it could be very easily incorporated to the current mobile agent systems' security mechanisms.

The majority of Java-based mobile agent systems define an abstract class called Agent. All the agents programmed by the user inherit from this class the required functionality, so that the agent can migrates from one host to another or can create more agents.

This abstract class usually follows a pattern like this:

```
public abstract class Agent implements java.io.Serializable{
  public void run()
  public final java.lang.Object clone()
  public final void createAgent(.....)
  public final void dispatch(java.net.destinationURL)
  public final void revert()
    ...
}
```

We just need to add an addsecure() method to the agent abstract class in order to allow the agents to securely store sensible information in the data structure(the matrix of bitmaps) that is carried with them, so we could define:

```
        public final void addsecure(Object data)
```

The implementation of this method will encrypt the information using the algorithm described in the previous section and will store it in the next row available of the matrix that is carried with the agent.

The matrix can be easily defined in Java using a Java array (i.e. an instance of the class vector) that will hold the background bit map originated in the source server.

In this way, each time that an agent is created by a user (i.e. commerceAgent), it will inherit the addsecure() method allowing it store information in a secure way.

```
        public class commerceAgent extends Agent {
        .....
        }
```

When the user creates an instance of commerceAgent, the instance will be able to protect the information it is gathering, just by invoking the addsecure() method. For example:

```
        commerceAgent findFlyAgent;
```

[1] Only the next available rows, not used yet, can be seen. This can be used for an attack that is described later.

defines an agent of type commerceAgent. The run method of the findFlyAgent would contain the instructions to query the price of the fly it is looking for, once it gets the price it records for later analysis at the home server, so the last line of the run method would be:

findFlyAgent.addsecure(FlyPrice);

in order to protect the sensible datum it has gotten. The current server will execute the encrypting process and will store the FlyPrice safely in the matrix. The next server visited will not be able to find out what was the price in the previous one and it just will be able to encrypt the information that the agent gathers in that server.

5 Limitations of the Method

The algorithm allows to protect the information the agent decides during its itinerary, and to verify that it has not been altered when the agent returns. The algorithm does not prevent the possible alteration of data from malicious hosts, but detects any modification that has been made. In this way, if any alteration is detected (which means a CRC field is invalid) the agent will reject the information since it would be considered invalid. In this way our technique offers integrity.

The current server will never be able to access the previously encrypted data since it ignores the data and the random number used to apply the XOR operation. However, it can see and copy the still available rows with random numbers that will be used to encrypt the next data not only in the current server but in the next server as well.

The first, and most evident deriving of this, is that a visited server cannot retrieve the data that was encrypted before, but could easily make a copy of the rest of the background bitmap. This means that a server could potentially retrieve the data encrypted in the future by an agent, assuming that the agent visits again the same server. Thus, an agent should not visit the same server twice if it wants to be completely secured.

Another possible attack (although less probable) is that two cooperating malicious servers teamed to retrieve the information carried by the agent. The first server would send to the second one a copy of the unused part of the background bitmap already known by the first server (the available rows of the matrix). If the agent arrived later to the second malicious server, it would be able to retrieve the data encrypted since the agent left the first malicious server and then modify the values.

The last limitation is that there is a fixed maximum number of data that can be protected, which is given by the length of the matrix (the length must be set in advance). However, in practice, a reasonable length could be set, according with the expected task to be carried by the agent.

Finally, this technique does only protect the part of the data state of the agent that the agent wishes to encrypt. The rest of the data, such as local variables, etc. are not protected.

6 Future Work

We will continue working on this technique in order to implement an improved algorithm that avoids the current limitations. Nevertheless we intent to use only the operations included in this paper (bit rotations and XOR operation), or equally fast or simple ones, so that we can keep its simplicity and fast speed which are the objectives and philosophy of this work.

7 Conclusions

One of the problems that a mobile agent system must solve is the protection of agents from malicious hosts, which includes the protection of the data state of an agent. This is very important in order agent technology be adopted in e-commerce applications, for example in applications where agents collect information (such as flight prices) for later analysis at the source server.

Protecting this data gathered by the agent (and not the whole data state, which is not vital) is the objective of the research described in this paper. For example, malicious servers should not be able to see or modify the information gathered in order to change previous low prices to make its price appear as the best.

Other techniques such as [5] and [6] try to solve the problem by privacy of execution applying very complex techniques, which are very difficult to implement, and, more importantly, are very expensive computationally, as key cryptography is used. This is a hurdle very difficult to overcome in practical systems.

We propose a new technique to protect the part of the data state of an agent (the data gathered the agent wishes to protect) that dose not suffer from these limitations, as it is fast and easy to implement.

A matrix is generated at the source server, and filled initially with random bit numbers. Each row is used to protect one datum, and is divided into 128 bit blocks. A copy of the matrix is stored at the source server, and the other copy travels with the agent.

To protect one data item, the agent uses a row, and applies the XOR operation to the data item with the random number held previously, encrypting it and overwriting the initial random number with the result. A CRC is computed using XOR operations also, in order to detect alterations when returning to the source server. Some bit alterations would not change the CRC, so the random bitmaps are bit-rotated n-times, as indicated by a random codeword that is also held in the matrix (and is also rotated).

The original server is the only one able to decrypt the information, since the inverse algorithm (basically undoing the rotations and applying the XOR operation with the original random bit map) requires the knowledge of the original random bit map and codewords, which is only know (the bitmaps) by the original server.

The only limitation is that an agent should not visit the same server twice, or a server co-allied with a malicious server, as a copy of the matrix could be made and subsequent encrypted data items could be retrieved. A minor limitation is that the agent should estimate the maximum number of data items to protect, as the matrix must be generated beforehand.

The technique we have presented removes the complexity and computational limitations of other techniques, which hinder the acceptance of agent technology in real applications. Agent's data state protection is made feasible in practical applications, as no performance hit is introduced because no expensive key cryptography is used. Furthermore, the algorithm could be a lot of times faster than any other that uses traditional key cryptographic techniques since only bits operations are used.

This algorithm can be easily integrated in current mobile agent systems in order to create basic e-commerce applications that compile information securely.

References

[1] Pérez Díaz Jesús Arturo, Álvarez Gutiérrez Darío. *Sahara: a comprehensive security architecture for mobile agents systems.* Simposio Español de Informática Distribuida. ISBN: 84-8158-163-1. Orense, Spain.

[2] Pérez Díaz Jesús Arturo, Álvarez Gutiérrez Darío, Cueva Lovelle Juan Manuel. *An implementation of a secure Java2-based mobile agent system.* The Fifth International Conference on The Practical Application of Intelligent Agents and Multi-Agent Technology. *PAAM 2000.* ISBN: 1 902426 07 X. Manchester, U.K.

[3] Fraunhofer IGD. Project http://www.informatik.uni-stuttgart.de/ipvr/vs/projekte/mole/mal/preview/SeMoA-(Secure-Mobile-Agents).9656.txt.html

[4] Wilhelm Uwe G., Staamann Sebastian. *Protecting the Itinerary of Mobile Agents.* Anales del ECOOP Workshop on Distributed Object Security and 4th Workshop on Mobile Object Systems. Julio 21-22 1998. Belgium.

[5] Hohl Fritz. *An approach to solve the problem of malicious hosts in mobile agent systems.* Institute of parallel and distributed systems, University of Stuttgart, Germany. 1997.

[6] Sander Thomas, Tshudin Christian. *Protecting Mobile Agents against Malicious Host.* Lecture Notes in Computer Science (LNCS), Springer-Verlag, New York, USA, 1419, June 1998.

[7] Sander Thomas, Tshudin Christian. *Towards mobile cryptography.* International Computer Science Institute (ICSI) Technical Report, 97(049):1-14. November, 1997.

[8] Joan Feigenbaum, Peter Lee. *Trust Management and Proof carriying code in Secure Mobile-Code Applications.* Accepted paper to the DARPA Workshop on Foundations for Secure Mobile Code Workshop, 26 - 28 March 1997.

[9] Sobrado Igor. *Evaluation of two security schemes form mobile agents.* Proc. ACM SIGCOMM – LatinAmerica and Caribbean 4/01, San Jose, Costa Rica, April 3-5 2001.

Evaluation of Technologies for Business Process Automation

Maja Pušnik, Matjaž B. Jurič and Ivan Rozman
University of Maribor, Faculty of Electrical Engineering, Computer and Information Science,
Institute of Informatics,
Smetanova 17, 2000 Maribor
E-mail: maja.pusnik@uni-mb.si

Keywords: business process automation technologies, decision models, ebXML, XLANG, RosettaNet

Received: June 2, 2002

The importance of process automation for B2B (business to business) collaboration is rising. The efforts are directed towards automating business processes and forming a global electronic market. In this paper we present and evaluate the three most important technologies for business process automation: ebXML (Electronic Business XML- eXtensible Markup Language), RosettaNet and XLANG. They differ in terms of features, quality and serviceability. We analyze, compare and evaluate those technologies from the perspective of SME (small and medium enterprises). Based on the comparison we define a multi-criteria decision model with twenty parameters and the corresponding weights, we evaluate the alternatives and define a utility function, which helps us to select the most suitable technology. The contributions of this paper are the in-depth evaluation of technologies and the definition of a multi-criteria decision model.

1 Introduction

The well-known fact is that business must be altered to survive the upcoming changes and progress. To make the idea of a global marketplace and B2B work, proper technologies, which will assure safety and efficiency, must be created. They have to be appropriate for all kinds of enterprises, small and large, for those with great financial recourses and responsibilities and for those with limited budgets. Only with such universal technologies, a global market and complete serviceability will be realized.

In the paper we will review and compare the three most important technologies for business process automation: ebXML, XLANG and RosettaNet. We will define criteria for their evaluation and build a decision model with twenty criteria. We will evaluate the results and choose the most suitable technology from the perspective of a SME. All three technologies are based on XML and build on the functionality of web services, where they reuse existing web service technologies, such as SOAP (Simple Object Access Protocol), UDDI (Universal Description, Discovery and Integration) and WSDL (Web Service Definition Language). We will see that they differ in some features while in others they are complementary. Because they are based on open standards, they are reachable in aspects of price and complexity, not only to large enterprises, but also to small and medium enterprises.

The review of related research has shown that there are not many similar analyses. The comparison made in [6] only compares ebXML and RosettaNet in an informal way and does not define a decision model. The author in [20] compares B2B standards, which include RosettaNet, ebXML, OAGIS (Open Applications Group Integration Specification) and Simple Web Services. The same author explains in a different article [21] how RosettaNet, ebXML, OAGIS and EDI (Electronic Data Interchange) fit together. However the author does not define a formal decision model. In [22] the author again compares RosettaNet, ebXML, OAGIS, Web Services, xCBL (UBL) – XML Common Business Library (Universal Business Language) and cXML (commerce XML) and creates a comparison framework.

Our paper is organized in the following order: the needs of the market are evaluated in the second chapter. Third chapter makes a comparison of the ebXML, RosettaNet and XLANG. Fourth chapter defines a multi-criteria decision model and evaluates them. The last, fifth chapter, gives a conclusion of the results.

2 Needs of the Market

The way enterprises work, understand their existence and survive must be retained. But the way they do business and communicate with each other must be improved. So business processes must still work on and through the net, just as they have manually. Technologies must describe business processes in a consistent and safe way and more. They must enable changes, upgrades and adaptations, since business is a living process and therefore must be flexible and manageable.

But this is only the first step. There is still the question of automation. A business process consists of many steps and includes many people, some of them completely unnecessary, which only enlarges the possibility of making a mistake. One of the goals in creating a global electronic market is to automate everything that can be automated, including routine work or explicitly defined processes with long-term rules and foreseen conditions.

Some solutions have already been created in the past, more or less successfully, but by far not sufficient enough for goals and ambitions of the millennium. The web services have only created an initiation of what is yet to come. They enabled process describing, but not automation. The ultimate goal of those technologies is making business as safe and as accessible as possible for all businesses all over the world [19].

3 Comparison of Technologies

There are several technologies for coordination and automation of business processes. Some of them have been present on the market for quite a long time, for example EDI. But the problem with vintage ones is inaccessibility for smaller enterprises and an obvious inflexibility, since most of them require a large initial investment and expensive support. The up to date technologies build upon legacy technologies, which have used older proprietary standards.

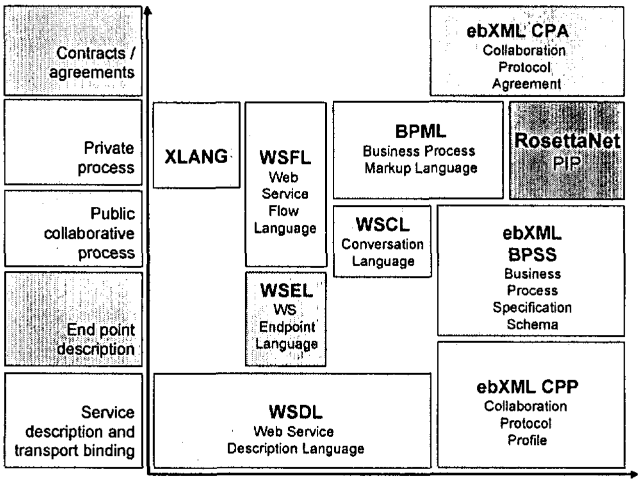

Contracts / agreements
Private process
Public collaborative process
End point description
Service description and transport binding
ebXML CPA
Collaboration Protocol Agreement
XLANG
WSFL
Web Service Flow Language
BPML
Business Process Markup Language
RosettaNet PIP
WSCL
Conversation Language
ebXML BPSS
Business Process Specification Schema
WSEL
WS Endpoint Language
ebXML CPP
Collaboration Protocol Profile
WSDL
Web Service Description Language

Figure 1: Process Coordination Framework [1]

The need for different kind of technologies has increased. Modern technologies are mutually connected and complemented. Figure 1 presents their relationships, horizontally divided by the level of provided services and vertically by the initiative organization, by which they were sponsored and created [1]. All of the technologies are an upgrade of web services and they are all based on the XML language. Their design priorities and fields of concentration however differ.

Service description and transport binding was assured in WSDL and in ebXML CPP (Collaboration Protocol Profile) as the EDI follower. With time, new technologies emerged from them in different directions. Their relations are seen from Figure 1: WSEL (Web Service Endpoint Language), ebXML BPSS (Business Process Specification Schema), WSCL (Web Service Conversation Language), WSFL (Web Service Flow Language), XLANG, BMPL (Business Management Markup Language), RosettaNet PIPs (Partner Interface Process) and ebXML CPA (Collaboration Protocol Agreement) [1].

WSDL is meant for describing network services as a set of endpoints, operating on messages. WSEL is meant for non-operational features of web services like security. WSCL allows that we define abstract interfaces for web services for business process conversation. WSFL allows the description of business processes or interaction patterns, based on the web services operations.

ebXML provides a set of technologies for describing various stages of business collaboration. CPPs enable companies to specify their profiles in which they define the terms for collaboration. CPAs are the computer equivalents of trading partner agreements. They can be defined manually or automatically generated from two or more CPPs. The actual flow of a business process is specified using BPSS. Shared public and private business processes for collaboration between two or more partners are specified using BPML. The focus of XLANG and RosettaNet PIPs is similar to BPML and will be further discussed later in this article.

XML, concentrated on the contents, enables remote systems to interchange and interpret the documents without the human intervention. XML document is basically an ordinary text file with markup [1]. The combination of structure, flexibility and verification makes XML useful not only for electronic publishing, but also for designing business messages, exchanged between enterprises [1]. While building larger processes, all business partners must agree upon the vocabulary, interfaces and the type of method invocation, before they send individual messages.

XML vocabularies can define all kinds of business documents or even whole frameworks, which provides interoperability and functionality.

3.1 ebXML

ebXML is a family of specifications that enable companies of all sizes to collaborate with each other, independently of the location [2], through the exchange of XML-based messages [8]. Development of the ebXML specifications is an on-going effort sponsored by OASIS (Organization for the Advancement of Structured Information) and UN/CEFACT (United Nations Center For Trade Facilitation & Electronic Business) [8].

The need for ebXML lies in the experience from the past. EDI, the anterior technology for data interchange among enterprises, was unreachable for most SMEs, since the costs were too high and the implementation too complex. ebXML is based on XML, web services and open standards and is publicly available. It overcomes this barrier and enables the creation of software for building applications, based on mutual structure and syntax, which will lower the costs of business data interchange. ebXML mission is to provide an open XML-based infrastructure, enabling the global use of electronic business information in an interoperable, secure and consistent manner by all parties [8].

ebXML architecture was primarily designed for B2B interaction. UDDI and SOAP offer services with similar functionality on the low level. EbXML uses and builds upon these standards. It provides safe and reliable messaging and adds a set of higher level specifications for expressing the semantics of B2B collaborations. For these purposes it provides CPPs, CPAs, BPSS, core components, registry/repository and BPML [11].

ebXML provides an effective platform for long-term business transactions and enables us to express the following:

- quality of service,
- timeouts,
- conformations,
- multi-language support,
- authentication,
- authorization,
- privacy,
- integrity and
- non-repudiation.

Example of ebXML usage, shown in Figure 2:

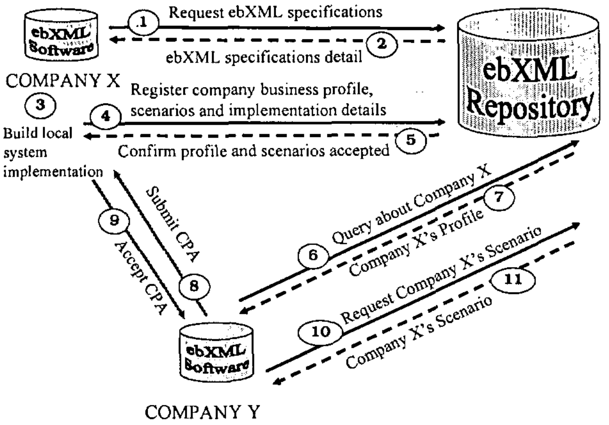

ebXML Software
COMPANY X
1
Request ebXML specifications
2
ebXML specifications detail
ebXML Repository
3
Build local system implementation
4
Register company business profile, scenarios and implementation details
5
Confirm profile and scenarios accepted
Submit CPA
8
9
Accept CPA
6
Query about Company X
7
Company X's Profile
10
Request Company X's Scenario
11
Company X's Scenario
ebXML Software
COMPANY Y

Figure 2: ebXML in practice

By using ebXML, companies have a standard method to exchange business messages, conduct trading relationships, communicate through data in common terms, define and register business processes [8]. It enables all parties to complement and extend current EC/EDI (electronic commerce/EDI) investment and it expands electronic business to new and existing partners. It also facilitates convergence of current and emerging XML efforts [8].

ebXML delivers the value by [8]:

- using the strengths of OASIS and UN/CEFACT to ensure a global, open process,
- developing technical specifications for the open ebXML infrastructure,
- creating the technical specifications with the world's best experts,
- collaborating with other initiatives and standard development organizations,
- building on the experience and strength of existing EDI knowledge,
- enlisting industry leaders to participate and adopt ebXML infrastructure and
- realizing the commitment by ebXML participants to implement the ebXML technical specifications.

3.2 XLANG

XLANG is a notation for the specification of message exchange behavior among participating web services, supporting especially the automation of business processes [9]. It is expected to serve as the basis for automated protocol engines that can track the state of process instances and help to enforce protocol correctness in message flows.

XLANG is based on XML and is used for describing business processes in the BizTalk initiative. It offers a model for orchestration of services and contract collaboration between partners [3]. XLANG is fully focused on public processes. It supports long-term operations and nesting. It enables:

- exception handling,
- restoring operations,.
- behavior,
- actions,
- control flow,
- correlations,
- contents of transaction,
- service management,
- time-outs,
- custom correlation of messages,
- modular behavior description and
- contracts with multiple roles [3].

However, it does not define authentication or the quality of service nor the non-repudiation [4]. The goal of XLANG is to make it possible to formally specify business processes as state-full long-running interactions [9].

Main features of XLANG include [1]:

- *behavior;* container for the description of the service's behavioral aspects, including support for looping, concurrency and exception handling,
- *actions;* atoms of behavior, referencing WSDL operations on available ports,

- *control flow;* sequence in which the service performs actions,
- *correlations;* structure, the service uses to route messages to correct workflow instances,
- *context;* context for long-running transactions,
- *service management;* features of service instance management and
- *port mapping;* method for plugging in the service user and the service provider.

XLANG is an extension of WSDL and dynamics in processes are supported with different flows [3]:

1. *Message flow*, where actions are the basic constituents of an XLANG process definition that specifies the behavior of the service. The actions are request/response, solicit response, one way, notification, timeouts and exceptions.
2. *Data flow*, the base of XLANG is fed by the message flow and supports the control flow decisions.
3. *Control flow*, which provides support for looping, besides the regular elements. It also enables exception handling and transactional behavior.

XLANG also supports business process contracts, however they are merely mappings between two port types, which interact together. A contract can only map ports that are unidirectional [3].

The unit of action, offered by a service is an *operation*. An operation can be a single asynchronous message, or a request/response pair of messages with optional fault messages. The operation can be either incoming or outgoing. But WSDL does not say what is the operation semantics. There are three possibilities [17]:

1. In the first case the operation is a *stateless service that has no memory of previous operations,* such as a stock quote service.
2. The second possibility is an *operation on an object,* in the usual sense of object-oriented programming systems, in which case the object will have the ability to use its state variables to keep a record of the consequences of previous operations. In the latter case, we usually think of the object as being subservient to the caller, since the caller controls the entire life cycle of the object. The object itself has low influence regarding the order in which its operations are invoked and no independent behavior.
3. The third possibility is *autonomous agents with full state representation of the service.* In this case the service supports long-term interactions with full state, in which every interaction has a beginning, defined protocol for operation call and the ending. The supplier has to provide a service, which starts an interaction by receiving an order through the entering message, then returns the acknowledgement to the buyer, if the order can be accomplished.

Enterprise workflow systems today support the definition, execution and monitoring of long-running processes that coordinate the activities of multiple business applications. But they do not separate internal implementation from external protocol description [9].

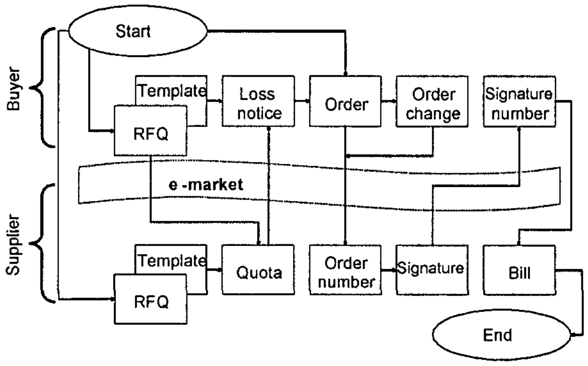


Figure 3: XLANG connecting two parties [9]

The Figure 3 represents the dynamics between two participants inside an electronic market, where XLANG is the translating key between a buyer and supplier that cooperate on the net using the advantages of the electronic market.

3.3 RosettaNet

RosettaNet is a non-profit consortium of more than 400 of the world's leading Information Technology, Electronic Components, Semiconductor Manufacturing and Solution Provider companies, working to create, implement and promote open electronic business process standards [7].

RosettaNet was created as a compromise between EDI and SOAP. Its main goals are reaching dynamic, flexible trading networks, operational efficiency and new business opportunities [10]. It enables:

- real time complex transitions,
- checking,
- confirmation,
- non-repudiation,
- multiple languages,
- additional standards in industry,
- SSL (Secure Socket Layer) authentication,
- digital signature and
- data encoding.

Its biggest advantage is the well defined although inflexible PIP [5]. The purpose of every PIP is to offer general business data models and documents, which enable interface implementation by system developers. Every interface includes [14]:

- XML document, which is based on the DTD (Document Type Definition) and specifies PIP services, transactions and management, which include dictionary properties,
- class and sequence diagrams in UML,
- validation tool and
- implementation guide.

PIP interface offers mechanism for sending messages and reporting failures. It demonstrates the integration of web services and its safety features, demanded at RosettaNet [19]:

- two – way SSL authentication,
- digital signature,
- data encryption and
- non-repudiation.

RosettaNet PIP defines an automated business process among trading partners for demanding and offering product prices and availability information [16]. Different business processes are covered with:

1. RosettaNet *executive plan*, which offers a general guidance, priorities of addresses and integration through tables.
2. Individual *plan of supply chain*, which address of the supply chain – specific theme, prioritization, sources, implementation and adaptation.
3. RosettaNet *partners*, which enable voting about standards, participants in workshops and implementation.

RosettaNet standards are managed on a global level. Locally they are focused on implementation and support. So partners can choose between global or local membership [13].

RosettaNet is very rich in its supporting tools: the RosettaNet implementation tool including the current PIP template, a Partner Agreement Wizard for quick importation, development and testing of customized PIP and more. It also contains RosettaNet dictionary and RosettaNet implementation framework. The template enables the development of new PIPs. The Partner Agreement Wizard enables importing of trading partners and a fast development of new processes. Embedded PIP enables implementation of only that certain PIP the partner needs. It includes support for all published RosettaNet PIPs as well as for CIDX (Chemical Industry Data Exchange) and PIDX (Petroleum Industry Data Interchange). PIP can also be tested before actually applied and used.

RosettaNet is also focused on the industry support; the adapter for industry development enables integration with new and existing applications and ways of business [15].

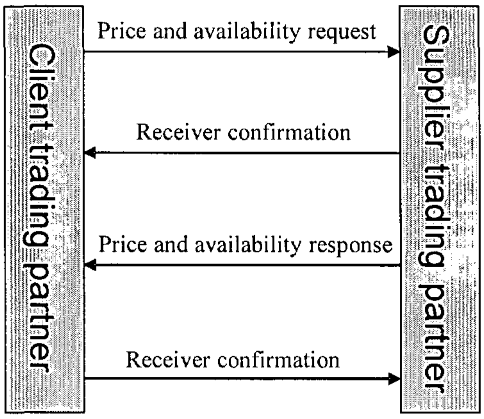


Figure 4: RosettaNet communication

RosettaNet plans to integrate support for the ebXML Messaging Services Specification in future releases of RosettaNet's Implementation Framework (RNIF). While RosettaNet remains committed to developing business process standards, required to support the complex needs of the high-technology industry, it also wants to ensure interoperability across all supply chains. Figure 4 represents the communication between two trading partners with help of RosettaNet PIP – which enables connection of business processes [12].

4 Evaluation Model

4.1 Criteria

To be able to evaluate the technologies for describing business processes for their suitability and quality, we have defined a multi-criteria decision model. We have identified the following criteria [18]:

Defining and describing processes: Evaluates the architectural support, syntax and semantics for describing all the features of the process and the support for the transition from classical to electronic business from aspects of flexibility, simplicity, user friendliness and compliance to standards.

Collaboration description: Evaluates the support for business interactions and defining relationships between partners, from aspects of flexibility, safety and complexity.

Role model: Evaluates the support with modeling tools for describing roles and collaboration between them.

Small/big/medium enterprises support: Evaluates the appropriateness and flexibility of the technology for different company sizes with different characteristics, needs and preferences.

Complexity and learning effort: Evaluates the amount of effort and change, needed to learn and understand the technology and all its features.

Efficiency: Evaluates how efficient is the technology at describing and specifying the business processes.

Maturity: Evaluates the maturity, based on the number of years the technology exists.

Tools support: Evaluates the support within tools and integrated development environments, which ease the development and assure quality.

Synchronous communication support: Evaluates support for synchronous, short-term transactions, which require immediate answer.

Asynchronous communication support: Evaluates support for asynchronous, long-term transactions.

Independency of communication protocols: Describes the relationship between communication protocol and the technology.

Quality of service: Evaluates the possibilities for specifying service quality of certain flows, which can be done either by raising the priority of a flow or limiting the priority of another flow.

Authentication: Evaluates the level of verification of the senders identity - whether the business message sender is or is not who he claims to be [6].

Authorization: Evaluates the level of verification, whether the sender of a message is permitted to send the subject message to the receiving partner [6].

Integrity: Evaluates, whether the messages remains unaltered during transportation [6].

Encryption: Evaluates the coding and the level of security of messages against unauthorized readers [6].

Non-repudiation: Evaluates the mechanism for verifying whether an originating trading partner can or cannot deny having originated and sent a message and that a receiving trading partner can or cannot deny having received a message, sent by its partner [6].

Exceptions handling: Evaluates the business preparation for every sort of failures, duplications and losses of data.

Claim detection: Evaluates the preparation and support for events of claim loss.

Data transformation: Evaluates the possibilities, tools and technologies for data transformation between collaborating enterprises.

	Criteria definition	*Scale defining*
c1	Defining and describing processes	2 – flexible, simple, compliant with standards 1 – simple and user friendly 0 – basic features only
c2	Collaboration description	2–multiple language, flexibility, safety 1 – safety and basic features 0 – basic features
c3	Role model	1 – yes 0 – no
c4	Small/big/ medium enterprises support	2 – big/medium/small 1 – big/medium 0 – big
c5	Complexity and learning effort	2 – simple 1 – moderate effort 0 – great effort
c6	Efficiency	2 – very efficient 1 – averagely efficient 0 – low efficiency
c7	Maturity	Actual number in years
c8	Tools support	2 – many 1 – medium 0 – low
c9	Synchronous communication support	1 – yes 0 – no
c10	Asynchronous communication support	1 – yes 0 – no
c11	Independency from communication protocols	1 – yes 0 – no
c12	Quality of service	1 – yes 0 – no
c13	Authentication	1 – yes 0 – no
c14	Authorization	1 – yes 0 – no
c15	Integrity	1 – yes 0 – no
c16	Encryption	1 – yes 0 – no
c17	Non-repudiation	1 – yes 0 – no
c18	Exceptions handling	2 – handling message loss, resolution, system recovery 1 – two above 0 – one above
c19	Claim detection	2 – good 1 – average 0 – poor
c20	Data transformation	2 – good 1 – average 0 – poor

Table I: Criteria and scale

4.2 Utility Function

We have defined the utility function, which organizes the results, for them to be comparable (on scale between 0 and 1). In the case, where input value is an actual number, the utility function transforms it to the closed interval from 0 to 1.

Equation 1: Utility function

$$u_j = \sum_{i=1}^{6} \left(\frac{w_i}{100} \cdot \frac{c_i(A_j)}{2}\right) + \left(\frac{w_7}{100} \cdot \frac{c_7(A_j)}{\sum_{k=1}^{N} c_7(A_k)}\right) + \sum_{i=8}^{29} \left(\frac{w_i}{100} \cdot \frac{c_i(A_j)}{2}\right)$$

Equation 2: Maximum utility

$$U = \max_{j=1}^{N}(u_j)$$

Meaning of the symbols:

- U – maximum utility,
- u_j – utility of alternative j,
- c_i – criterion i (Table I),
- A_j –alternative j (ebXML, XLANG, RosettaNet),
- w_i – weight of criterion i,
- N – total number of alternatives.

4.3 Results

For the purposes of the evaluation of the technologies in this article we have selected the weights based on the preferences of a SME, where security (authentication, authorization, integrity, encryption and non-repudiation), defining and describing processes, collaboration support, complexity and learning effort, maturity, tools support, data transformation, exception handling and quality of service are particularly important. The selection of the weights is based on the survey, done in [18]. The weights can however be altered according to the needs and priorities of each distinctive business. The Table II shows the evaluation of ebXML, RosettaNet and XLANG. It is divided in 5 columns. The first column presents criteria. The second column shows the weights, which we assigned to each criterion. The rest of the columns show evaluations for each technology, using the scale, explained in the third column of Table I. In the last row we show the results calculated using the utility function.

As seen in Table II ebXML has achieved the highest result. It turns out that ebXML is the best technology for most of the businesses.

XLANG is second best, although it lacks the quality of service, authentication and non-repudiation. However, it is integrated within the BizTalk Server Initiative, which is very promising. We believe that it will get improved over time.

RosettaNet is the least appropriate for general SMEs. Its main preference lies in technical features and level of development. Since it is the oldest technology of the three, it is the most mature one. Its main disadvantage is in the fact that it is suitable mainly for very large companies, since its framework PIP is very inflexible, and once created, very difficult to alter thus inappropriate for smaller businesses.

c	*w*	*ebXML*	*XLANG*	*RosettaNet*
c1	9	2	2	0
c2	8	2	1	1
c3	2	1	0	1
c4	10	2	2	0
c5	9	1	1	2
c6	7	2	1	1
c7	8	2	2	4
c8	2	1	1	2
c9	2	1	1	1
c10	2	1	1	1
c11	2	1	0	1
c12	6	1	0	1
c13	3	1	1	1
c14	3	1	1	1
c15	5	1	1	1
c16	3	1	1	1
c17	4	1	1	1
c18	6	2	2	1
c19	2	2	1	1
c20	7	2	2	1
		0,735	0,606	0,498

Table II: Evaluation matrix and results

5 Conclusions

The need to do business on the net and to automate business processes is increasing, as is the need for supporting technologies. Such technologies must satisfy certain standards, they must be flexible and available to all organizations, large but particularly to small and medium enterprises. Describing business processes must be relatively simple, so that even non-programmers can use it, since the business process experts usually do not have the necessary knowledge, needed to work with complex languages.

In the article we have identified, compared and evaluated the features of the three most important technologies and upon our findings defined a multi-criteria decision model for their quantitative evaluation. The defined decision model is usable for all kinds of enterprises. They can express their priorities through criteria weights. For the purposes of this article we have also defined a common set of weights for small and medium enterprises and done the evaluation of the technologies. From this perspective we have determined that ebXML technology is the most suitable with the widest range of possibilities, followed by XLANG and RosettaNet.

References

[1] Selim Aissi, Pallavi Malu, Krishnamurthy Srinivasan (May 2002), E-Business Process Modeling: The Next Big Step, *IEEE Computer*, pp. 55-62.

[2] Alan Kotok, David R.R. Webber (2002), *The new global standard for doing business over the Internet ebXML,* New Riders Publishing.

[3] EbPML.org (2002), XLANG, http://www.ebpml.org/xlang.htm.

[4] David O'Riordan (April 10th 2002), Business Process Standards for Web Services, The candidates, *Services Business Strategies and Architectures,* http://www.webservicesarchitect.com/content/articles/oriordan01.asp

[5] Joe McKee (May/June 2002), RosettaNet at the Dance–an e-business standard does its own choreography, *Oracle technology network*, pp. 51-52.

[6] Pekka Kantola, Janne J. Korhonen (15.5.2002), RosettaNet vs. ebXML–Security Solutions and exception handling, *Helsinki University of Technology*, http://www.soberit.hut.fi/T-86/T-86.161/2002/RosettaNet

[7] Vitria (February 26th 2002), RosettaNet E-Business Process Standards for the High-Tech Industry, *Vitria Technology, Inc.,* http://www.vitria.com/news/press_releases/pr_2002-02-26.html.

[8] EbXML, Technical specifications, http://www.ebxml.org/.

[9] Cover Pages hosted by Oasis (June 6th 2001), XLANG, Technology Reports, http://xml.coverpages.org/xlang.html.

[10] RosettaNet Overview, Background Information, http://www.rosettanet.org/RosettaNet/Rooms/DisplayPages/LayoutInitial.

[11] Madhu Siddalingaiah (August 17th 2001), Overview of ebXML, , *Technical overviews, SUN Microsystems,* http://dcb.sun.com/practices/webservices/overviews/overview_ebxml.jsp

[12] Arsin Corporation (2002), Solution Integration Services, *Arsin RosettaNet PIP Solution,* http://www.arsin.com/docs/RNTFactSheet_final.pdf.

[13] Andy Moir (April 2002), Introduction to RosettaNet, *XML.gov,* http://xml.gov/presentations/rosettanet.

[14] Cover Pages hosted by Oasis (November 2002), RosettaNet, Technology report, http://xml.coverpages.org/rosettaNet.htm.

[15] Microsoft BizTalk Server (2002), BizTalk Accelerator for RosettaNet Features, *Microsoft,* http://www.microsoft.com/biztalk/evaluation/features/rosettanet.asp.

[16] Bea Web-Logic Integration (2002), RosettaNet 2.0 Security Sample, *Bea,* http://edocs.bea.com/wli/docs70/b2bsampl/rn2sec.htm .

[17] XLANG (2001), Web Services for Business Process Design, *2001 Microsoft Corporation,* http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm.

[18] Object Technology Center (2002), Preferences of small and medium businesses, *Technical Report, FERI*

[19] Matjaz B. Juric, S. Jeelani Basha, Rick Leander, Ramesh Nagappan (December 2001), *Professional J2EE EAI*, Wrox Press Ltd.

[20] Arijit Sengupta (2002), Oracle's support for open eBusiness standards, *Oracle corporation,* http://www.idealliance.org/papers/xmle02/slides/Sengupta/sengupta.ppt.

[21] Arijit Sengupta (2002), Data integration, process integration and trading partner agreements, *Oracle corporation,* http://www.edifice.org/ERUG/Sengupta_ERUG.ppt.

[22] Paavo Kotinurmi (Oktober 22nd 2002), Comparing XML Based B2B Integration Frameworks, *Helsinki University of Technology SoberIT,* http://www.soberit.hut.fi/ICTEC/lectures/20021022_Kotinurmi.pdf.

Visual Secret Sharing Watermarking for Digital Image

Shen-Chuan Tai, Chuen-Ching Wang*, and Chong-Shou Yu
Institute of Electrical Engineering,
National Cheng Kung University
Tainan, Taiwan, R.O.C
Address: Institute of Electrical Engineering (computer / group 92533)
National Cheng Kung University, Tainan, 701, Taiwan
Email: wcj@rose.ee.ncku.edu.tw

Keywords: Visual Secret Sharing Watermarking (VSSW), Discrete Cosine Transform (DCT), Digital Watermarking, Copyright Protection.

Received: February 23, 2001

A visual secret sharing watermarking (VSSW) technique is proposed as a way of solving copyright protection problems for digital images. The proposed watermarking technique employs a visual secret sharing (VSS) scheme and separates the watermark into two parts, a public watermark and a secret watermark. For watermarking security, only the public watermark is inserted into the original image, while the owner holds the secret watermark. Without the secret watermark, it is almost impossible to extract the watermark even if the embedding algorithm is published. To meet requirements of robustness and imperceptibility, we modify DCT coefficients belonging to the middle frequency band to embed the public watermark. Importantly, the watermark can be retrieved from the watermarked image without resorting to the original image. Various experiments using the proposed watermarking method are presented to demonstrate robustness to tampering and a to variety of common image processing operations and geometric manipulations.

1 Introduction

Protection of intellectual property is an increasingly important concern as widespread use of the Internet is making multimedia data increasingly easily copied and distributed. Fortunately, digital watermarking techniques allow us to embed copyrights into digital contents and later extract the watermark to detect copyright infringement and confirm legal ownership.

Many watermarking techniques [1 – 12] have been published in the literature. These published techniques utilize either transform domain or spatial domain. In [2], Cox et al. describe a method to embed into the host image a watermark composed of a randomly generated sequence. This scheme applies the full-frame Discrete Cosine Transform (DCT) to the original image, producing a set of coefficients. Then, a subset of these coefficients is chosen according to a rule that depends on the most perceptually significant coefficients of the DCT transform domain. To embed the watermark, they use a scaling modulator to alter the values of these coefficients according to the values of the watermark. Cox's watermarked version is robust to some attacks involving common signal and geometric processing operations. However, there are some drawbacks to this technique. First, extracting the watermark requires the original image for watermark detection. This limitation affects its application on the Internet. Second, the authors use a threshold of similarity measure to determine whether the host image is watermarked or not. In practice, selection of too small or large a threshold will lead to watermark detection error. Third, modulating the most significant coefficients (excluding the DC term) with a random sequence degrades image quality, and is thus an unreasonable requirement for a general watermarking scheme.

Hsu and Wu [4] proposed a frequency-domain watermarking technique that used fixed block-base DCT transformation. The method first breaks up the host image into 8×8 blocks and then performs the DCT on each block. In the embedding algorithm, they select 16 middle-band coefficients from each block and then modify these coefficients according to the residual mark to reverse the corresponding polarity. After this procedure, the watermark is embedded into the host image to form a watermarked image. Unlike [2], this method makes use of a binary image as the watermark, thereby allowing identification of the extracted watermark by direct use of the unaided human eye. Both human visual recognition and a similarity measurement were used experimentally to verify the efficiency of their watermark extraction method. However, there are some drawbacks to this technique. First, this method cannot overcome certain attacks, for example image rotation and image resampling. Second, for extraction, this scheme also requires the original image to extract the watermark information. For watermarking systems applicable to the Internet, security is a very important concern. The watermarking systems of [2] and [4] require the original image for watermark retrieval, making verification complicated, necessitating the original image be shared in a public place or network for ownership verification and thus making these systems unsuitable for Internet application.

In this paper, a new watermarking technique based on VSS scheme for enhancing the watermarking security is presented. The proposed method operates in a full-frame DCT domain, which allows a reasonable tradeoff between quality and robustness. More importantly for Internet application, watermark extraction does not require the original image and the original watermark, thus simplifying the watermarking system and allowing the original image and original watermark to be kept secret. Further, because it is dangerous to trust only a single person or organization to manage very important information, the proposed algorithm includes a visual secret sharing scheme (VSS) which shares a secret among a limited number of members.

This paper is organized as follows. Section 2 introduces the basic concept of the VSS as applied in the proposed watermarking system. The watermarking technique itself is described in section 3. Experimental results are shown in section 4. Finally, section 5 presents conclusions.

2 The Basis of Visual Secret Sharing Scheme

VSS is a well-known cipher technique for digital images. Decoding can be performed by the naked eye, with no instrumentation or complex computation. The concept of VSS is derived from [13]. In [14], Naor and Shamir extended this idea to (*k, n*)-VSS, which is designed to break a shared image into *n* different shadows. Each single shadow look like random data. The shared image can be recovered easily from *k* ($k \leq n$) shadows or more. That is, k person's permission is required to decode the shared image. More detailed description can be found [13-15].

For simplicity, the *(2,2)*-VSS scheme is used in the proposed watermarking method. The shared image is divided into 2 shadows that consist of random dots. The mapping relationships of the *(2,2)*-VSS scheme based on [14] are shown in Table 1. For each pixel in the shared image, two blocks of 2×2 pixels are generated in the corresponding location of the shadow images, one for shadow 1 and the other for shadow 2. If a pixel P_i in the shared image is black, then any one of the first six rows in Table 1 for the two 2×2 blocks of shadow images can be selected. If P_i is white, any one of the last six rows in Table 1 for the two 2×2 blocks of the shadow images can be selected. The two 2×2 blocks are copied to the corresponding position in shadows 1 and 2, respectively. According to the human visual system, both shadow 1 and shadow 2 now contain 50% white sub-pixels and 50% black sub-pixels and appear like random noise. The human eye cannot read the secret message in the independent shadow and, thus, the shared secret can be concealed in shadow 1 and shadow 2.

Table 1 The mapping relationship function of (2,2)-VSS scheme based on [14].

Pixel	block1	block2	block1 *visual OR* block2
■	▚	▞	█
■	▞	▚	█
■	▌	▐	█
■	▐	▌	█
■	▄	▀	█
■	▀	▄	█
□	▌	▌	▌
□	▐	▐	▐
□	▚	▚	▚
□	▞	▞	▞
□	▀	▀	▀
□	▄	▄	▄

When both shadows are superimposed on each other, the shared secret is clearly visible to the human visual system. The decoder is thus the unaided human eye, which is very similar to an ordinary OR function performed on two sub-pixels. Therefore, if pixel P_i in the shared image is black, then the corresponding 2×2 block created by superimposing shadows 1 and 2 will contain four black sub-pixels. On the contrary, if P_i is white, then the corresponding 2×2 block by superimposing shadows 1 and 2 will contain two black sub-pixels and two white sub-pixels. From the viewpoint of the human visual system, the block with two black sub-pixels and two white sub-pixels will be recognized as a white pixel, while the block with four black sub-pixels will be recognized as a black pixel. Consequently, the secret information in the shared image can be easily detected when these shadows are superimposed together by way of a pixel-by-pixel visual OR operation.

3 Watermarking Implementation Process

The proposed watermarking process involves watermark generation, embedding and extracting. During watermark generation, the VSS technique is used to enhance the security of the embedded watermark. In our proposed scheme, only the public watermark is embedded in the image. The copyright owner reserves the secret watermark for reconstructing the watermark. In order to embed the watermark, we first transform the host image as DCT domain and process the coefficients in the middle band into coefficient pairs. Next, each pixel of the public watermark is inserted into the middle band by modifying the location of the two coefficients in each coefficient pair. Then, inverse DCT transformation is performed and thus a watermarked image is obtained. Finally, the reconstructed public watermark extracted from the watermarked image may be superimposed on the secret watermark to recover the watermark.

Consider a grayscale image H of size $N \times N$ pixels. Also, let W be a binary image of size $k \times l$ pixels that we will use as a watermark. We want to embed an invisible watermark into H to form a watermarked image H_W. Host image, watermark and watermarked image can be denoted by:

$$H = \{h(\rho,\sigma)|0 < \rho,\sigma < N\}h(\rho,\sigma) \in \{0,1,...255\} \quad (1)$$

$$W = \{w(i,j)|0 \le i < k, 0 \le j < l\}w(i,j) \in \{0,255\} \quad (2)$$

$$H_W = \{h'(\rho,\sigma)|0 < \rho,\sigma < N\}h'(\rho,\sigma) \in \{0,1,...,255\} \quad (3)$$

3.1 Watermark Generation

As described in section 2, the VSS technique can partition a shared image into two or more shadows. For convenience, we set the watermark size at 50 by 50. Watermark W is broken down into two sub-images, the public watermark W_P and the secret watermark W_S. Each sub-image with size of 100×100 sub-pixels is just a shadow of the (2,2)-VSS scheme. Each pixel in the original watermark is now represented as the ORed result of two 2×2 blocks, each from the corresponding position of W_P and W_S, respectively. In the proposed method, the owner randomly assigns the secret watermark. After defining the secret watermark, the public watermark can be generated by using the relationships based on (2,2)-VSS scheme and listed in Table 1. For each pixel P_i in W and each block B_S in W_S, the block B_P in W_P can be derived as:

$$B_p = \begin{cases} \overline{B_S} & \text{if } P_i = black \\ B_S & \text{if } P_i = white \end{cases} \quad (4)$$

where $\overline{B_S}$ stands for inverting each sub-pixel in the block B_S. For example, if P_i is a black pixel in W and the corresponding block B_S in W_S is $\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$, then the corresponding block B_P in W_P is the complement of $\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$, that is $\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$. In accordance with the ongoing example, the public watermark is finally obtained by collecting each block B_P. It can be expressed as:

$$W_P = \{w_p(\alpha,\beta) \in \{0,255\}0 \le \alpha < 2k, 0 \le \beta < 2l\} \quad (5)$$

3.2 Watermark Embedding

3.2.1 DCT Transformation of the Image

In this approach, each pixel of the public watermark is embedded into the middle-frequency area of the host image. The block-diagram of watermark embedding is shown in Fig. 1. To spread the energy of the host image, we use full-frame DCT instead of block DCT transform, after which the watermark information can be embedded into the transform domain. When the watermark information is dispersed over the entire spatial image, then the watermark can easily survive common image processing. The transformation is:

$$\widetilde{H}(u,v) = C(u)C(v)\sum_{\rho=0}^{N-1}\sum_{\sigma=0}^{N-1} h(\rho,\sigma)\cos\left[\frac{(2\rho+1)u\pi}{2N}\right]\cos\left[\frac{(2\sigma+1)v\pi}{2N}\right] \quad (6)$$

$$C(u) = \begin{cases} \sqrt{1/N}; & u = 0 \\ \sqrt{2/N}; & otherwise \end{cases}$$

where $\widetilde{H} = \{\widetilde{h}(u,v) \in R, \quad 0 \le u,v < N\}$ and R is a real number.

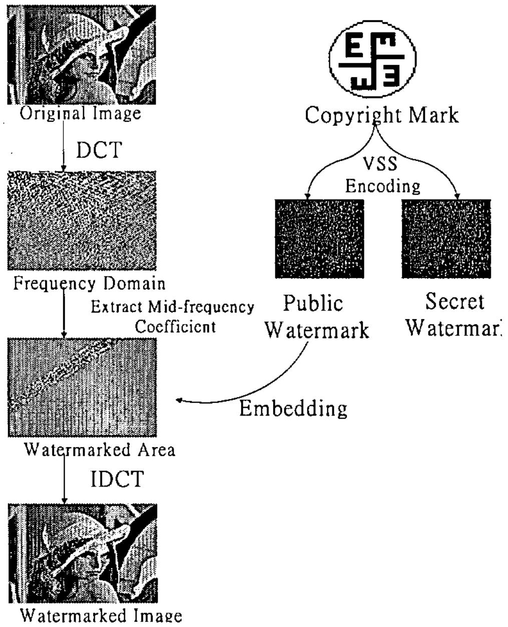


Figure 1: Block-diagram of embedding procedure.

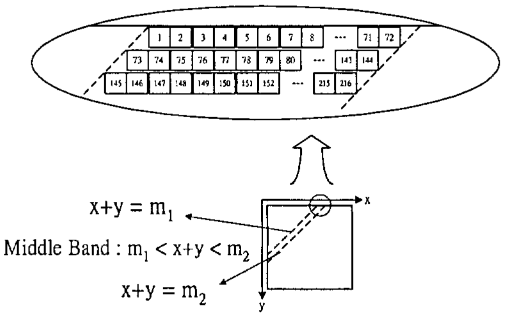


Figure 2: Middle-band determined by two lines.

3.2.2 Selection of the Middle-band Coefficients

In fact, the robustness and invisibility are conflict each other. To be a reasonable tradeoff, we select middle band components for embedding the watermark. As seen in Fig. 2, the two lines $x+y=m_1$ and $x+y=m_2$ set the boundaries for the region selected for hiding the watermark. That is, the middle band is selected by:

$$m_l < x+y < m_2 \qquad (7)$$

Here, x, y is the position of each coefficient in the host image, and m_l and m_2 represent intercepts that are located on the x and y axis respectively. These parameters, i.e. m_l and m_2, can be adaptively selected to control the size of the middle band for embedding watermark.

3.2.3 Processing the Rank of the DCT Coefficient Pair

As mentioned above, the watermark information can be embedded in middle band, the watermark information, W_P, is added into the middle band as follows.

Step 1. Process the coefficients of the middle band to form a one-dimensional sequence by row major scanning. That is, C = $\{ C_1, C_2 ..., C_{2\times 2k\times 2l} \}$.

Step 2. Process the W_P as a one-dimensional sequence by row major scanning. Eq. (5) will be written as:

$$\psi = \{ \overline{\varpi}(\gamma) \in \{0,255\} 0 \le \gamma < 4kl \} \ (8)$$

where $\varpi(\gamma) = w_p(\alpha,\beta)$ with $\gamma = \alpha \times 2l + \beta - 1$

Step 3. For i = *1,2...,2k× 2l*, pack the coefficient,C_{2i-1} , and the neighboring coefficient, C_{2i}, into an i-th coefficient pair, $CP_i = (C_{2i-1}, C_{2i})$.

Step 4. Select a random number as a secret key, S, to generate the $2k\times 2l$ different random sequence over the interval [1, $2k\times 2l$]. That is,

R=$\{R_1, R_2 ..., R_{2k\times 2l}\}$, where R_i denotes the i-th random number.

Step 5. Define RANK:

$$(C_{2i-1}, C_{2i}) = \begin{cases} 1 & if (C_{2i-1} > C_{2i}) \\ 0 & if (C_{2i-1} \le C_{2i}) \end{cases} \qquad (9)$$

Step 6. For i =*1,2...,2k × 2l*, embed the public watermark into the middle band by modifying all coefficient -pairs according to the following rule.

If $\{\ \varpi(R_i) \oplus RANK(C_{2i-1}, C_{2i}) = 1\ \}$

Then { swap the coefficients of C_{2i-1} and C_{2i} }

Else { no operation }

where ϖ (.) represents a gray level pixel of watermark ; here we suppose that gray level 255 and 0 denote logic 0 and logic 1, respectively.

3.2.4 Inverse DCT Transformation

As mentioned in the previous section, some coefficients in the middle-band will change their position. Indeed, this process is equivalent to the watermark embedding operation. To obtain a watermarked image, the Inverse Discrete Cosine Transform is used to transfer the frequency domain. The watermarked image is:

$$H_W = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} C(u)C(v)\widetilde{h}(u,v)\cos\left[\frac{(2\rho+1)u\pi}{2N}\right]\cos\left[\frac{(2\sigma+1)v\pi}{2N}\right] \qquad (10)$$

3.3 Extracting Watermark Procedures

In the extraction procedure, reconstructing watermark is really quite simple. The proposed technique does not use the original image to detect the watermark. The extraction procedure is shown in Fig. 3. First, we use the DCT transform on the watermarked image and, thus, $\widetilde{H}_w$ = DCT(H_w) is obtained. Next, the coefficients of the middle band are selected as paragraph 3.2.2. Then, the coefficients are expanded and packed into coefficients by the same method as the watermark embedding process. After that, the secret key, S, is applied to generate the predefined random number and, thus, each pixel of the public watermark, according to this predefined random order, can be retrieved from the watermarked image. The procedure for extracting the reconstructed watermark W_P' is briefly described as follows:

Step 1. By row major scanning, we form the coefficients of the middle band in $\widetilde{H}_w$ as a one-dimension form, $C' = \{ C_1', C_2' ..., C_{2\times 2k\times 2l}' \}$.

Step 2. For i = *1,2...,2k × 2l*, pack the coefficient, C_{2i-1}' , and the neighboring coefficient, C_{2i}' , into an i-th coefficient pair, CP_i = (C_{2i-1}', C_{2i}').

Step 3. Use the secret key, S, to generate the predefined random number as a set, that is

$R = \{R_1, R_2 ..., R_{2k\times 2l}\}$, where R_i denotes the i-th random number.

Step 4. According to R, extract the i-th pixel of the watermark by judging the R_i coefficient pair as the following equation:

$$w'_p(i) = \begin{cases} 1 & if\, C_{2i}' > C_{2i+1}' \\ 0 & if\, C_{2i}' \le C_{2i+1}' \end{cases} \qquad (11)$$

where $w_p'(i)$ stands for the i-th pixel in the W_P' and "1" and "0" represent gray-level 0 and gray-level 255, respectively.

Step 5. Assemble all the pixels to obtain a reconstructed public watermark W_P'.

Step 6. Superimpose the reconstructed public watermark (W_P') on the secret watermark (W_S) to obtain the reconstructed watermark (W').

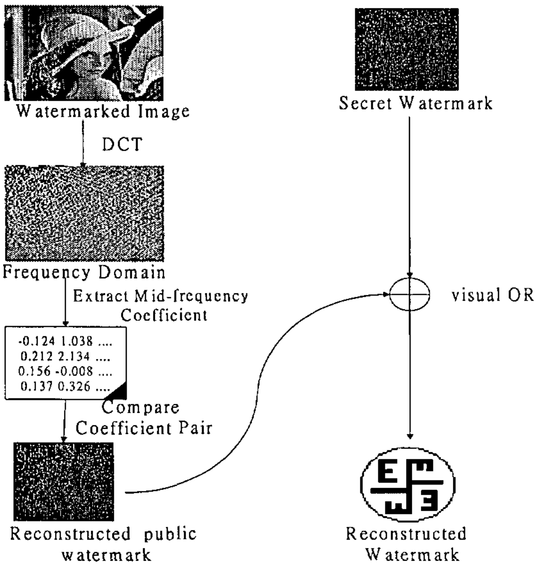


Figure 3: Block-diagram of extraction procedure.

3.4 Reduction Process

As mentioned section 3.3, the reconstructed watermark can be directly used for identifying the ownership protection. To improve the clarity of W', we use a post-process called the "reduction process" to reduce the redundancy of data caused by VSS scheme. Indeed, this process is a quite simple lookup table (Fig. 4) which performs the reduction process by direct mapping. That is, a block data with four sub-pixels located in each group will be transferred into a corresponding pixel. This means that each block data in group A will be mapped into a black pixel (gray-level 0), and a block data with four white sub-pixels in group C will be mapped into a white pixel (gray-level 255). Especially, each block data which contains one black sub-pixel and three white sub-pixels located in group B will be assigned to a gray pixel (gray-level here is 125). Suppose that the four inputs, a_1, a_2, a_3, a_4, in the block of each group represent either white or black gray-level, and the 3 outputs, 00, 10, 11, represent black, gray and white pixels respectively. Then, these 16 possible states in Fig.4 can be further mapped into 3 possible states. Referring to Fig. 4, let $f_1 f_2$ be the output bits, which is controlled by a_1, a_2, a_3, a_4. Then $f_1 f_2$ may be expressed as :

$$f_1(a_1,a_2,a_3,a_4)=a_1a_2a_3+a_2a_3a_4+a_1a_3a_4+a_1a_2a_4 \quad (12)$$

$$f_2(a_1,a_2,a_3,a_4)=a_1a_2a_3a_4 \quad (13)$$

Via the reduction process, the reconstructed watermark is reduced to the same size as the original watermark. Also, the reduced version of reconstructed watermark is more visible to human vision. Comparison of with-reduction and without-reduction is shown in Fig. 5. Since the reduction process mitigates noise effects caused by common image operations, the with-reduction result shown in Fig. 5(b) yields superior quality relative to the without-reduction result of Fig. 5(a).

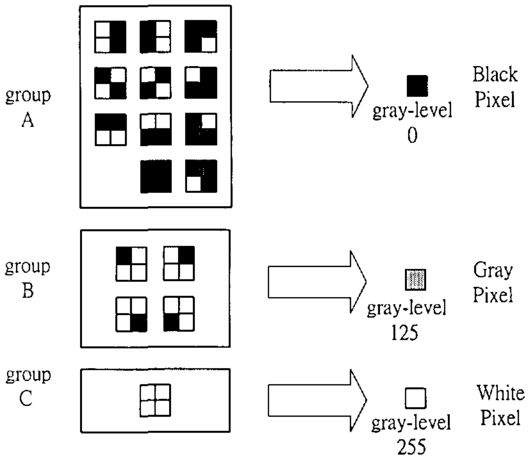


Figure 4: Reduction process lookup table for 4:1 data reduction rate.

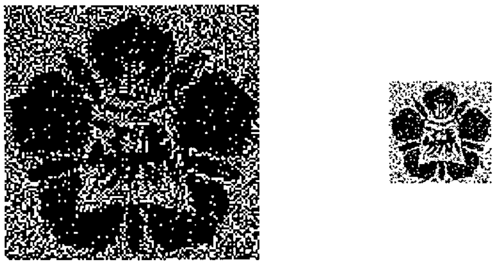

(a) (b)

Figure 5: Performance test for reduction process: (a) original reconstructed watermark; (b) reduced version of (a).

4 Simulations Results

To demonstrate the performance of the proposed scheme, watermarking is performed on the "Lena," "Baboon" and "Airplane" standard images, and the watermarked images are subjected to robustness and quality testing. The employed images are of size 512×512 pixels. The original watermark is of size 50 × 50 pixels. In the robustness test, comparison between extracted watermark and original watermark is made by unaided human vision. Further, we define Detection Rate (*DR*) as a quantitative measurement for evaluating extraction fidelity. If $n \times l$ is the size of the original watermark, then *DR* can be expressed as

$$\mathrm{DR} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{l} f(i,j)}{n \times l} \qquad (14)$$

where

$$f(i,j) = \begin{cases} 1, & if \quad w'(i,j) = w(i,j) \\ 1/2, & if \quad w'(i,j) = gray\ pixel \\ 0, & if \quad w'(i,j) \neq w(i,j) \end{cases} \qquad (15)$$

Here, $w(i,j)$ represents each pixel of the original watermark and $w'(i,j)$ represents each pixel of the reconstructed watermark. Gray pixels are given a 50% hit ratio when computing the Detection Rate. The quality of a watermarked image is estimated by using peak-to-peak signal-to-noise ratio (*PSNR*), expressed by:

$$PSNR = 10\log_{10} \frac{255^2}{\frac{1}{T}\sum_{\rho=0}^{N-1}\sum_{\sigma=0}^{N-1} (h(\rho,\sigma) - h_w(\rho,\sigma))^2} \quad (16)$$

where h(ρ,σ), $h_w(\rho,\sigma)$ represent each pixel of the host image and the watermarked image respectively.

4.1 Watermarked Image Quality

"Lena", "Baboon" and "Airplane" images, each of size 512×512 pixels with 256 gray levels, were used for testing. Figs. 6(a), 6(c) and 6(e) show the original images. Figs. 6(b), 6(d) and 6(f) show the watermarked images embedded with the public watermark. The PSNR for Figs. 6(b), 6(d) and 6(f) are 38.66 dB, 29.46 dB and 37.49 dB respectively, showing that these watermarked images retain reasonable quality.

4.2 Attack Testing

To test the robustness of the proposed watermarking scheme, we apply typical attacks such as common image processing and geometric manipulations. These attacks are performed using the commercial image-processing tool, Photoshop 5.0.

4.2.1 Lossy Compression with JPEG

JPEG lossy compression is a standard for still images. We applied it to the watermarked image to simulate an attack. Table 2 shows some experimental results including extracted watermarks, image quality and Detection Rate at different compression ratios. It can be seen that the proposed watermark can withstand JPEG attack at approximately CR=10, i.e. the reconstructed watermark can still visually identify ownership rights.

4.2.2 Filtering Operation with Blurring and Sharpening

In a poor transmission system, many image operations such as blurring and sharpening operations are used to enhance subjective quality. The coefficients used for the blurring filter and the sharpen filter are $\frac{1}{9}\times\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}$ and $\frac{1}{6}\times\begin{bmatrix}0&-1&0\\-1&10&-1\\0&-1&0\end{bmatrix}$, respectively. Figure 7(a) shows a blurred version of the watermarked image, and Fig. 7(b) shows the result from a version of the watermarked image blurred with a low-pass filter. Similarly, Fig. 8 shows the result obtained from a version of watermarked image sharpened with high-pass filter.

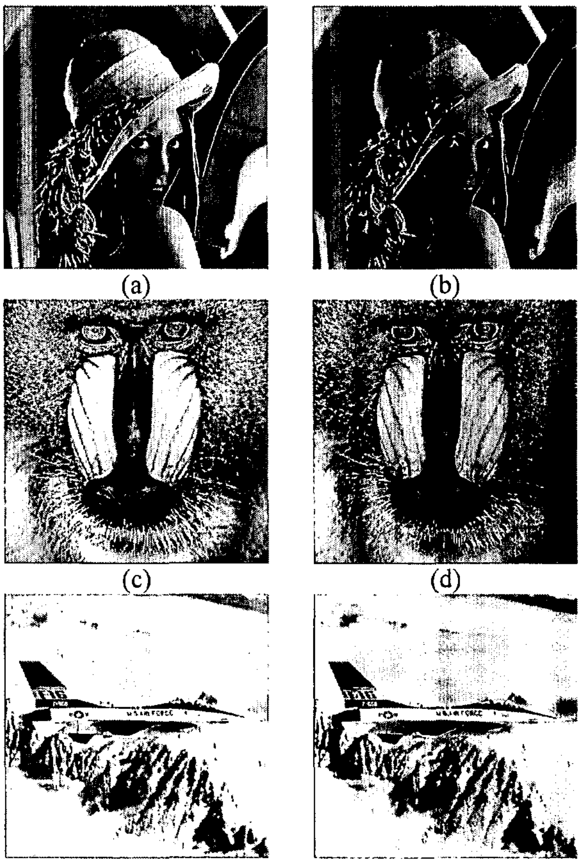
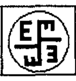
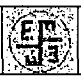
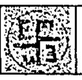
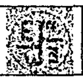
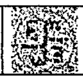
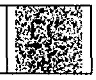

(a) (b)
(c) (d)
(e) (f)

Figure 6: Invisibility test for VSSW: (a), (c) and (e) are original images; (b), (d) and (f) are watermarked images; quality of (b), (d) and (f) are 38.66 dB, 29.46 dB and 37.49 dB respectively.

Table 2 Watermarked image Fig. 6(b) tested by JPEG attack.

Extracted image						

Ratio of JPEG	CR=1	CR=6.04	CR=8.02	CR=10.24	CR=12.00	CR=14.09
Watermarked quality(dB)	38.66	34.65	33.85	33.37	32.77	32.60
Detection Rate	100 %	88.72 %	82.59 %	80.69 %	76.97 %	71.37 %

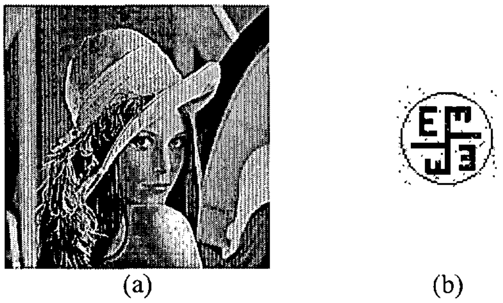

(a) (b)

Figure 7: Robustness test against blurring filtering for VSSW: (a) blurred image;(b) extracted watermark.

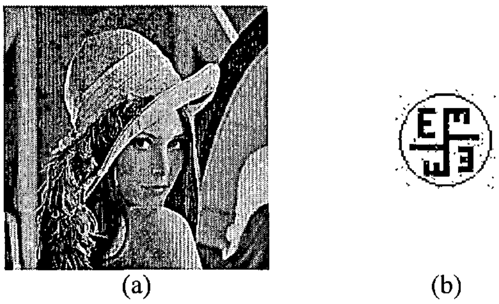

(a) (b)

Figure 8: Robustness test against sharpening filtering for VSSW: (a) sharpened image; (b) extracted watermark.

4.2.3 Cropping Attack

Fig. 9(a) shows a cropped version of the watermarked image of Lena. The watermarked image is clipped into a quarter of the original image size. Under these conditions, however, we can still clearly extract the watermark from the watermarked image. The result is shown in Fig. 9(b). Fig. 10(a) shows the interesting case where part of watermarked image is cut off by an image editor. However, the watermark shown in Fig. 10(b) can still be detected from the cropped version of the watermarked image.

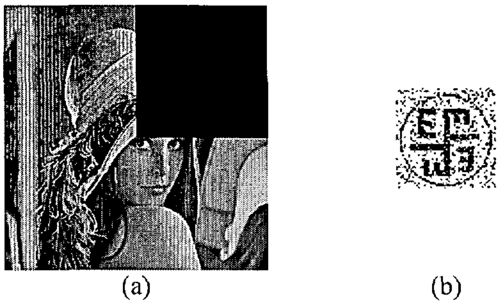

(a) (b)

Figure 9: Robustness test against image cropping for VSSW: (a) cropped image; (b) extracted watermark.

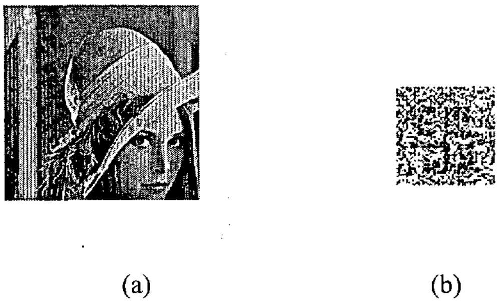

(a) (b)

Figure 10: Robustness test against image cropping for VSSW: (a) cropped image; (b) extracted watermark.

4.2.4 Rotation Attack

Fig. 11(a) shows a rotated version of the watermarked image. The watermarked image is first rotated two degrees by a rotation operation. Fig. 11(b) shows that despite the rotation, the extracted watermark is still acceptable to the human eye.

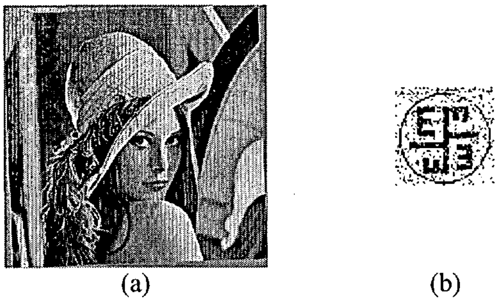

(a) (b)

Figure 11: Robustness test against image rotation for VSSW: (a) rotated image; (b) extracted watermark.

4.2.5 Rescaling Operation Attack

First, the watermarked image is reduced by a 2:1 resizing operation. Next, the image is enlarged to restore the image size. The result shown in Fig. 12(b) is extracted from the restored version of Fig. 12(a). Also, a 1:1.1 resizing operation is applied to the watermarked image, and the result is shown in Fig. 13. Again, the extracted watermark is acceptable.

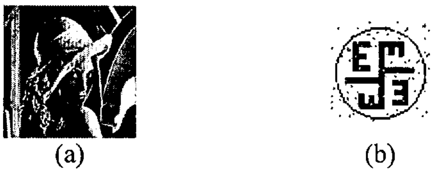


Figure 12: Robustness test against image rescaling (2:1) for VSSW: (a) rescaled image; (b) extracted watermark.

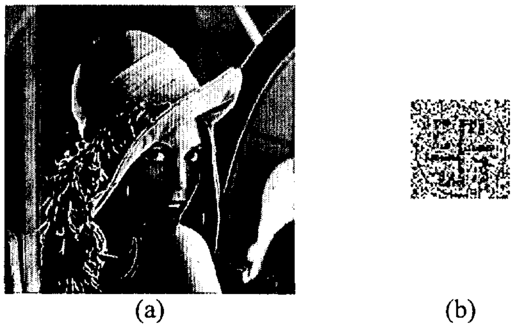


Figure 13: Robustness test against image rescaling (1:1.1) for VSSW: (a) rescaled image; (b) extracted watermark.

5 Conclusions

A secure and efficient watermarking system based on a VSS scheme has been proposed. The proposed system can embed watermark information into an image while maintaining good image quality. Without a valid secret watermark, it is virtually impossible to retrieve the watermark. Under this watermarking scheme, checking image ownership is only possible for someone who has possession of the correct secret watermark. This matches the cryptographic standard that a cryptosystem should be secure if someone knows the cryptographic algorithm used but does not have the appropriate key. The proposed technique thus achieves highly security. Furthermore, it is achieved by modification of only the coefficients in the middle band on full-frame DCT domain. Unlike some watermarking techniques, the proposed approach does not require an original image to extract the watermark. Thus, the proposed watermarking scheme may be applied easily in networks such as the Internet. Also, the proposed method can withstand various signal processing attacks, including lossy compression, sharpen filtering, blur filtering and image cropping. Especially, it achieves robustness with respect to the image rotation and image rescaling. For future work, we intend to extend the proposed process to compressed images by an advanced algorithm that will allow concurrent image compression and image watermarking.

References

[1]G. W. Braudaway, K. A. Magerlein & F. Mintzer, (1996), "Protecting publicly-available images with a visible image watermark," *Proc. of SPIE*, vol. 2659, pp. 126-133.

[2]I. J. Cox, J. Kilian, F. T. Leighton & T. Shmoon, (1997), "Secure spread spectrum watermarking for multimedia," *IEEE Trans. on Image Processing*, vol. 6(12), pp. 1673-1687.

[3]C. I. Podilchunk & W. Zeng, (May 1998), "Image-adaptive watermarking using visual models," *IEEE J. Sel. Areas in Comm.*, vol. 16(4), pp. 525-539.

[4]C. T. Hsu & J. L. Wu, (Sep. 1996), "Hidden signatures in images," *Proc. of ICIP'96*, vol. 3, pp.743-746.

[5]F. Hartung and M. Kutter, (Jul. 1999), "Multimedia watermarking techniques," *Proc. of the IEEE*, vol. 87(7), pp.1079-1107.

[6]D. Kundur & D. Hatzinakos, (1998), "Improved robust watermarking through attack characterization," *OPTICS EXPRESS*, vol. 3(12), pp.485-490.

[7]K. Hara, T. Shimomura & T. Hasegawa, (1988), "An improved method of embedding data into pictures by modulo masking," *IEEE Trans. on Comm.*, vol. 36(3), pp. 315-331.

[8]C. T. Hsu & J. L. Wu, (Aug. 1998), "Multiresolution watermarking for digital images," *IEEE Trans. on Circuits and Syst.-II Analog and Digital Signal Processing*, vol. 45(8), pp.1097-1101.

[9]R. Ohbuchi, H. Masuda & M. Aono, (1998), "Watermarking three-dimensional polygonal models through geometric and topological modifications, " *IEEE J. on Sel. Areas in Comm.*, vol. 16(4), pp.551-560.

[10] M. Maes, T. Kalkerr, J. Haitsma and G. Depovere, (1999), "Exploiting shift invariance to obtain a high payload in digital image watermarking," *IEEE Int. Conf. on Multimedia compu. and Syst.*, vol. 1, pp.7-12.

[11] J. Meng & S. F. Chang, (1998), "Embedding visible video watermarks in the compressed domain," *The Proc. of ICIP*, vol. 1, pp. 474-477.

[12] D. Anand & U.C. Niranjan, (1998), "Watermarking medical images with patient information," *Proc. of 20th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society*, vol. 20, pp. 703-706.

[13] A. Shamir, (1979), "How to share a secret," *Communications of the ACM*, vol. 22, pp. 612-613.

[14] N. Naor & A. Shamir, (1995), "Visual cryptography," *Advances in Cryptology Eurocrypt'94*, Springer-Verlag, Berlin, pp. 1-12.

[15] B. Schneider, (1996), "Applied cryptography," New York, Wiley.

Artificial Neural Networks Based Systems for Recognition of Genomic Signals and Regions: A Review

Vladimir B. Bajic[1,2], Suisheng Tang[1], Hao Han[1] and Vladimir Brusic[1],
1/ BioDiscovery Group, Laboratories for Information Technology, 21 Heng Mui Keng Terrace, Singapore 119613, Tel: +65-6874-8800; fax: +65-6774-8056, e-mail: bajicv@lit.a-star.edu.sg
2/ South African National Bioinformatics Institute (SANBI), University of the Western Cape, Private Bag X17, Bellville 7535, South Africa
AND
Artemis G. Hatzigeorgiou
Department of Genetics, Department of Engineering and Applied Science, Center for Bioinformatics, University of Pennsylvania, Philadelphia, USA.

Keywords: Artificial Neural Networks, Bioinformatics, Genomics

Received: April 18, 2002

In this review we present a number of important applications in computational genomics of a class of intelligent systems, namely artificial neural networks (ANNs). We present the current state-of-the-art solutions used in recognition of different genomic signals and regions. All systems to be commented are based fully or in part on the ANNs. We included systems that recognize different aspects of a/ transcriptional control information related to promoters, TATA-box regulatory region, and polyA signal, b/ those that relate to translation process comprising recognition of the translation initiation site, coding cDNA/EST fragments, reading frame-shift errors and their correction, and c/ splice-sites recognition. The review includes some of the most efficient systems for the indicated recognition problems in bioinformatics and aims to be an initial guide for those interested in these challenging problems.

1 Introduction

Bioinformatics is a complex and relatively new field of research which deals with the application of computational methods to the analysis of biological data (Attwood & Parry-Smith 1999, Baxevanis & Ouellette 2001, Mount 2001). A lot of biologically relevant information has been deposited in public databases and is available for scientific community (Attwood & Parry-Smith 1999, Baxevanis & Ouellette 2001, Mount 2001). However, generally speaking, there are no standards adopted. While all basic sequence data is obtained from experiments, the annotation of that mass of data is not always supported by experimental evidence and a lot is based on the computational analysis (Benson et al 2000). This, in addition to the incomplete understanding of the most of the biological processes (Hartwell et al 2000), complicates the application of computer methods for different analyses tasks.

Bioinformatics role is predominantly in extracting relevant information from the large quantity of biological sequence data, but also in producing hopefully accurate predictions of different important biological signals and regions contained in biological sequences, which can help in the sequence annotation and in reducing the quantity of target sequences for wet-lab analyses. Every signal and region recognition task in bioinformatics can be considered as a pattern recognition problem and handled by a variety of intelligent systems techniques. The first efficient use of artificial neural networks (ANNs) in genomics was the application of a perceptron for the recognition of start codons in *E.coli* (Stormo et al 1982), but applications exist that cater for the whole gene structure predictions (see, for example, Cai & Bork 1998, Snyder & Stormo 1993, 1995, Uberbacher & Mural 1991, Uberbacher et al 1996, Xu et al 1996, etc). In this review we will focus on several common computational genomics problems which are solved as pattern recognition tasks by ANNs. One of the challenges is the recognition of transcriptional control signals (Weinzierl 1999), such as transcription start sites (TSS), TATA-box, and transcription termination signals such as polyA signal (Zhao et al 1999). See also reviews in Fickett & Hatzigeorgiou (1997), Pedersen et al (1999). Another group of challenging problems are systems that deal with the translation process (Kozak 1999), including recognition of the start codon (translation initiation site – TIS) (Hatzigeorgiou 2002), coding measures required for recognition of coding exons, reading frame-shift errors and their correction (Hatzigeorgiou et al 2001). In addition we will present a system for recognition of splice-sites which separate exons and introns in eukaryotes. Our presentation includes up-to-date solutions used in this field and can help as a guide to computer science community for further exploration of this fascinating field.

2 Genomic Signals and Regions Included

Biological processes in cell of every living organism are orchestrated by complex regulatory mechanisms at different hierarchical levels (Hartwell et al 2000). DNA molecule contains inherited information that comprises signals required to control biochemical processes in the cell. DNA contains units of inheritance, genes, which determine to the greatest extent how the organism will develop and how it would respond under specific internal and external conditions (Hartwell et al 2000). To achieve its biological function, a gene has to pass several stages of biochemical processing. These are broadly characterized as transcription and translation (Hartwell et al 2000). In eukaryotes, initially, the gene segment of DNA is copied into the so-called pre-mRNA sequence in the process named transcription Weinzierl (1999). After the primary transcript is formed, it is further processed in the so-called RNA processing, and eventually, it is translated into the final gene product in a process named translation (Hartwell et al 2000).

We will consider here the recognition of several transcription control signals. Initiation of transcription is mainly controlled by a region called promoter (Weinzierl 1999). This region contains the so-called transcription start site (TSS) where the transcription starts. It also contains numerous transcription factor binding sites (TFBSs), short stretches of DNA with certain characteristic composition. Proteins called transcription factors (TFs) bind to promoter region to the binding sites in order to provide favorable environment for an RNA polymerase to initiate the transcription. One of the challenges is that there are no unique characteristics that can describe eukaryotic promoters, since TFBSs appear in different combinations at different mutual distances and in different orientation (Klingenhoff et al 1999, Zupicich et al 2001, Helhl & Wingender 2001, Kel et al 2001). Also, TF can bind to different TFBSs. At this moment there are more than 10,000 categorized TFBSs in TRANSFAC database (Wingender et al 2001), but it is reasonable to assume that there are much more that are not discovered yet. Some of the better known TFBS are TATA-box, GC-box, CCAAT-box, initiator (Inr), etc. (see Fickett & Hatzigeorgiou 1997, Pedersen et al 1999). Since eukaryotic promoters are short of common characteristics, they are very difficult to discover accurately by computational methods (Pedersen et al 1999, Fickett & Hatzigeorgiou 1997). We will present several systems based on ANNs that search for TSSs, TATA-box, and polyA signal in anonymous (non-annotated) DNA sequences.

When the transcript is formed from the template DNA, it is further subjected to RNA processing and translation. RNA processing will transform the primary transcript by splicing out sections called introns (which are internally bounded by the splice-sites) and will perform some additional alterations at the ends of the transcript, forming the mRNA. This mature RNA is required to generate the final gene products through the translation process. The final gene products are peptides/proteins required by the biochemistry of the cell. Translation is a complex process which essentially uses information contained in the so-called open reading frame (ORF) which is a section of the mRNA that starts with the so-called start codon, and terminates with the stop codon. Codons are groups of three consecutive nucleotides that are converted into one amino acid during the translation process. The length of the ORF is divisible by three, since it contains consecutive codons. The first nucleotide of the start codon is called translation initiation site (TIS) (Hartwell et al 2000). The ORF part of the mRNA is biased and can be detected by assessing the so-called coding measures (see Hatzigeorgiou et al 2001, Hatzigeorgiou 2002) of this region. If the number of nucleotide deletion or insertion in ORF has remainder after divided by three, the translation template will be changed and will cause frame-shift. These would result in the synthesis of wrong final product of the gene, which may be fatal for the organism. Since in the biological databases there is a limited amount of data that have proper experimental verification, it is of interest to be able to detect computationally the frame-shift errors and when possible to correct them.

We will present several systems that use ANNs in the detection of the TIS, frame-shift errors and their correction, ORFs in the mRNA or cDNA sequences, and splice-sites.

3 Systems which Recognize Control Signals for Transcription Process

Roughly we can classify transcription control signals as those related to initiation of this process and termination of it. The transcription initiation signals are associated with promoters. Recognition of the promoter region and its signals is one of the most difficult problems in computational genomics. There have been several systems developed to attack this problem with different degrees of success. For the reviews and comparisons see Fickett & Hatzigeorgiou (1997), Reese et al (2000), Bajic et al (2002b), Bajic (2000), Scherf et al (2000). We will present here eight systems which use ANNs as a part of the solution. In what follows we use three measures to describe the system performance: sensitivity Se = TP/(TP+FN), specificity Sp = TN/(TN+FP), and positive predictive value ppv = TP/(TP+FP), where TP, TN, FP, and FN denote the number of true positive, true negative, false positive, and false negative predictions, respectively. In addition, we also consider the frequency of prediction in DNA sequence.

3.1 Grail's Promoter Recognition Module

Grail program (Xu et al 1996, Matis et al 1996) is a gene recognition system. It has a module that can recognize some types of promoters. Grail recognizes mainly TATA-box containing promoters and it has sensitivity of 0.66 for such promoters making one prediction per approximately 23000 nt ('nt' stands for nucleotide). This system uses a feedforward ANN and input signals from five sensors: TATA-box, GC-box, CCAAT-box, Inr, TIS, as well as distance information between these signals. It also employs a set of refined rules associated with the distances of important signals to reduce the level of false positive predictions. The schematic diagram is depicted in Fig. 1.

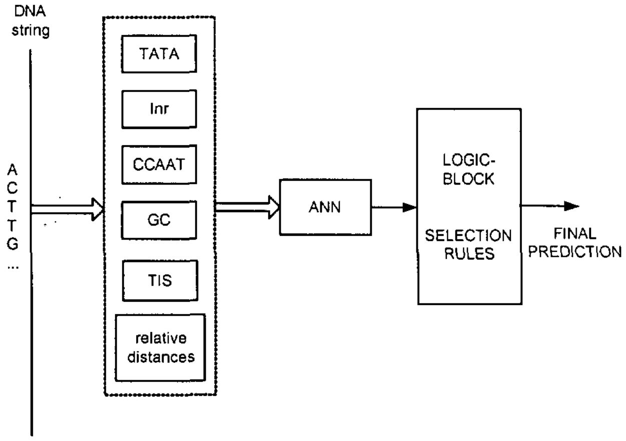


Figure 1: The structure of the Grail's promoter prediction module

3.2 McPromoter ver.3

McPromoter is a system that uses integration of certain physical properties of DNA, promoter region segmentation, and an ANN to produce prediction of promoters (Ohler et al 2001). Promoter region is divided into six sub-regions, two of which corresponding to the core promoter. These six regions correspond to the two upstream promoter segments, TATA-box, spacer between the TATA-box and Inr, Inr element, and downstream promoter element (Ohler & Niemann 2001, Ohler et al 2001). These segments represent the states in the stochastic segmentation model of the promoter region and each of the states is modeled by an interpolated Markov model (Ohler & Niemann 2001, Ohler et al 2001). Segment probabilities are calculated as joint probabilities of sequences and physical profiles. Physical profiles are derived for a number of features including GC content, DNA bendability, DNA conformation, etc. As background models, coding and non-coding sequences were used. Segment probabilities and the likelihoods of promoter and background sequences are used as inputs to the feedforward ANN to generate the final predictions. The system is developed for invertebrates and makes approximately one prediction per 3000 nt, at the sensitivity of 0.36.

3.3 NNPP System

The NNPP program (see Reese et al 1996, Reese 2001) is based on the recognition of two specific signals within the core promoter region: the TATA-box and Inr, as well as their mutual distance. This system uses three time-delay ANNs (see Fig. 2). One ANN recognizes TATA-box, the other recognizes Inr. The third time-delay ANN combines the outputs of the previous two ANNs with the spatial distance between the TATA-box and Inr signals. The system produces one prediction per 550 nt at the sensitivity of about 0.75.

Another system based on a similar architecture was developed by Mache et al (1996) and achieved sensitivity of 0.5 with the average frequency of one prediction per 3100 nt. It has been shown in another report (Hatzigeorgiou et al 1994) that better performance of promoter recognition can be obtained with the feedforward networks than with the time-delay neural networks.

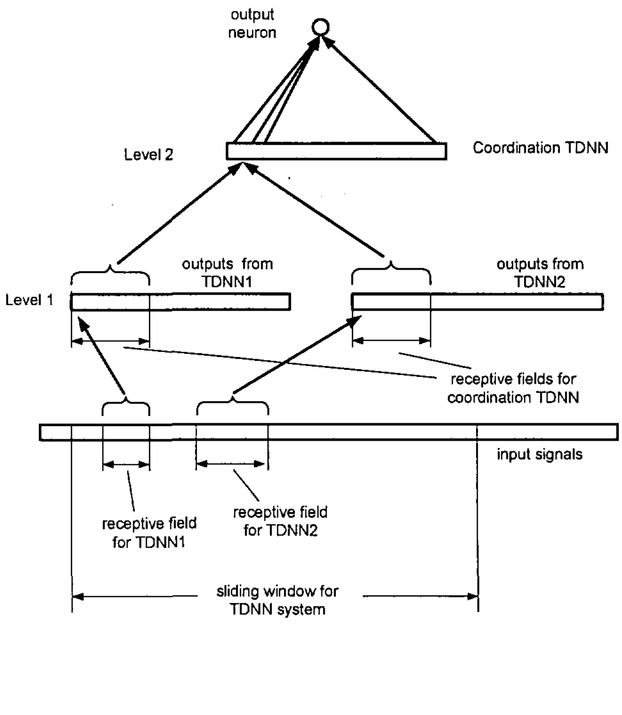

output neuron
Level 2
Coordination TDNN
outputs from TDNN1
outputs from TDNN2
Level 1
receptive fields for coordination TDNN
input signals
receptive field for TDNN1
receptive field for TDNN2
sliding window for TDNN system

Figure 2: The structure of the NNPP system

3.4 Promoter 2.0

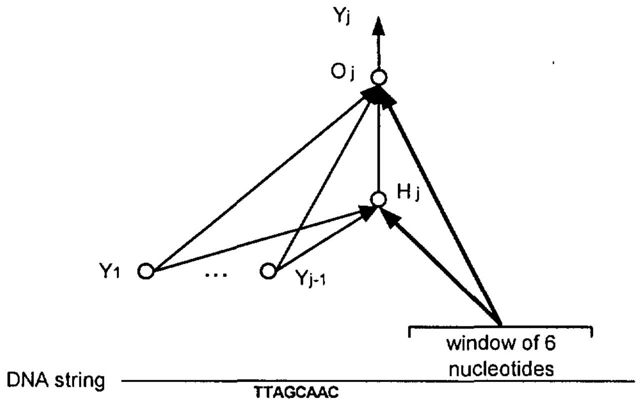


Figure 3: The structure of one of the layers in the Promoter2 system

Promoter 2.0 (Knudsen 1999) uses four ANNs to recognize promoter. The ANNs are employed to model TATA-box, GC-box, CCAAT-box and Inr, as well as distances between these elements. The ANNs are connected in a special hierarchical manner, so that the output of one ANN serves as an additional input for the higher level ANN (see Fig. 3 for one layer structure). Due to the unusual structure, the conventional backpropagation and similar training algorithms cannot be used directly. The system is trained with a simplified version of a genetic algorithm. It uses a window of 6 nt in the scanning process along the DNA sequence.

3.5 SPANN2 System

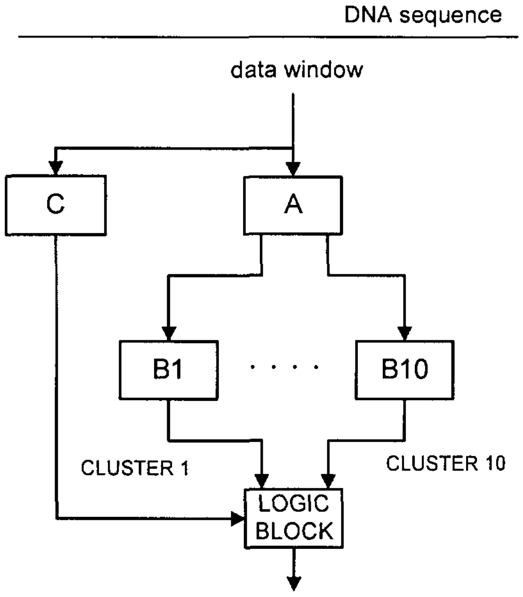


Figure 4a: Structure of the SPANN2 system

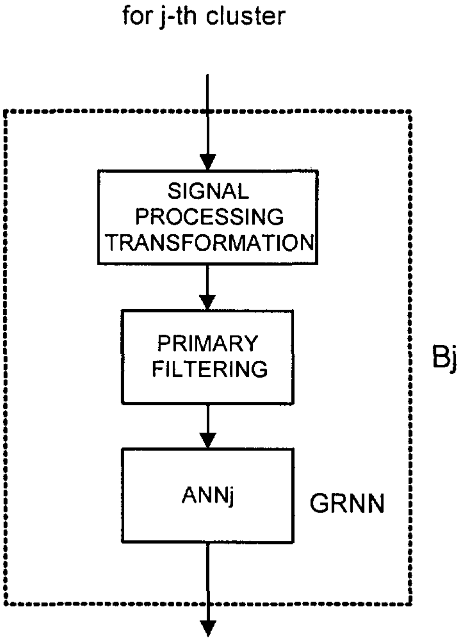


Figure 4b: Information processing in Bj block which corresponds to the j-th data cluster

SPANN2 system (Bajic & Bajic 2000) uses domain transformations to convert the primary DNA information into more convenient form for ANN processing. The system uses 11 ANNs: one SOM ANN which partitions input data into 10 clusters, and one generalized regression ANN for each of the clusters to make the final prediction. On the test-set of Fickett & Hatzigeorgiou (1997) the system exhibits a sensitivity of 0.33 and 16 false positive predictions which is a favorable performance on this test set. The structure of the system and information processing is depicted in Figs. 4a-c.

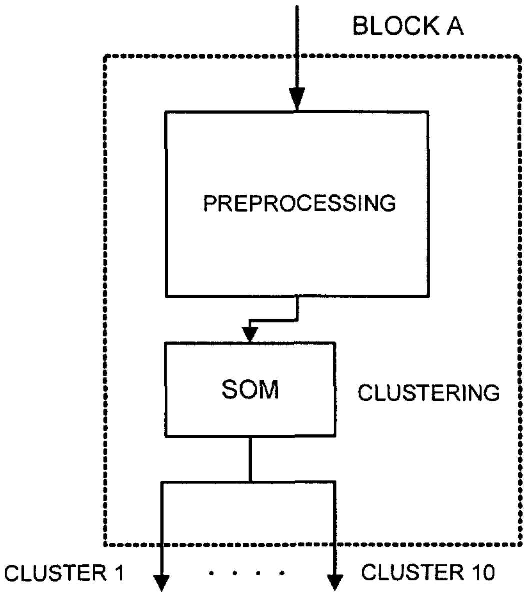


Figure 4c: Information processing in block A

3.6 LVQ Networks for TATA-box Recognition

The system of Wang (2001) (Fig. 5) uses two LVQ ANNs in the process of recognition of TATA-box. The system considers 8 derived features from the TATA-box and its neighborhood. These features are used in the preliminary filtering process which reduces the number of false positive predictions. This filtering is based on the analysis of a number of statistical properties. For the sequences that pass the filter, data compression is applied that reduces the dimension of the transformed feature space to 6. This serve as input data for the LVQ system that uses two LVQ networks and a set of rules to further reduce the false positives. The system is trained by a genetic algorithm due to combination of rules and two LVQ networks. The system achieves sensitivity of 0.33 and 47 FP on the dataset from Fickett & Hatzigeorgiou (1997), which can be considered favorable on the basis of comparison measures used in Bajic (2000).

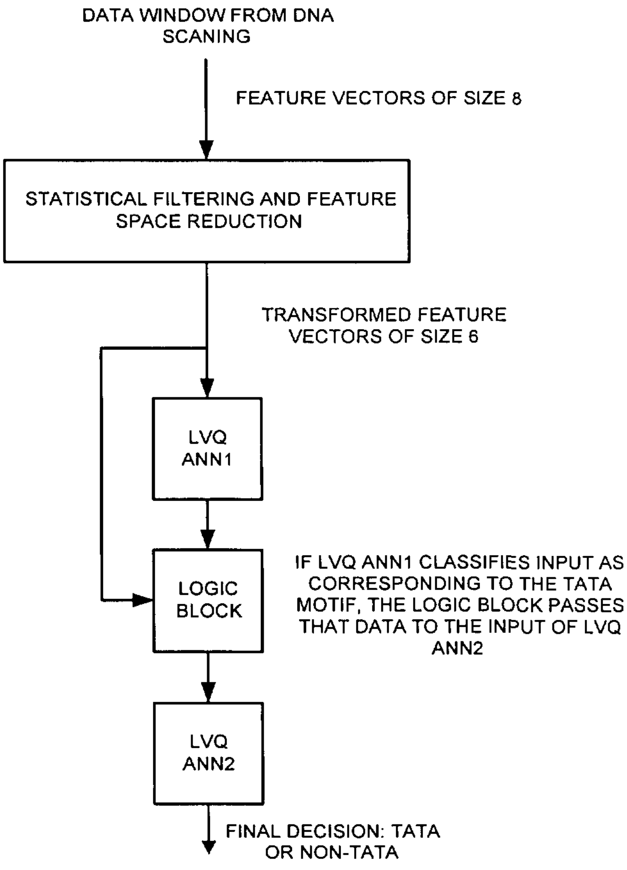

DATA WINDOW FROM DNA SCANING
FEATURE VECTORS OF SIZE 8
STATISTICAL FILTERING AND FEATURE SPACE REDUCTION
TRANSFORMED FEATURE VECTORS OF SIZE 6
LVQ ANN1
LOGIC BLOCK
IF LVQ ANN1 CLASSIFIES INPUT AS CORRESPONDING TO THE TATA MOTIF, THE LOGIC BLOCK PASSES THAT DATA TO THE INPUT OF LVQ ANN2
LVQ ANN2
FINAL DECISION: TATA OR NON-TATA

Figure 5: Structure of the Wang's system for TATA-box recognition

3.7 Hamming Networks and Recognition of TATA-box and PolyA-signal

The systems developed by Milanesi et al (1996) use Hamming clustering method to partition the input training space to clusters, so that for each cluster a specific representative prototype is selected. The input signals for the TATA-box motif and polyA-signal motifs are given in a special binary form, and Hamming distance is used in the clustering process. After the initial clustering of the input data, the feedforward ANN is trained with the representative cluster prototypes and with negative data to produce ANN that will recognize the required type of input signal.

3.8 Dragon Promoter Finder ver.1.3

Dragon Promoter Finder (DPF) is an enhanced integrated promoter recognition system (see Bajic et al 2002a,b,c). It uses a composite-hierarchical approach, artificial intelligence, statistics, and signal processing techniques. It separates promoter sequences to those that are G+C-rich or G+C-poor. The system structure is represented in Figs. 6a-b. There are separate models for different expected sensitivity levels (Fig. 6a). Each model is optimized for a particular sensitivity level to minimize the number of FP predictions. Also, models are derived separately for G+C-rich and G+C-poor sequences. Within a particular model three sensors for promoter, intron and coding exon sequences compete and an ANN determines if the input sequence corresponds to the promoter or not. DPF allows scanning of complete vertebrate genomes for promoters with significantly reduced number of false positive predictions (see Bajic et al 2002a,b,c). It can be used in combination with the gene finding programs for more accurate prediction of the 5'end of genes. The system was evaluated on a large and diverse human sequence-set and exhibited several fold less false positive predictions at the same level of sensitivity than several publicly available TSS-finding programs. Results obtained using human chromosome 22 data showed even greater specificity than the evaluation set results. The system has been implemented in the Dragon Promoter Finder package, which can be accessed at http://sdmc.krdl.org.sg:8080/promoter/. The system is capable of successfully recognizing broad classes of promoters, regardless of whether they are CpG-island related or not, or whether they are in the G+C-rich and G+C-poor regions. This makes it quite universal as opposed, for example, to solutions in Hannenhalli & Levy (2001), Ioshikhes & Zhang (2000) which are specialized in recognizing CpG-island related promoters, or TATA-box containing promoters in G+C-rich regions (Down & Hubbard 2002). Based on the results on the mouse genome

analysis, the average frequencies of predictions based on several sensitivity settings for DPF are summarized in Table 1. The predictions that were within 1000 nt apart were combined into one prediction.

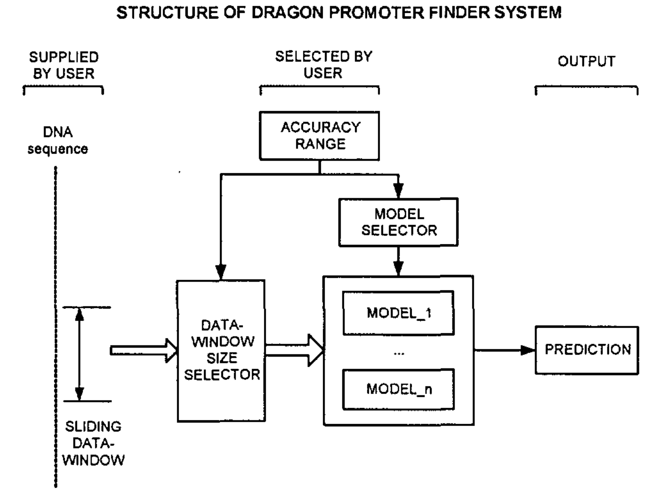


Figure 6a: Dragon Promoter Finder system structure

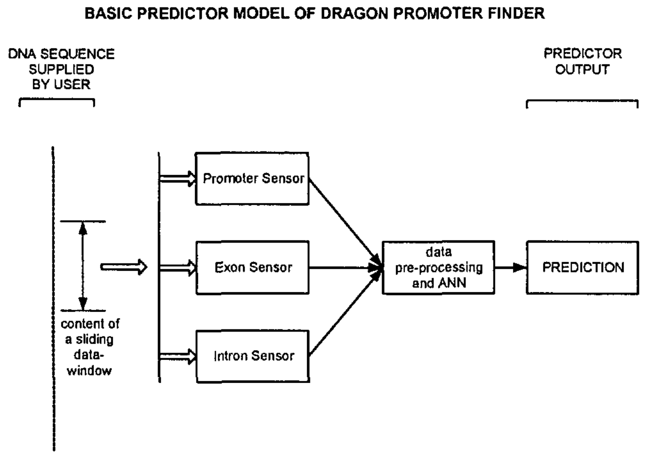

BASIC PREDICTOR MODEL OF DRAGON PROMOTER FINDER
DNA SEQUENCE SUPPLIED BY USER
PREDICTOR OUTPUT
Promoter Sensor
Exon Sensor
Intron Sensor
data pre-processing and ANN
PREDICTION
content of a sliding data-window

Figure 6b: Structure of one of the models in DPF

Testing results of the DPF ver.1.3 system on the annotated known genes on human chromosome 22 are summarized in Table 2. The annotation data (Rel.2.3) for human chromosome 22 were produced by the Chromosome 22 Gene Annotation Group at the Sanger Centre and were obtained from the World Wide Web (Dunham et al.). It should be noted that human chromosome 22 is very G+C rich and consequently atypical for human genome, and thus the performance of prediction programs on this chromosome cannot serve as a basis for assessment of the general prediction ability. However, it is good to use it as a relative reference source. We considered the predicted TSS positions correct if they fall within the interval of length equal to the average length of the promoter region predicted by PromoterInspector (555bp), while the other conditions were the same as in Scherf et al (2001). If the predictions were within 1000 nt apart they were combined in one prediction. For example, Table 2 implies that for each TP at Se = 80%, an average of 3.34 FP predictions were made; at Se = 0.58%, 1.4 FP predictions were made; and at Se = 0.49% on average 1.1 FP predictions were made.

TABLE 1 Average predictions of DPF ver.1.3 on the non-masked version of the Mouse Genome Assembly (v 3 dated May 01, 2002)

Expected Se	Frequency of predictions	Total number of predictions
50%	40957 nt	135377
65%	16433 nt	337399
80%	4973 nt	1114806

TABLE 2 The results of promoter prediction by DPF ver.1.3 on human chromosome 22.

Se	ppv	Cost of making one TP prediction, i.e. FP/TP
49%	48%	1.1
58%	42%	1.4
64%	33%	2.0
74%	30%	2.34
80%	23%	3.42

4 Recognition of Translation Related Signals and Regions

The problem of TIS recognition appears in different contexts, from the recognition of TIS in anonymous DNA (Derst et al. 2000), to its recognition in mRNA, cDNA or EST fragments (Pedersen & Nielsen 1997, Salzberg 1997, Burge & Karlin 1997, Agarwal & Bafna 1998a,b, Salamov et al 1998, Zien et al 2000, Nishikawa et al 2000, Hatzigeorgiou 2002, Hatzigeorgiou et al 2001). Also, most of the gene recognition programs have modules that aim at recognizing TIS. In a vast majority of cases (Kozak 1999) TIS conforms the well-known 'ATG' consensus in DNA ('AUG' in mRNA). On average, this consensus appears once in 64 nt on a completely random DNA.

The recognition of TIS can be efficiently set up if the ribosome scanning model (selection of the most 5' AUG in the proper context in mRNA as the initiation codon, Kozak 1999), is used in the recognition process such as in Hatzigeorgiou (2002). Unfortunately, there are many mRNA/cDNA sequences that have long 5'UTR which can contain other upstream ATGs (upstream of the main TIS), upstream stop codons, or even complete upstream ORFs (Kozak 1999). This makes recognition of the functional TIS far more difficult. The recognition problem is even more complicated by having partial cDNA or mRNA sequences, those that contain incomplete main ORF. The ability of computational prediction of TISs is hampered in such cases.

When the length of the 5'UTR is greater, the possibilities increase that such sequences could contain AUG triplets, stop codons or the 'so-called' mini ORFs. This opens ways for violating the ribosome scanning rule (Kozak 1999) and generally allows several scenarios for a downstream AUG to be used instead of the first AUG. These phenomena are known as leaky scanning, reinitiation and direct internal initiation. Without entering into details of these phenomena (Kozak 1999) we point out that such possibilities are considerably reduced in the case of short 5'UTRs. The computational problem is significant when 5'UTRs are longer and this motivated Salamov et al (1998) to develop a method (ATGpr) for TIS recognition that can cope with such a situation.

Several techniques have been employed in attempts to develop efficient method for computer-based recognition of TIS. These include support vector machines (Zien et al 2000), linear discriminant analysis (Salamov et al 1998), different ANNs (Hatzigeorgiou 2002, Hatzigeorgiou et al 2001, Derst et al 2000, Pedersen & Nielsen 1997), generalized higher-order profiles (Agarwal & Bafna 1998a,b), positional conditional probability matrices (Salzberg 1997) and other statistical techniques (Burge & Karlin 1997, Nishikawa et al 2000). Different sequence features were employed in these recognition methods, and also different lengths of TIS neighborhood are used to extract these features. The best reported results on TIS recognition in cDNA (Hatzigeorgiou 2002) indicate sensitivity and positive predictive value of 0.94. Unfortunately, that method cannot be applied to partial cDNA since it uses a variant of the ribosome scanning model which essentially requires a complete ORF. Other methods, such as ATGpr (Salamov et al 1998), WWW_Evaluator (Rogozin et al 1996) and NetStart (Pedersen & Nielsen 1997) do not rely on the ribosome scanning model and do not require a full length ORF in cDNA fragments.

The most direct way to characterize the coding regions of genomes and provide reliable information for structural annotation of genes in genomic sequences still remains the analysis of sequences from cDNA libraries. As a consequence of their contribution to rapid gene discovery, full and partial cDNA sequences have been generated in very large numbers, both in public and private sectors. ESTs make up currently more than 60% of all the database entries and EST sequencing projects have already started to have a major impact on biomedical research, by accelerating the identification of new genes of interest as potential targets for drug discovery, and by providing target sequences for genome wide expression profiling.

Unlike high quality finished genome sequences, which are double-strand and multiple-pass, cDNA and EST sequences are mostly single-strand, single-pass sequences which frequently contain sequencing errors. Errors may result in nucleotide substitutions, insertions or deletions, leading to frame-shifts which cause problems in the analysis of these sequences. The analysis of ESTs is further complicated by the fact that they are usually short, 300-600 nucleotides in length, they originate from different parts of the cDNA, and may include only sequence of non-translated regions.

4.1 Translation Initiation Site

There are several systems used for recognition of the TIS in cDNA based on the use of ANNs.

4.1.1 Diana-TIS

The system Diana-TIS (Hatzigeorgiou 2002) uses two ANNs. The first one, the consensus ANN, assesses the TIS and its immediate surrounding (see Fig. 7a). The second one, the coding ANN, assesses the coding potential of the regions upstream and downstream of TIS. Several approaches have been investigated to find a good method for coding region prediction. The best performance is obtained by an ANN that uses preprocessed data as input. The most efficient preprocessing method was the counting of codons in a window using a step of three nucleotides and starting with the window's first codon (see Fig. 7b). Negative examples are extracted randomly from the non-coding regions and also from those windows of the coding region that start with the second and third nucleotide of a codon (i.e., those that are out of frame). The finally assembled system uses these two networks and several additional rules to predict the TIS. The consensus ANN is with short-cut connections, has two hidden neurons, and it is trained by cascade-correlation algorithm. The coding ANN is a feedforward network trained by resilient backpropagation.

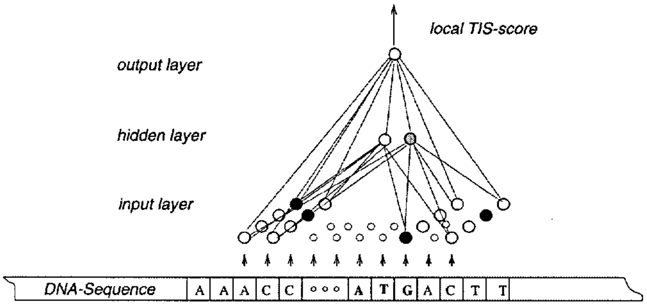


Figure 7a: shows the architecture of the implemented TIS recognition module. A sliding window of 12 nucleotides is presented to the trained ANN. A high score at the output of the ANN indicates a possible TIS.

Abilities of individual ANNs and the integrated method to correctly recognize TIS are summarized in Table 3. Both Se and ppv are obtained from the results on the positive and negative test sets.

TABLE 3. Performances of three ANNs used

	(Se+ppv)/2
Coding ANN	0.764
Consensus ANN	0.825
Integrated method	0.94

The system is available under: http://diana.pcbi.upenn.edu

4.1.2 NetStart

The NetStart system (Pedersen & Nielsen 1997) predicts TIS by an ANN. The input to the ANN is the nucleotide sequence translated through the binary code. Data window is of 203 nucleotides in length and a feedforward three layer ANN is used with 30 hidden neurons. The system achieves Se = 0.78 and Sp = 0.87 and can be accessed at http://www.cbs.dtu.dk/services/NetStart/

4.2 Coding Regions and Frame-shift

DIANA-EST system (Hatzigeorgiou et al 2001) is aimed at recognition of the coding or non-coding EST/cDNA sequences, reading frame-shift errors and corrections of some of these. It uses three ANNs.

DIANA-EST is based on a combination of ANNs and statistics for the characterization of coding regions within ESTs. Two major problems in the analysis of ESTs are their short length and frequent sequencing errors. Their short length makes it difficult to use non-frame-specific coding measures and, conversely, frame-specific methods suffer from the frequent frame shifts introduced by sequencing errors. To overcome these problems DIANA-EST incorporates two separate modules: a coding/non-coding sensitive module and a frame/non-frame sensitive module. Both modules are based on frame-specific codon usage statistics combined with an ANN (Hatzigeorgiou, 2001).

The selection of positive and negative data was made similarly as in the case of the DIANA-TIS system. The performance of this particular ANN on human sequence has an accuracy characterized by (Se+ppv)/2 = 84% (Hatzigeorgiou et al 2001).

For the prediction of the coding regions in EST's all three modules are integrated into one system. In the first step the frame-ANN is applied on a sliding window along the sequence. If the sequence is derived from a coding region without sequencing errors the output will be a sequence of numerical values with a high score in every third position. This is the position of the first nucleotide of a codon. If a deletion or insertion occurs, this periodicity will get disturbed. In the ideal case the response of the ANN in scanning of a coding nucleotide sequence will be made of a chain of alternations of 1 0 0, starting with 1 (for example: 1 0 0 1 0 0 1 0 0.....1 0 0 1 0 0). In the second step of the algorithm the ideal chain of the sequence gets aligned with the real score values in order to maximize the overall coding score potential (which, in this case, is calculated by

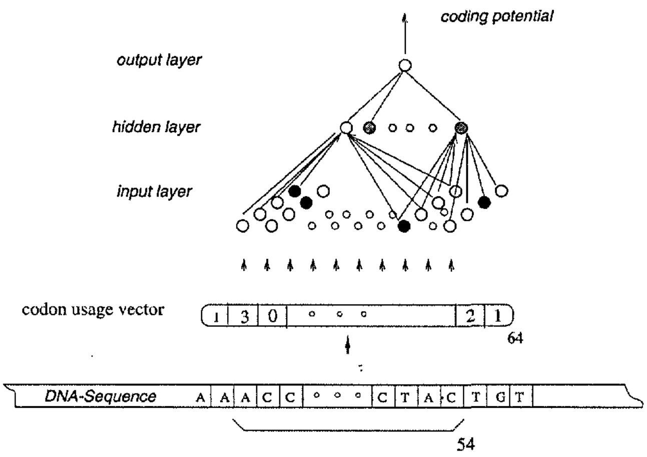


Figure 7b: shows coding region recognition module. A window of 54 nucleotides slides along the sequence. After preprocessing, the data of each window are presented to the trained ANN. A high ANN output indicates a window of a coding region starting on the first nucleotide of a codon.

The selection of positive and negative data was made similarly as in the case of the DIANA-TIS system. The performance of this particular ANN on human sequence has an accuracy characterized by (Se+ppv)/2 = 84% (Hatzigeorgiou et al 2001).

For the prediction of the coding regions in EST's all three modules are integrated into one system. In the first step the frame-ANN is applied on a sliding window along the sequence. If the sequence is derived from a coding region without sequencing errors the output will be a sequence of numerical values with a high score in every third position. This is the position of the first nucleotide of a codon. If a deletion or insertion occurs, this periodicity will get disturbed. In the ideal case the response of the ANN in scanning of a coding nucleotide sequence will be made of a chain of alternations of 1 0 0, starting with 1 (for example: 1 0 0 1 0 0 1 0 0.....1 0 0 1 0 0). In the second step of the algorithm the ideal chain of the sequence gets aligned with the real score values in order to maximize the overall coding score potential (which, in this case, is calculated by multiplying the values of the two sequences). This alignment is made by a dynamic programming approach. Finally, retracing back the path of the best alignment (the alignment with the best score) it is possible to locate the frame changes. This method is similar with the approach described in (Xu et al 1995). In order to avoid a frequent frame-switching it is necessary to introduce frame-change penalty.

For the recognition of the coding start the consensus-ANN is used to calculate a score for every ATG (putative TIS). In Addition the non-coding/coding potential around every ATG is calculated by building the difference between all coding scores (calculated by the coding ANN) of in-frame 60 positions before and after the ATG. If the product of the consensus score and the non-coding/coding difference is above a certain threshold (here 0.2) and the ATG is on the leading frame, the ATG is characterized as TIS.

Stop codons are permitted on the predicted coding region. Possible ends of the coding sequence are determined by the presence of stop codons in a local coding frame. Local coding frame means that the frame of the stop-codon has a high score for 60 nucleotides before the stop codon.

The integrated method of detection of coding/non-coding sequences has Se = 0.865 and ppv = 0.796. Correction ability of the integrated system is estimated to be characterized by (Se+ppv)/2 = 0.897.

The system is available under: http://diana.pcbi.upenn.edu

5 Splice Sites Recognition

Splice sites represent boundaries between the introns and exons in eukaryotic organisms. They are located in introns and the boundary between the exon and the adjacent intron is characterized by a canonical nucleotide pair GT and called 'donor' site, while the boundary between the intron and the adjacent exon is characterized by AG nucleotide pair and it is called 'acceptor' site. Consequently introns are bounded by GT and AG splice sites. Since many of the internal exons are also coding exons, a nucleotide bias is present in such coding exons and this frequently can help in predictions of the splice sites.

5.1 NetPlantGene

One of the systems for splice sites prediction based on ANNs is NetPlantGene (Hebsgaard et al 1996). This system uses ANNs and additional rules to infer the predictions of splice sites. It combines local and global sequence information. ANNs used in this system are all feedforward, three layer ANNs, trained by backpropagation algorithm. The input sequence is obtained from appropriate nucleotide data window, where nucleotides are coded by a sparse binary code A = (1000), C = (0100), G = (0010), T = (0001). Each ANN has only one output node which produces real number response in the range [0,1]. Two types of ANNs are designed for this system: 1/ ANNs for detection of coding nucleotides as opposed to non-coding ones, and 2/ ANNs used for detection of splice sites. The coding/non-coding ANNs are trained to achieve maximal correlation coefficient (CC). The splice sites ANNs are trained to achieve Se = 0.95 while the number of FPs is minimized. Data window used for splice site ANNs is 23 nt, and 10 hidden layer neurons were used. An ensemble of 10 ANNs which were randomly initialized is formed and the average of the outputs of ensemble ANNs is used for predictions.

On *A.thaliana* dataset this ensemble achieves Se = 0.61, ppv = 0.69 for donor sites. For acceptor sites the performance is similar, but due to complexity of acceptor sites a window of 61 nt is used and all ANNs in the ensemble of 10 ANNs had 15 hidden layer neurons. An ensemble of coding/non-coding ANNs is formed from 6 ANNs. One ANN uses data window of 101 nt, 4 ANNs use data window of 201 nt, and one ANN a data window of 251 nt. These different window sizes are used to cater for short and long exons. This ensemble achieves Se = 0.91, Sp = 0.895. One may consider the splice site ensemble as dealing with the local information, while the coding/non-coding ensemble deals with the global information surrounding the sites. These two ANN ensembles are combined into the final system and additional rules are used for post-prediction filtering. The final system achieves the following results on *A.thaliana* data set:

5.2 Other Splice Site Systems Based on ANNs

Several other systems for splice site recognition based on different types of ANNs are developed. Generally, their technical solutions exploit the coding/non-coding region contrast and immediate surrounding of the splice sites. The best known of these systems are NetGene2 (http://www.cbs.dtu.dk/services/NetGene2/) which evolved from NetGene (Brunak et al 1991) and NetPlantGene, NNSplice (http://www.fruitfly.org/seq_tools/splice.html), BRAIN (Rampone 1998) and EXON-ENet (Fu 1999). A comparison of the many splice site recognition systems including some of those based on ANNs are given in the study of Thanaraj (2000).

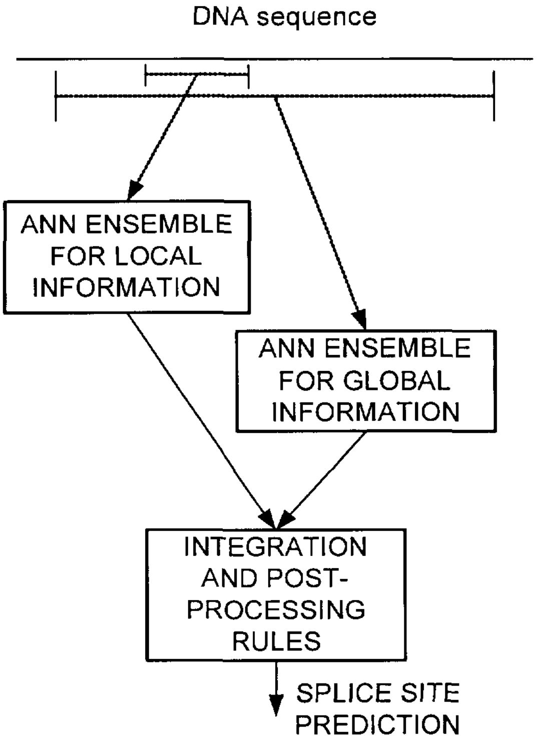

DNA sequence
ANN ENSEMBLE FOR LOCAL INFORMATION
ANN ENSEMBLE FOR GLOBAL INFORMATION
INTEGRATION AND POST-PROCESSING RULES
SPLICE SITE PREDICTION

Figure 8: NetPlantGene system structure

TABLE 4. Performances of NetPlantGene ANN ensembles

	Se	FP
Integrated method (donor sites)	0.80	0.011%
Integrated methods (donor sites)	0.95	0.097%
Integrated method (acceptor sites)	0.80	0.034%
Integrated methods (acceptor sites)	0.95	0.26%

6 Conclusion

We presented an overview of a number of ANN based systems used for different problems associated with transcriptional and translational signals and regions, as well as splice-sites. These systems show overall improved performances as compared to other non-ANN based systems and thus represent very promising tools for specific pattern recognition in computational genomics. The indicated performances of these systems can be used only as a guideline since the training and test sets for these systems were, generally, different. Interested reader can find details of the solutions in the respective literature.

References

[1] P. Agarwal, V. Bafna (1998a) Detecting nonjoining correlations within signals DNA. In *Proceedings of the 2nd Annual International Conference on Computational Molecular Biology*, RECOMB98. ACM Press, pp.1-7.

[2] P. Agarwal, V. Bafna (1998b) Translation initiation: implications for gene prediction and full-length cDNA. In *Proceedings of the 5th International Conference on Intelligent Systems for Molecular Biology*, ISMB98. AAI Press, pp.2-7.

[3] T. K. Attwood, D. J. Parry-Smith (1999) *Introduction to Bioinformatics*, Addison Wesley Longman Ltd., Essex, UK

[4] V. B. Bajic (2000) Comparing the success of different prediction software in sequence analysis: A review, *Briefings in Bioinformatics*, Vol. 1, No. 3, pp. 214-228.

[5] V. B. Bajic and I. V. Bajic (2000) Neural network system for promoter recognition, Chapter 14 in *Future Directions for Intelligent Systems and Information Science* (Nik Kasabov, Ed.), pp. 288-305, Physica-Verlag, New York.

[6] V. B. Bajic, S. H. Seah, A. Chong, G. Zhang, J. L. Y. Koh, V. Brusic (2002a) Dragon Promoter Finder: recognition of vertebrate RNA polymerase II promoters, *Bioinformatics*, 18(1):198-199.

[7] V. B. Bajic, A. Chong, S. H. Seah, V. Brusic (2002b) Intelligent System for Vertebrate Promoter Recognition, *IEEE Intelligent Systems Magazine*, July/August, 17 (4): 64-70.

[8] V. B. Bajic, S. H. Seah, A. Chong, S. P. T. Krishnan, J. L. Y. Koh, V. Brusic (2002c) Computer model for recognition of functional transcription start sites in polymerase II promoters of vertebrates, *Journal of Molecular Graphics & Modeling*, in print.

[9] D. Baxevanis & B. F. F. Ouellette (2001) Bioinformatics, *A Practical Guide to the Analysis of Genes and Proteins*, Wiley – Interscience, New York.

[10] D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, B. A. Rapp, D. L. Wheeler (2000) GenBank, Nucleic Acids Research 28: 15-18.

[11] S. Brunak, J. Engelbrecht, S. Knudsen (1991) Prediction of Human mRNA Donor and Acceptor Sites from the DNA Sequence, *Journal of Molecular Biology*, 220, 49-65.

[12] Burge, S. Karlin (1997) Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.* 268, 78-94.

[13] Y. Cai, P. Bork (1998) Homology-Based Gene Prediction Using Neural Nets, *Analytical Biochemistry* 265:269-274

[14] Derst, M. Reczko, A. Hatzigeorgiou (2000) Prediction of human translational initiation sites using a multiple neural network approach, *International Journal of Computers, Systems, and Signals*, 1, 169-179.

[15] T. A. Down, T. J. P. Hubbard (2002) Computational detection and location of transcription start sites in mammalian genomic DNA, *Genome Research*, 12:458-461.

[16] Dunham et al., Unpublished data, http://www.sanger.ac.uk/HGP/Chr22/

[17] J. W. Fickett, A. G. Hatzigeorgiou (1997) Eukaryotic promoter recognition, *Genome Research*, 7 861-878.

[18] L. M. Fu (1999) An Expert Network for DNA Sequence Analysis, *IEEE Intelligent Systems*, Jan./Feb. 65-71.

[19] Hannenhalli, S. Levy (2001) Promoter prediction in the human genome, *Bioinformatics*, 17 Suppl. 1, S90-S96.

[20] L. H. Hartwell, L. Hood, M. L. Goldberg, A. E. Raynolds, L. M. Silver, R. C. Veres (2000) *Genetics: From Genes to Genomes*, McGraw-Hill Higher Education, Boston.

[21] G. Hatzigeorgiou (2002) Translation initiation start prediction in human cDNAs with high accuracy, *Bioinformatics*, 18, 343-350.

[22] G. Hatzigeorgiou, P. Fiziev, M. Reczko (2001) DIANA-EST: a statistical analysis, *Bioinformatics*, 17, 913-919.

[23] G. Hatzigeorgiou, N. Mache, J. Wieland, M. Reczko, A. Zell (1994) Recognition of promoters and coding regions on eukaryotic sequences with neural networks, in *Bioinformatik 94*, pp.70-74, gopher://gopher.imb-jena.de/11/ftp/bioinf94.

[24] R. Hehl, E. Wingender (2001) Database-assisted promoter analysis. *Trends in Plant Science*, 6(6) 251-255.

[25] S. M. Hebsgaard, P. G. Korning, N. Tolstrup, J. Engelbrecht, P. Rouze, S. Brunak (1996) Splice site prediction in Arabidopsis thaliana pre-mRNA by combining local and global sequence information. *Nucleic Acids Res.* Sep 1;24(17):3439-52.

[26] P. Ioshikhes, M. Q. Zhang (2000) Large-scale human promoter mapping using CpG islands, *Nature Genetics*, 26, 61-63.

[27] E. Kel, O. V. Kel-Margoulis, P. J. Farnham, S. M. Bartley, E. Wingender, M. Q. Zhang (2001) Computer-assisted identification of cell cycle-related genes: new targets for E2F transcription factors. *J. Mol. Biol.* 309, 99-120.

[28] K. Klingenhoff, K. Frech, T. Quandt, T. Werner (1999) Functional promoter modules can be detected by formal models independent of overall nucleotide sequence similarity. *Bioinformatics*, 15(3) 180-186.

[29] S. Knudsen (1999) Promoter2.0: for the recognition of Pol II promoter sequences, *Bioinformatics*, 15, 356-361.

[30] M. Kozak (1999) Review: Initiation of translation in prokaryotes and eukaryotes, *Gene*, 234, 187-208

[31] N. Mache, M. Reczko, A. Hatzigeorgiou, Multistate time-delay neural networks for the recognition of POL II promoter sequences, ISMB96, St. Louis, http://www.informatik.uni-stuttgart.de/ipvr/bv/personen/mache 1996

[32] S. Matis, Y. Xu, M. Shah, X. Guan, J. R. Einstein, R. Mural, E. Uberbacher (1996) Detection of RNA Polymerase II Promoters and PolyAdenylation sites in human DNA sequence, *Computers & Chemistry*, 20, 135-140.

[33] L. Milanesi, M. Muselli, P. Arrigo (1996) Hamming-Clustering method for signals prediction in 5' and 3' regions of eukaryotic genes. *Comput Appl Biosci.* Oct; 12(5):399-404.

[34] W. Mount (2001) *Bioinformatics : Sequence and Genome Analysis.* Cold Spring Harbor Laboratory Pr., NY, USA

[35] T. Nishikawa, T. Ota, T. Isogai (2000) Prediction whether a human cDNA sequences contains initiation codon by combining statistical information and similarity with protein sequences, *Bioinformatics*, 16, 960-967.

[36] U. Ohler, H. Niemann (2001) Identification and analysis of eukaryotic promoters: recent computational approaches, *Trends Genet.* 17:56-60.

[37] U. Ohler, H. Niemman, G-c. Liao, G. M. Rubin (2001) Joint modeling of DNA sequence and physical properties to improve eukaryotic promoter recognition, *Bioinformatics*, 17 Suppl. 1, S199-S206.

[38] G. Pedersen, P. Baldi, Y. Chauvin, S. Brunak (1999) The biology of eukaryotic promoter prediction - a review, *Computers & Chemistry*, 23, 191-207.

[39] G. Pedersen, H. Nielsen (1997) Neural network prediction of translation initiation sites in eukaryotes: perspectives for EST and genome analysis, *Proceedings of the 5th International Conference on Intelligent System for Molecular Biology*, ISMB97, AAAI Press, pp.226-233.

[40] S. Rampone (1998) Recognition of Splice-Junctions on DNA Sequences by BRAIN learning algorithm. *Bioinformatics*, 14(8), 676-684.

[41] M. G. Reese (2001) Application of a time-delay neural network to promoter annotation in the Drosophila melanogaster genome. *Computers & Chemistry*, Dec; 26(1):51-56.

[42] M. G. Reese, N. L. Harris, F. H. Eeckman (1996) Large scale sequencing specific neural networks for promoter and splice site recognition, Biocomputing: *Proceedings of the 1996 Pacific Symposium* (L. Hunter and T. E. Klein, Eds.), January 2-7, World Scientific Publishing Co., Singapore, http://www.fruitfly.org/seq_tools/promoter.html

[43] M. G. Reese, G. Hartzell, N. L. Harris, U. Ohler, J. F. Abril, S. E. Lewis (2000) Genome

annotation assessment in Drosophila melanogaster, *Genome Research*, 10, 483-501.

[44] B. Rogozin, A. V. Kochetov, F. A. Kondrashov, E. V. Koonin, L. Milanesi (2001) Presence of ATG triplets in 5' untranslated regions of eukaryotic cDNAs correlates with a "weak" context of the start codon. *Bioinformatics*, 17(10):890-900.

[45] Salamov, T. Nishikawa, M. B. Swindells (1998) Assessing protein coding region integrity in cDNA sequencing projects, *Bioinformatics*, Jun; 14(5):384-390.

[46] S. L. Salzberg (1997) A method for identifying splice sites and translational start sites in eukaryotic mRNA, *Comput. Appl. Biosci*, 13, 365-376.

[47] M. Scherf, A. Klingenhoff, T. Werner (2000) Highly specific localisation of promoter regions in large genomic sequences by PromoterInspector: A novel context analysis approach, *J. Mol. Biol.*, 297, 599-606

[48] M. Scherf, A. Klingenhoff, K. Frech, K. Quandt, R. Schneider, K. Grote, M. Frisch, V. Gailus-Durner, A. Seidel, R. Brack-Werner, T. Werner (2001) First pass annotation of promoters on human chromosome 22, *Genome Research*, 11:333-340

[49] E. Snyder, G. D. Stormo (1995) Identification of coding regions in genomic DNA, *J. Mol. Biol.*, 248, 1-18

[50] E. Snyder, G. D. Stormo (1993) Identification of coding regions in genomic DNA sequences: an application of dynamic programming and neural networks, *Nucl. Acids Res.*, 21(3):607-613.

[51] D. Stormo, T. D. Schneider, L. Gold, A. Ehrenfeucht (1982) Use of the 'Perceptron' algorithm to distinguish translational initiation sites in *E.coli*, *Nucl. Acids Res.* 10:2997-3012

[52] T. A. Thanaraj (2000) Positional characterization of false positives from computational prediction of human splice sites, *Nucl. Acids. Res.*, 28(3):744-754.

[53] E. C. Uberbacher, R. J. Mural (1991) Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach, *Proc. Natl. Acad. Sci.* USA, 88:11261-11265, Dec.

[54] E. C. Uberbacher, Y. Xu, R. J. Mural (1996) Discovering and understanding genes in human DNA sequence using GRAIL, *Methods Enzymol.* 266:259-281.

[55] Wang (2001) Statistical pattern recognition based on LVQ artificial neural networks: Application to TATA-box motif, (M.Tech.degree), Technikon Natal, South Africa

[56] R. O. J. Weinzierl, *Mechanism of Gene Expression*, Imperial College Press, London, 1999.

[57] E. Wingender, X. Chen, E. Fricke, R. Geffers, R. Hehl, I. Liebich, M. Krull, V. Matys, H. Michael, R. Ohnhäuser, M. Prüß, F. Schacherer, S. Thiele, and S. Urbach (2001) The TRANSFAC system on gene expression regulation, *Nucl. Acids. Res.* 29: 281-283.

[58] Y. Xu, R. J. Mural, E. C. Uberbacher (1995) Correcting sequencing errors in DNA coding regions using a dynamic programming approach.*Comput Appl Biosci.* Apr;11(2):117-24.

[59] Y. Xu, R.J. Mural, J.R. Einstein, M.B. Shah, E.C. Uberbacher (1996) GRAIL: A Multi-Agent Neural Network System for Gene Identification, *Proceedings of IEEE*, 84(10) 1544-1552.

[60] Zhao, L. Hyman, C. Moore (1999) Formation of mRNA 3' ends in eukaryotes: mechanism, regulation, and interrelationships with other steps in mRNA synthesis. *Microbiol Mol Biol Rev*, Jun; 63(2):405-45

[61] Zien, G. Raetsch, S. Mika, B. Schoelkopf, C. Lemmen, A. Smola, T. Lengauer, K. –R. Mueller (2000) Engineering support vector machine kernels that recognize translation initiation sites, *Bioinformatics*, 16, 799-807.

[62] Zupicich, S. E. Brenner, W. C. Skarnes (2001) Computational prediction of membrane-tethered transcription factors. *Genome Biology* 2(12) 1-6.

Adapting Proactive Mobile Agents to Dynamically Reconfigurable Networks

Christian Erfurth and Wilhelm Rossak
Friedrich-Schiller-University Jena
{cen,rossak}@informatik.uni-jena.de, http://swt.informatik.uni-jena.de

Keywords: mobile agents, intelligent navigation, itinerary problem

Received: June 15, 2002

This paper describes our current results and research activities in the field of intelligent distributed computing in dynamic computer networks. In this area, we focus on mobile agents and mobile agent systems, especially on the itinerary problem of mobile agents. We describe a framework to improve an agent's migration process and to increase its autonomy and adaptivity in the planing and travel phase. As a basis we utilize the Tracy mobile agent system, developed at FSU during the last few years. For our purposes we expand this system by adding intelligent and adaptable system components, as well as a basic information infrastructure.

1 Introduction

Mobile agents are autonomous and proactive software entities which act on behalf of an owner, communicate and cooperate with each other, and have the ability to migrate through a heterogenous network of computers [21, 3]. Within the last years an increasing amount of research has been devoted to this area. Topics like communication [2], cooperation [15], migration [20], applications [16, 17], security issues [19], etc. are investigated at several universities and companies.

Mobile agents can be used in distributed real-time applications to avoid network latency by migrating to the critical domains in the network, help to reduce network traffic on the WWW by moving the code to the data sources, build and manage intelligent on-line communities (for humans and/or agents), etc. There are lots of other possible application areas that have been widely discussed. True is that mobile agents are not limited to a single killer application, or even need one. We at FSU look at (mobile) agents mostly from the viewpoint of distributed computing systems, i. e. as a new paradigm for open peer-to-peer computing.

In a typical scenario, a mobile agent visits more than one host in a network of so-called agencies to fulfill its task. Thus, it needs a "travel plan" when it starts its journey, a usually fixed plan that is provided by its programmer/owner. However, due to the size and possibly dynamical behavior of modern (ad-hoc) networks that change much faster than the agent-owner's perception of the net, mobile agents must by now develop the ability to construct and pro-actively adapt their own travel plans, which we call an *itinerary*.

We at FSU, in the TRACY project [1], focus as one hotspot on the itinerary problem of mobile agents by improving the basic technical efficiency of their migration and by increasing their autonomy and adaptivity in the planning and travel phase. This paper outlines our current research and results in the area of itinerary planning.

2 A Sample Scenario

Before we start to explain our framework we want to introduce a general scenario for the application. It should help to describe our idea and our goals. In the scenario, the mobile agents visits a set of agencies while migrating through the network to fulfill its task.

A user (the owner) hands over a task to an agent. Normally, such a task should not contain information on *HOW* to fulfill. Hence, the agent has to organize the journey through the network by itself. Therefore, the agent searches for suitable services at a map provided by the local agency. This map contains information on services within the net and some network characteristics. The search result is a set of agencies within the network that should be visited. Now the agent may trigger a route planer to use the available map's information on connection topology and qualities to identify a possible trip through the network. The result is a first travel plan – the itinerary. Before the agent begins the trip, it might use a migration planer to optimize the trip from an efficiency perspective.

Now the agent "executes the itinerary" and starts the migration. During the trip the agent visits service points and communicates and cooperates with other agents. At any point in time, but at least when migrating to further away agencies (map's information is more blurred for further away agencies), the agent may fine-tune and re-adapt its itinerary. This is achieved by taking advantage of the more detailed information now available locally. Finally, after its trip, the agent hands over the results to its principal. This might include a description of the visited agencies, a kind of travel report.

As indicated, we want to provide an infrastructure that enables agents to be more autonomous. A user should concentrate on *WHAT* the agent has to do and not on *HOW*.

3 The Framework

To improve the autonomy of mobile agents means mostly to improve the pro-activity of an agent. The agent has to initialize its itinerary on its own by locating services within the network which are suitable to fulfill the owner-given task. While visiting these service points (so-called agencies) within the network, the agent may in addition adapt to the changing environment (broken links, new services and platforms, etc.) and modify its itinerary dynamically.

We strive to provide an infrastructure that enables mobile agents to plan and modify their itinerary in an autonomous fashion. We utilize the existing systems *Tracy – The Mobile Agent System* and *The Tracy Domain Service*. Tracy is a general purpose mobile agent system implemented in Java 2 at FSU Jena [6, 4]. The Tracy Domain Service is a mechanism to network multiple sub-networks of agencies [5]. Thereby, the full set of available agencies is split into disjunct domains (local sets of agencies) to achieve better scalability and an improved information handling potential within a possibly dynamical network environment (see also sec. 4.1). To implement this idea, we expand the existing agent system middle-ware by adding intelligent and adaptable system components and a basic information infrastructure to each agency (see sec. 4.2 to sec. 4.4).

Our main idea is to build and provide *Dynamic Domain Maps* [8, 18]. Such a map contains information on available Domain Nodes (agencies within the local domain) and their services, as well as on the quality of the available connections in-between these nodes (e. g., bandwidth, reliability, etc.). The data on the map is kept up-to-date at regular intervals. The amount of data to be included can be handled quite easily due to the limited number of nodes within a domain. These domain maps form the basic information infrastructure that enables our mobile agents to act and plan in an intelligent fashion.

Between domains, compressed (summarized) domain maps can be exchanged. Thus, each domain has at least a limited view of (neighboring) remote domains and mobile agents are, therefore, able to target services also in other, non-local domains.

Fig. 1 shows the system components available on each agency the agent may visit – the *Map Module*, the *Route Planer*, and the *Migration Planer*. The Map Module is designed to support the agent in choosing and locating task-relevant services within the Dynamic Domain Maps. The Route Planer can be used by the agent to plan an efficient path through the (local) domain or to modify the itinerary during the journey. The Migration Planer is a Tracy-specific module to minimize network load and migration time. It links to the capability of Tracy to optimize the actual technical migration process once an itinerary has been developed.

These tasks can be triggered by any agent on demand while residing in an agency, without having to program and replicate the respective algorithms into each single agent instance (mobile agents want to be slim and fast). However, if and when an agent actually wants to use this type of infrastructure service remains within its own power of decision. Thus, it could also ignore the offered capabilities of this infrastructure completely and apply its private algorithms and rules.

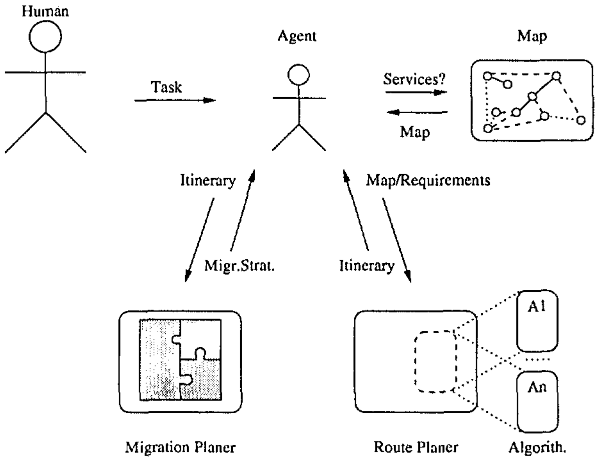


Figure 1: *Major Infrastructure Components*

4 Module Description

This section describes the major infrastructure components shown in fig. 1 in more detail, as well as the Tracy Domain Service which is an important basis for the framework.

4.1 Tracy Domain Service

An agent system network consists of an unstructured set of agencies. Generally such an agency is independent and not networked with other agencies. In Tracy the whole set of agencies is structured into domains, i. e. every agency belongs to exactly one domain and is no longer unknown to other local agencies. Within every domain there is a *Domain Manager*, which is a special agency responsible for administration tasks within the local domain. Every *Domain Node* (agency within the domain) needs to register with the Domain Manager which can, therefore, provide a list of known Domain Nodes and the services they offer.

On the left hand side of figure 2 you can see a local domain and its Domain Manager. Remote Domains are illustrated on the right hand side. Which of the many available domains is "local" depends for a specific agent simply on the fact on which agency it currently resides.

Multiple domains are networked via a *Domain Master* which acts as a Domain Manager for Domains (represented by the Domain Manager). Thus, the Tracy Domain Service is a two-stage concept: networked Domain Nodes within a local domain and networked domains within the web.

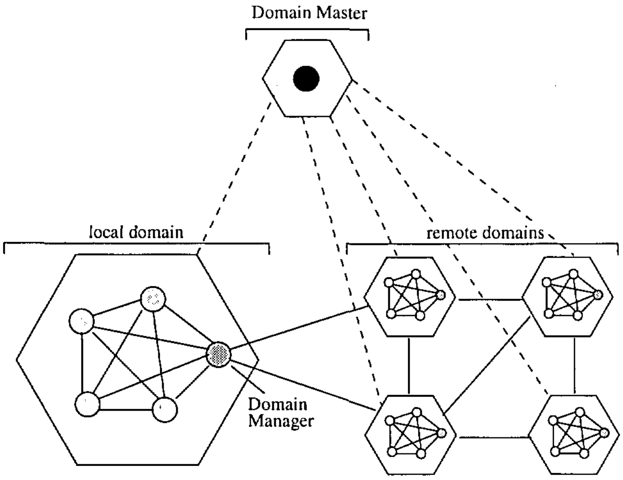


Figure 2: *Tracy Domain Concept*

4.2 Map Module

The starting point to build a local domain map is the list of nodes available within the domain, provided by the Domain Manager, and the relations between each one of them. This is the basic matrix structure we use for the map. An additional component is needed to collect detailed information on each node and on the network (line) characteristics, to fill the matrix with information. This component is the *Network Monitoring Module* [18].

The main problem within such a domain, and thus the complete network, is its possibly highly dynamical behavior and its heterogeneity. At anytime additional nodes may be added on while other nodes may drop out. The performance and line characteristics of nodes may be completely different, reaching from PDAs to high performance servers with good bandwidth. Actually measured values of line characteristics may oscillate quite intensively over time.

To get an up to date view of the current network situation, a lot of measurement experiments have to be done at short and regular time intervals. The more frequently, the better the data. However, network load is also increased and a good compromise has thus to be found.

The amount of data which has to be stored is increased by the number of nodes. If there should be information available on line characteristics for each node-to-node connection, the amount of required memory rises to the square with the number of nodes. This data can not be handled efficiently for a large set of unstructured nodes. The use of the Tracy Domain Service gives us thus the chance to limit the number of nodes to a fitting size (domains are limited in size, as they correspond usually to logical sub-networks).

As mentioned before, between domains there is a possibility to exchange data (domain maps) with compressed information. Of course, it is not so important to have an up-to-date map of remote domains as it is for the local domain. For this reason, and because of memory efficency, the map within the Map Module is split into a local domain map and a map of remote domains (compressed domain maps).

In fig. 3 the data structure of a domain map is shown in more detail. Local and remote maps are structured identically, the difference is in the granularity of the data. We distinguish between *Node Information* and *Line Information*. Node Information contains a technical description for each node (e. g. supported protocols) and information on provided services. If we talk about the local domain map, the term "node" means a single agency. Otherwise, if we talk about remote domains, a node represents a complete domain, represented by its Domain Manager. Line Information contains data on important line characteristics (e. g. bandwidth).

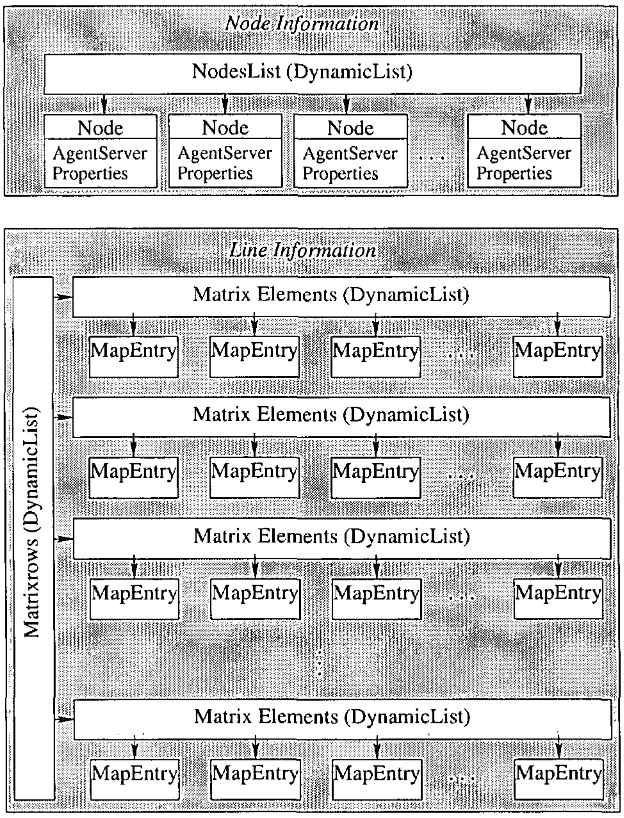


Figure 3: *Map Data Structure*

The Line Information part is mainly a connection matrix $\mathrm{M_c}$:

$$\mathrm{M_c} = \begin{pmatrix} c_{11} & c_{12} & \cdots \\ c_{21} & c_{22} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

with entries c_{ij} that are each a list of values $v_1, \ldots, v_k$ for the different line characteristics, representing the directed connection from node i to node j.

Map information is located on every agency. Within the matrix $\mathrm{M_c}$, there is one row i (Line Information part) which represents the own local agency i. This row contains the line information entries to all known hosts within the local domain. The list values v_i within the entries are extracted from measurement experiments.

The current values we provide for our agents planning purposes are always forecasts for the next expected result, based on a time-line of multiple existing measurements, and not simply the last available measurements. This helps to prevent an agent from extracting a possibly singular value that is wide off the usual scale and bases the agent's prognosis on a more solid and larger set of values.

To measure a certain line characteristic, we use software sensors. Such a sensor is located on every node and conducts experiments with the sensors located on other nodes within the same domain. If the local node is a Domain Manager, the sensor makes experiments also with remote Domain Managers. The precondition to do experiments is a list of known hosts, received from the Domain Manager or, on the domain level, from the Domain Master.

Within the matrix $\mathbf{M_c}$ there is only one row filled with entries by experiments of the local node. This line must be communicated to the other nodes within the local domain to fill their respective line structures, representing the remote matrices. We use the ongoing bandwidth experiments to transmit these data as well as Node Information. Once a local node has received data from all other nodes within the local domain, its local domain map is complete. The same procedure is done at the domain level to fill the map's structure for remote domains. As a result, every node (agency) has quite detailed information with regards to the local domain and a rough knowledge on known remote domains.

4.3 Route Planer Module

The Route Planer Module uses the Dynamic Domain Maps created by the Map Module. The task of the module is to generate a path through the network which can be used by the agent to visit all targets within nearly minimal migration time. Therefore, the agent has selected interesting nodes from the map or even has its own list of interesting agencies which will be a starting point to calculate the path through the net. For this calculation the module has to query the Map Module for line characteristics which can be used to weight connections in-between nodes.

The calculation for the whole trip can be done only, if the agent travels within a known area, like the local domain. If the agent visits remote domains, information is incomplete, i. e. the exact targets within the remote domains are not known yet and the line qualities within the remote domains are not available in detail. Hence, the agent should use the Route Planer again to re-calculate the path.

The route planing process itself is basically the Traveling Salesman Problem which is an NP complete type of problem. As a consequence, getting an optimal solution in practical application is ruled out. There are some heuristic algorithms (such as local search algorithm, genetic algorithm, simulated annealing, neural networks etc.) that have been applied extensively for solving such problems [12]. The comparative performance of the algorithms depends on the problem and the given detailed circumstances.

To be able to test different algorithms, the Route Planing Module has an interchangeable calculation component (see fig. 1). Our current results indicate that classical local optimization algorithms (2-Opt, 3-Opt, Lin-Kernighan) seem to be the best fit for our application ([13] compares various algorithms). Thereby, in order to improve a feasible tour it is modified by deleting some edges, thus breaking the tour into paths, and then reconnection those paths in another possible way [14]. Basically the algorithms in this family differ in the number of edges which will be exchanged: 2-Opt uses 2 edges, 3-Opt uses 3 edges, and Lin-Kernighan uses a variable number of edges.

These algorithms are designed to handle symmetrical distance/weight matrices. There are also algorithms for asymmetrical distance/weight matrices [11], which we will test in more detail at the moment.

4.4 Migration Planer Module

This module plans the agent's actual migration process from a technical perspective. In contrast to the other modules, it is a very Tracy specific enhancement and will not be covered in any real depth in this paper.

Tracy provides adaptable *Migration Strategies*, i. e. the way how migrations are processed. In general, an agent consists of different parts: the agent's state and its variable data parts (the serialized agent), and the agent's byte code (several class files). To execute an agent on a remote platform, at least the state and the data part have to be transmitted. The byte code files have to be transmitted only, if they are needed on the remote platform and are not available there.

Based on this basic observation, different types of migration strategies are possible: push strategies, pull strategies, and its variations. A *push* strategy means to transmit byte code along with the agent or even before the agent arrives at the remote platform. The opposite is a *pull* strategy where the agent loads code on demand.

The purpose of the use of different migration strategies is to transmit only code units which are needed (see fig. 4) and thus to cause minimal network load [7]. The granularity level in this question ranges from a set of class files to single classes, or even to single methods (planned) [9].

The decision which type of migration strategy should be used in a given situation is mainly influenced by three facts: network characteristics, node properties/contents, and the agent's own characteristics and behavior.

An example for a network characteristic based decision would be an agent starting from a node which is not connected to the web all the time, like a dial-up host or a mobile device. It should be transmitted as a whole because a dynamical download from code parts on request is only possible if the node is online. In general, this holds for all types of instable connections.

For a first realization of the Migration Planner Module, we currently limit our efforts to network characteristic based decisions and a class based granularity. Due to this

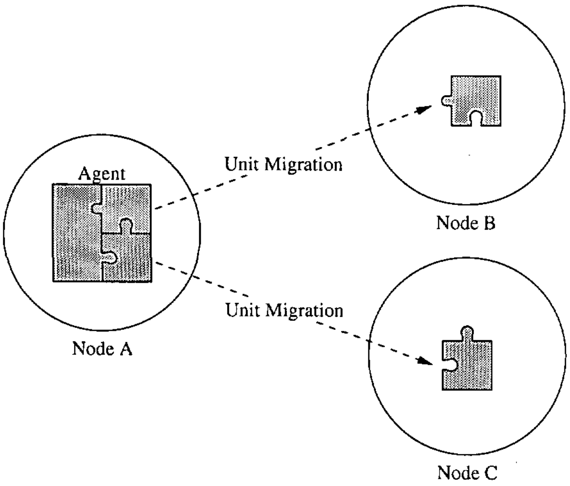


Figure 4: *Migration of selected agent parts*

limitations the module is also less Tracy specific and could be ported relatively easily to other available environments.

5 Conclusion

We have established an infrastructure of Dynamic Domain Maps as a basis to handle and structure a potentially slowly dynamical and very large network of agencies. This infrastructure enables mobile agents to navigate through a partially unknown network of nodes. With the help of additional modules at each agency, we enable mobile agents to plan an efficient itinerary and to adapt to changing environments, if they choose to do so. This solution respects each agent's autonomy, its independence and, at the same time, offers additional value and intelligence whenever necessary.

Tracy is by now a fully operational middle-ware for mobile agents that will be outsourced into a small company within the next year. The realization of the additional modules discussed in this paper (fig. 1) is ongoing research – the Map Module is already implemented and the Route Planer is well under way. We already started with the development of the Migration Planer based on the possibilities of Tracy's Migration Strategies. Furthermore, we plan to run more experiments to get an improved feeling for the potentials and possible drawbacks of our framework.

References

[1] Tracy – The Mobile Agent System. http://tracy.informatik.uni-jena.de, 2002.

[2] J. Baumann, F. Hohl, N. Radouniklis, K. Rothermel, and M. Straßer. Communication Concepts for Mobile Agent Systems. In Rothermel [17], pages 123–135.

[3] J. Bradshaw, editor. *Software Agents*. The MIT Press, Menlo Park, CA, 1996.

[4] P. Braun, J. Eismann, C. Erfurth, and W. Rossak. Tracy – A Prototype of an Architected Middleware to Support Mobile Agents. In *Proceedings of the 8th Annual IEEE Conference and Workshop on the Engineering of Computer Based Systems (ECBS), Washington D.C. (USA), April 2001*, pages 255–260, 2001.

[5] P. Braun, J. Eismann, and W. Rossak. A Multi-Agent Approach To Manage a Network of Mobile Agent Servers. Technical Report 12/01, Friedrich-Schiller-Universität Jena, Institut für Informatik, 2001.

[6] P. Braun, C. Erfurth, and W. Rossak. An Introduction to the Tracy Mobile Agent System. Technical Report Math/Inf/00/24, Friedrich-Schiller-Universität Jena, Institut für Informatik, Sept. 2000.

[7] P. Braun, C. Erfurth, and W. Rossak. Performance Evaluation of Various Migration Strategies for Mobile Agents. In *Fachtagung Kommunikation in verteilten Systemen (KiVS 2001), Hamburg (Germany), February 2001*, 2001.

[8] C. Erfurth and W. Rossak. Characterization and Management of Dynamical Behaviour in a System With Mobile Agents. In H. Unger, T. Böhme, and A. Mikler, editors, *Innovative Internet Computing System - Second International Workshop, IICS 2002, Kühlungsborn (Germany), June 2002*, volume 2346 of *Lecture Notes in Computer Science*, pages 109–119. Springer-Verlag, 2002.

[9] C. Fensch. Class Splitting as a Method to Reduce Network Traffic in a Mobile Agent System. Master's thesis, Friedrich-Schiller-Universität Jena, Institut für Informatik, 2001.

[10] G. Gutin and A. P. Punnen, editors. *The Traveling Salesman Problem and its Variations*. Kluwer Academic Publishers, May 2002.

[11] D. S. Johnson, G. Gutin, L. A. McGeoch, A. Yeo, W. Zhang, and A. Zverovitch. Experimental Analysis of Heuristics for the ATSP. In Gutin and Punnen [10], pages 445–488.

[12] D. S. Jonhson and L. A. McGeoch. The Traveling Salesman Problem: A Case Study in Local Optimization. In E.H.L.Aarts and J.K.Lenstra, editors, *Local Search in Combinatorial Optimization*, pages 215–310. John Wiley and Sons, Ltd., 1997.

[13] D. S. Jonhson and L. A. McGeoch. Experimental Analysis of Heuristics for the STSP. In Gutin and Punnen [10], pages 369–444.

[14] S. Lin. Computer Solutions of the Traveling Salesman Problem. *Bell System Technical Journal*, 44:2245–2269, 1965.

[15] A. Omicini, F. Zambonelli, M. Klusch, and R. Tolksdorf, editors. *Coordination of Internet Agents: Models, Technologies, and Applications.* Springer-Verlag, 2001.

[16] E. D. Pietro, O. Tomarchio, G. Iannizzotto, and M. Villari. Experiences in the use of Mobile Agents for developing distributed applications. In *Workshop su Sistemi Distribuiti: Algoritmi, Architetture e Linguaggi (WSDAAL'99), L'Aquila (Italy), September 1999*, 1999.

[17] K. Rothermel, editor. *Proceedings of the First International Workshop on Mobile Agents (MA'97), Berlin (Germany), April 1997*, volume 1219 of *Lecture Notes in Computer Science*, Berlin, 1997. Springer-Verlag.

[18] S. Schreiber. Beschreibung und Analyse von dynamischen Netzen für Agentensysteme. Master's thesis, Friedrich-Schiller-Universität Jena, Institut für Informatik, Apr. 2002.

[19] G. Vigna. *Mobile Agents and Security*, volume 1419 of *Lecture Notes in Computer Science.* Springer-Verlag, New York, 1998.

[20] D. E. White. A Comparison of Mobile Agent Migration Mechanisms. Senior Honors Thesis, Dartmouth College, June 1998.

[21] J. E. White. Mobile agents. In Bradshaw [3], pages 437–472.

User Profiling to Support Internet Customers: What Do You Want to Buy Today?

Giovanni Semeraro, Marco Degemmis and Pasquale Lops
Computer Science Department, Universita' di Bari, Via Orabona 4, Bari, Italy
semeraro@di.uniba.it, degemmis@di.uniba.it, lops@di.uniba.it

Keywords: User Profiling, Personalized Services, Intelligent Search

Received: July 27, 2002

In the recent years, the astonishing growth of the Internet and the considerable advances of Web technologies have promoted the development of electronic commerce. While e-commerce has not necessarily allowed businesses to produce more products, it has allowed them to provide consumers with more choices. Instead of tens of thousands of books in a superstore, consumers may choose among millions of books in an online store. Increasing choice has also increased the amount of information that scrupulous customers want process before they are able to select which items meet their needs. One way to address this information overload is the use of personalized systems able to support customers in retrieving information about products they are really interested in. Personalization has become an important strategy in Business-to-Consumer electronic commerce, where a user explicitly wants the e-commerce site to consider his or her own information, such as preferences, in order to improve access to relevant product information. In this paper, we propose a scheme to learn user profiles to support Internet customers. The proposed scheme is designed to handle different levels of users' interests simultaneously. Experimental evaluations show the promise of the approach.

1 Introduction

In the era of Internet, huge amount of data are available to everybody, in every place and at any moment. Even though this is extremely useful and exciting, the ever-growing amount of information at disposal generates cognitive overload and even anxiety, especially in novice or occasional users. Consumers have to spend more time to browse the net in order to find the information needed. Sometimes the contents of the Web pages are irrelevant to the consumers' expectation, but they have to read them in order to filter and get what they really want. The main challenge is to support web users in order to facilitate navigation through web sites and to improve searching among the extremely large Web repositories, such as Digital Libraries, online product catalogues or other generic information sources. A possible way to overcome this problem is the development of intelligent systems to provide personalized information services [Schafer et al., 2001]. A remarkable example arises in e-commerce Internet sites (marketplaces, electronic shops, and others), which provide information on thousands, even millions, of products and services and where it is well known that the process of buying products and services often implies a high degree of complexity and uncertainty. Effectively supporting user search and browsing over such large repositories entails the problem of properly understanding user needs, filtering out not relevant items, helping the user to formulate the most appropriate queries, and presenting results ranked according to their presumed relevance. The complexity of the problem could be lowered by the automatic construction of machine processable profiles that can be exploited to deliver personalized content to the user, fitting his or her personal choices. This is called *user modelling* process.

This paper discusses a method to model customers' preferences to offer personalized Internet services. Our approach relies on a two-step profiles generation process: in the first step, the system learns *coarse-grained profiles* in which the preferences are the product categories the user is interested into. In the second step, the profiles are refined by adding a probabilistic model of each preferred product category, induced from the descriptions of the products the user likes in these categories. The final outcome of the process is a more specific *fine-grained profile* able to discriminate between interesting and uninteresting products for the user.

The remainder of the paper is organized as follows. Section 2 discusses how the need to acquire new customers has made web personalization an indispensable part of e-commerce and how intelligent techniques are needed to achieve effective personalization. In Section 3, after introducing the use of learning techniques to construct models of users' preferences, a detailed description of the two-step profile generation process is provided. Section 4 presents some experimental results. Finally, conclusions and future works are drawn in Section 5.

2 Background

2.1 User profiling for web-content personalization

Internet services have evolved rapidly, leading to a constantly increasing number of modern Web sites. Enterprises are developing new business portals and providing enormous amounts of product information, which in many cases is heterogeneous, not structured and needs to be dealt with in a personalized way. It is particularly relevant to give the customers personal advice which reflects their *individual needs* and *interests*. Personalization has become an important strategy in Business-to-Consumer e-commerce, where a user explicitly wants the e-commerce site to consider his or her own information, such as preferences, in order to improve access to relevant products. A key issue in *web site personalization* is the automatic construction of accurate machine processable user profiles. This process is called user modelling and consists in ascertaining a few bits of information about each user, processing that information quickly and providing the results to applications, all without intruding upon the user's consciousness. User modelling is nothing more than a fancy term for automated personalization. Any application that behaves differently for different users employs a user model. The models themselves can be big or small, complex or simple, rich or sparse. They often have different names: personality profiles, psychographic profiles, or consumer databases. They are collections of information about an individual. Such collections of information are at best embryonic precursors of an ideal user model, which would possess an intimate and thorough knowledge of the user it refers to. In short, the user model should be able to recognize the user, know why the user did something, and guess what he or she wants to do next. Profiles could be used to deliver personalized content to the user, fitting his or her personal choices.

According to [Tasso and Omero, 2002], the main advantages of using the one-to-one personalization paradigm based on user profiling are:

- *Making the site more attractive for users* - A web site that takes into account user preferences is able to make recommendations reflecting user needs. Specifically in the e-commerce area, this will probably turn a significant part of browsers into buyers;

- *Obtaining trust and confidence* - Users will not be requested to explicitly insert information concerning their preferences, tastes, etc., but they will be able to participate in the management and updating of their personal profile. This will result in an increase of trust and confidence in a system able to automatically collect data about their preferences;

- *Improving loyalty* - The effectiveness of a personalization system improves in the long run. Every time a user interacts with the web site, the personalization mechanism collects new data about his or her preferences, so that a more and more satisfactory service can be offered. Even if a competitor uses a personalization system, it has to learn a lot of information about the new customer to be able to offer the same satisfactory service.

Among issues the personalization community is dealing with, the acquisition of a good user model is of special importance.

Machine learning techniques represent a very promising solution by which personalization in adaptive systems can be achieved. In fact, they have proved successful in cases where large data sets were available by providing tools for retrieval and filtering useful information and they applied to the definition of models of users interacting with an information system [Webb et al., 2001].

2.2 Related work

During recent years several systems have been designed to offer personalized services and to deliver user-tailored Web content. In this context, various learning approaches have been applied to discover user preferences and construct user profiles. A text categorization method is adopted by Mooney and Roy [Mooney and Roy, 2000] in their LIBRA system. It makes content-based book recommending by applying automated text categorization methods to product descriptions in Amazon.com, using a naïve Bayes text classifier.

A similar approach is adopted by Syskill & Webert [Pazzani and Billsus, 1997], which tracks the users browsing to formulate user profiles. The system identifies informative words from Web pages to be used as boolean features and learns a naïve Bayesian classifier to discriminate interesting Web pages on a particular topic from uninteresting ones.

Data mining methods are used by the 1:1Pro system [Adomavicius and Tuzhilin, 2001] in order to construct individual profiles made up of two sections. One part of the profile contains facts about a customer, and the other part is made up of rules describing the customer's behavior. The behavioral part of the profile is derived from transactional data, representing purchasing and browsing activities of each user.

A multistrategy machine learning approach is adopted in [Billsus and Pazzani, 1999] for the induction of accurate interest profiles that consist of separate models for long-term and short-term interests. This strategy is applied in designing an agent that learns about a user's interests in daily news stories. The use of this approach is due to the fact that the model must be capable of representing a user's multiple interests in different topics and must be flexible enough to adapt to a user's changing interests reasonably quickly, even after a long preceding training period.

3 Learning User Profiles for Intelligent Information Access

3.1 Exploiting Learning Techniques to construct models of users' preferences

By user profile we mean all the information collected about a user that logs to a Web site, in order to take into account his or her needs, wishes, and interests. Roughly, a user profile is a structured representation of the user's needs which a retrieval system could exploit in order to autonomously pursue the goals posed by the user. In a user profile modelling process, we have to decide *what* has to be represented and *how* this information is effectively represented. Generally the information stored in a user profile can be conceptually categorized in several classes, according to their source:

- *Registration Data,* such as name, address, email address, phone number, title, etc., useful to better serve the customer, for example to contact him or her either electronically or otherwise. Systems using personal data must abide to privacy laws [Kobsa, 2001];

- *Questions & Answers (Q&A),* that reveal a set of topics of possible interest for the customer;

- *Legacy Data,* gathered from external data sources such as CRM systems or ERP systems, etc.;

- *Past History,* that is data related to the browsing activity of users, or data stored in databases such as which of registered online customers responded to an e-mail campaign;

- *3rd Party,* gathered from marketing databases, demographic analysis, etc.;

- *Current Activity,* that contains the set of actions performed by the customer in the current session. This gives the flexibility and responsiveness to address rapidly changing customer needs.

A user profile is given by a list of attribute-value pairs, in which each attribute is given the proper value on the ground of the specific user it refers to. Each attribute-value pair represents a characteristic of that user. The list of attributes must be finite as well as the possible values related to each attribute. Examples of attributes in that list are: *last name*, *first name*, *age*, *address*, *job*, *annual income*, *preferences*, etc. The attribute list is the same for all the users. These attributes or features can be divided into three categories:

- *Explicit,* whose values are given by the user himself or herself (Registration Data or Q&A);

- *Existing,* that can be drawn by existing applications (e.g. job);

- *Implicit,* are elicited from the behavior of the user, through the history of his or her navigation or just from the current navigation.

A simple approach to acquire user preferences is the manual construction of a user profile: buyers have to fill an initial form that asks for personal data and some specific information (such as product categories of interest among the list of categories available in the store). In this way, only a limited amount of information can be acquired (customers might not be able or willing neither to fill large forms nor to provide personal details and preferences). The main problem of this process is its dependency on users willing to update their preferences. If users do not remember or do not want to spend their time in updating preferences, the personalization service will exploit unreliable or wrong data.

For these reasons, we adopt an approach that dynamically updates the user model by considering data recorded on past visits to the store (transactions). The approach uses rules - describing the behavior of a user - learned from customers' transactional data to construct individual profiles.

Profiles generated from a huge number of transactions tend to be statistically reliable.

In this paper we try to combine the rule-based approach with a text categorization method applied to semi-structured text - the product descriptions [Mooney and Roy, 2000] - in a two-step generation process in order to build more detailed user profiles. The two steps are described in the next subsections.

3.2 Discovering user preferences from transactional data

The first step of the profiling extraction process we propose is performed by the Profile Extractor (Figure 1), a module built upon an intelligent middleware component, called Learning Server, developed in the context of a digital library service [Semeraro et al., 2000, Semeraro et al., 2001]. It has in charge the induction of the rules to create the coarse-grained profiles containing the product categories the user is interested into.

The Profile Extractor employs supervised learning techniques to dynamically discover users' preferences from transactional data recorded during past visits to the e-commerce web site. Preferences are stored in a customer profile that could be useful to generate individual recommendations.

From our point of view, the problem of learning user's preferences can be cast to the problem of inducing general concepts from examples labelled as members (or non-members) of the concepts [Mitchell, 1997]. In this context, given a finite set of categories of interest $C = \{c_1, c_2, \ldots, c_n\}$, the task consists in learning the target concept T_i *"user interested in the category c_i"*.

In the training phase, each user represents a positive example of users interested in the categories he or she likes and a negative example of users interested in the categories he or she dislikes.

Moreover, we chose an operational description of the target concept T_i, using a collection of rules that match against the features describing a user in order to decide if he or she is a member of T_i.

In the COGITO project, the system was tested on the German virtual bookshop of the Bertelsmann Online company (www.bol.de). In this context, the set of categories of interest (translation in English of the German word is reported in parenthesis) is *C= {Belletristik (Fiction), Computer_und_Internet (Computer and Internet), Kinderbücher (Children's Books), Kultur_und_Geschichte (Culture and History), Nachschlagewerke (Reference Books), Reise (Travel), Sachbuch_und_Ratgeber (Monographs and Guidebooks), Schule_und_Bildung (School and Education), Wirtschaft_und_Soziales (Economics and Law), Wissenschaft_und_Technik (Science and Technique)}*. The members of *C* are the ten main book categories the BOL product database is subdivided into and represent the preferences of the users accessing the BOL Web site. Moreover, we needed to establish a formal description of the features (attributes) that represent each example. The complete set of attributes is listed in Table 1.

Transactional data about customers, stored in an XML file (Users' History), are used to set up the examples in order to train the system. A domain expert classifies each instance as member or nonmember of each book category, depending on the values of the attributes.

The architecture of the Profile Extractor (Figure 1) is made up of several sub-modules:

- *XML I/O Wrapper* - it extracts from the Users' History the data required to set up the instances used to train the learning component;

- *Profile Rules Extractor* - it processes the instances and induces a classification rule set for each book category. An example of a classification rule set inferred for a specific category of the BOL Web site is presented in Figure 2. The core of the module is WEKA [Witten and Frank, 1999], a tool that provides implementation of state-of-the-art learning algorithms, developed at the University of Waikato (New Zealand) and written in Java. The learning algorithm adopted in the rule induction process is PART [Frank and Witten, 1998], that produces rules from pruned partial decision trees, built using C4.5's heuristics [Quinlan, 1993].

- *Profile Manager* - it uses the rule sets inferred by the Profile Rules Extractor to predict whether a user is interested in each book category. All these classifications, together with the user's transactional data, are gathered to form the user profile. It is composed of two main frames: *factual*, containing personal and transactional data, and *behavioral*, containing the learned preferred book categories ranked according to the degree of interest computed by the learning system (Figure 3).

In the COGITO context, user profiles are used to personalize the search in the BOL product database according to the user's interests [Abbattista et al., 2002].

3.3 Exploiting the profiles for intelligent search

In the COGITO prototype, the coarse-grained profiles are exploited by a natural language assistant named Susanna. It offers a better support to customers using the BOL search engine to find interesting books. This improves the usability of the BOL web site, as demonstrated in the following scenarios.

Scenario 1: unknown user

A user is known by the COGITO system if he or she completes the BOL registration procedure. This step provides each customer with a personal identification number that is necessary both for recognizing a user and for collecting data on his or her preferences and for generating or updating his or her profile.

In the first scenario, an unknown user asks the chatterbot for a book by the author named King. Susanna finds several books through a remote call (deep linking) to the search engine available on the BOL Web site, and displays them as shown in Figure 4.

Notice that the books that rank at the top are authored by Stephen King. Books by other authors are found further down the list, which means that the user must scroll down a long list if he or she is not looking for a book by Stephen King. The customer not looking for a book by Stephen King can now choose to either refine the search by using an advanced search function or continue to chat with Susanna about different fields of interest.

Scenario 2: registered user

In the second scenario, the user has already been chatting to Susanna about some of his or her interests. Therefore, a profile of this user is available to the system, which can exploit it to accomplish a more precise search in the product database. Consider that the profile of such a user is the one presented in Figure 3 and the query submitted by the user is the same as in the previous scenario.

Now, the first book displayed is a book about Windows 2000 co-authored by Robert King (Figure 5). This result is due to the fact that the original query about King has been automatically expanded into "King AND Computer&Internet" (highlighted by the circle in Figure 5), since *Computer_und_Internet* is the category with the highest degree of interest in the profile of the user (Figure 3). This process is called query expansion. These scenarios highlight the dependence of the result set on the profile of the user that issued the query.

3.4 Refinement of profiles by learning from textual descriptions

The profiles inferred by the COGITO system are coarse-grained: they contain the book categories preferred by a user. Our intention was to enhance the profiles by taking into account the user's preferences in each category, in order to achieve more precise book recommendations. Thus, we adopted a probabilistic learning algorithm to classify the textual descriptions of the books, the naïve Bayes classifier [Mitchell, 1997, Sebastiani, 2002]. Naïve Bayes has been shown to perform competitively with more complex algorithms and has become an increasingly popular algorithm in text classification applications [Pazzani and Billsus, 1997, McCallum and Nigam, 1998].

Our prototype, called ITem Recommender (ITR), is able to classify books belonging to a specific category as interesting or uninteresting for a particular user: for example, the system could learn the target concept "*book descriptions the user finds interesting in the category Computer and Internet*".

Bayesian reasoning provides a probabilistic approach to inference. It is based on the assumption that the quantities of interest are governed by probabilistic distributions and that optimal decision can be made by reasoning about these probabilities together with observed data.

In our learning problem, each instance is represented by a set of *slots*. Each slot is a textual field corresponding to a specific feature of a book. The slots used by ITR are: *title*, *authors* and *textual annotation*. A simple pattern-matcher, the *Item Extractor* (Figure 6), analyzes the book descriptions and extracts the words, the *tokens* to fill each slot (it also eliminates stopwords and applies stemming). The text in each slot is a collection of words (a bag of word, *BOW*) processed taking into account their occurrences in the original text. Thus, each instance is represented as a vector of three BOWs, one for each slot. Moreover, each instance is labelled with a discrete rating (from 1 to 10) provided by a user, according to his or her degree of interest.

According to the Bayesian approach to classify natural language text documents, given a set of classes $C = \{c_1, c_2, \ldots, c_{|C|}\}$, the conditional probability of a class c_j given a document d is calculated as follows:

$$P(c_j|d) = \frac{P(c_j)}{P(d)} P(d|c_j)$$

In our problem, we have only 2 classes: c_+ represents the positive class (user-likes, corresponding to ratings from 6 to 10), and c_- the negative one (user-dislikes, ratings from 1 to 5). Since instances are represented as a vector of three documents, one for each BOW, the conditional probability of a category c_j given an instance d_i is computed using the formula:

$$P(c_j|d_i) = \frac{P(c_j)}{P(d_i)} \prod_{m=1}^{|S|} \prod_{k=1}^{|b_{im}|} P(t_k|c_j, s_m)^{n_{kim}} \qquad (1)$$

where $S = \{s_1, s_2, \ldots, s_{|S|}\}$ is the set of slots, b_{im} is the BOW in the slot s_m of the instance d_i, n_{kim} is the number of occurrences of the token t_k in b_{im}.

To calculate (1), we need to estimate the probability terms $P(c_j)$ and $P(t_k|c_j, s_m)$, from the training data, where each instance is weighted according to the user rating r:

$$w_+^i = \frac{r-1}{9}; \qquad w_-^i = 1 - w_+^i \qquad (2)$$

The weights in (2) are used for estimating the two probability terms according to the following equations:

$$\hat{P}(c_j) = \frac{\sum_{i=1}^{|TR|} w_j^i}{|TR|} \qquad (3)$$

$$\hat{P}(t_k|c_j, s_m) = \frac{\sum_{i=1}^{|TR|} w_j^i n_{kim}}{\sum_{i=1}^{|TR|} w_j^i |b_{im}|} \qquad (4)$$

In (4), n_{kim} is the number of occurrences of the term t_k in the slot s_m of the i^{th} instance, and the denominator denotes the total weighted length of the slot s_m in the class c_j.

This approach allows for the refinement of the profiles by including those words that turn out to be most indicative of user preferences for each preferred book category the system was trained on. An example of a fine-grained profile obtained by rating books about "Computer and Internet" is given in Figure 7.

The extraction phase from the BOL site is performed by the *Item Extractor*, and produces a local database of book descriptions. A profile-driven interface to the database has been developed in order to test the effectiveness of the fine-grained profiles in retrieving interesting items. When a user submits a query q to the system, the products in the result set R_q are ranked by the preferred category, as in the COGITO retrieval process, and by the classification value

$$P(c_+|d) \qquad d \in R_q$$

computed according to equation (1). This kind of profile gives a more precise products ranking with respect to the one returned by the coarse-grained profiles (Figure 8).

4 Experimental Results

Two different experiments were performed: the former consisted in observing the accuracy of ITR, the latter was conducted to evaluate the combination of the COGITO profiles with the ITR ones. For both experiments, 5 book categories at *uk.bol.com* were selected: for each category, the system has been trained by a specific COGITO user that rated approximately 90 books. In this way, a dataset of roughly 450 classified instances is obtained (Table 2).

The dataset was analyzed by means of a 10-fold cross-validation and several metrics were used in the testing phase. In addition to *Precision (Pr)*, *Recall (Re)* and *F-measure (F)*, we also adopted:

Normalized Distance-based Performance Measure (NDPM) - the distance between the ranking imposed by the user ratings and the ranking predicted by the system. Values range from 0 (agreement) to 1 (disagreement) [Yao, 1995];

Spearman's Rank Correlation (Rs) - a statistic measure used to establish whether there is any correlation between two sets of data. Its value falls between -1 and 1. A correlation coefficient of 0.3 to 0.6 is considered as moderate and above 0.6 is considered strong;

Error (E) - it is calculated as an average of the absolute difference between the user ratings and those predicted by the system.

Results of the experiment are reported in Table 3.

Values of *Pr*, *Re* and *F* provide evidence that ITR system produces accurate recommendations. NDPM is fairly consistent, while looking at *Rs* we observe that there is at least a moderate correlation for each category.

In the second experiment, each user was requested to submit 3 different queries to ITR. Then, a feedback is given to the system by rating the 20 top ranked books in each result set. The experiment has been modelled on the basis of two different scenarios. In the first scenario, books are ranked according to the COGITO profile (by category only), whereas in the second scenario the ranking is performed using the COGITO profile integrated with the ITR one. For both scenarios feedback evaluation results are given in Table 4.

For pairwise comparison of methods, the non-parametric Wilcoxon signed rank test is used [Orkin and Drogin, 1990], since the number of independent trials is relatively low and does not justify the application of a parametric test, such as the t-test. In this experiment, the test is adopted in order to evaluate the difference between the effectiveness of the different profiles by means of the metrics pointed out in Table 4, requiring a significance level $p < 0.05$.

On the basis of the values of the W statistic calculated above, we can deduce that there is a consistent statistically significant difference in performance among the two different profiles.

5 Conclusions and Future Work

In this paper we discussed the potential influence of using customers' profiles in retrieving relevant items in extremely large Web repositories, such as online catalogues. The success of the retrieval process can be measured in terms of the percentage of relevant and extraneous information retrieved. It is difficult to identify *qualitatively* the effectiveness of the retrieval process because only an individual user can determine what is truly relevant.

One way to address this issue is the use of techniques that automatically learn a personal profile containing a user's preferences by analyzing his or her browsing and purchasing history. In our approach, the added value of the profiles is to provide an intelligent search support in retrieving information customers are really interested in.

In this work, we propose a two-step profile generation process: in the first step, the system learns coarse-grained profiles in which the preferences are the product categories the user is interested into. In the second step, the profiles are refined by a probabilistic model of each preferred product category, induced from the descriptions of the products the user has labelled as interesting. The final outcome is a more specific fine-grained profile able to rank products according to the preferences it contains. Experimental results demonstrate the effectiveness of the strategy proposed.

In each classification problem, it is necessary to give many pre-classified examples, many of which positive, in order to learn a good classifier. This pre-classification process is often expensive and noise because it is done by a user who must classify each instance chosen to train the classifier, as in the second step of our profiling process. A possible solution for avoiding this hand-labelling job is using unlabelled examples in the learning phase maintaining the same classification process. The underlying idea is to use an algorithm for learning from few labelled examples and a large pool of unlabelled examples. In the future, we are planning to combine the *Expectation-Maximization (EM)* [Dempster et al., 1977] technique with the naïve Bayes classifier as in [Nigam et al., 2000]. EM is a class of iterative algorithms for maximum likelihood estimation in problems with incomplete data. The algorithm first trains a classifier using the available labelled documents, and probabilistically labels the unlabelled documents. It then trains a new classifier using the labels for all the documents, and iterates. In this way, a method for incorporating unlabelled data into supervised learning process is provided.

Acknowledgement

The COGITO project is funded by the European Commission under contract IST-1999-13347. The consortium is led by FhG-IPSI and comprises the following partners: Logica pdv Unternehmensberatung GmbH (Hamburg, Germany), BOL Medien GmbH (Rheda-Wiedenbrück, Germany), the System Analysis Department at Risø National Laboratory (Roskilde, Denmark), the LACAM Laboratory at the University of Bari (Bari, Italy), and Sword ICT S.r.l. (Bari, Italy).

References

[Abbattista et al., 2002] Abbattista, F., Degemmis, M., Licchelli, O., Lops, P., Semeraro, G., and Zambetta, F. (2002). Improving the usability of an e-commerce web site through personalization. In Ricci, F. and Smith, B., editors, *Recommendation and Personalization in Ecommerce, 2^{nd} International Conference on Adaptive Hy-*

permedia and Adaptive Web Based Systems, pages 20–29.

[Adomavicius and Tuzhilin, 2001] Adomavicius, G. and Tuzhilin, A. (2001). Using data mining methods to build customer profiles. *IEEE Computer*, 34(2):74–82.

[Billsus and Pazzani, 1999] Billsus, D. and Pazzani, M. J. (1999). A personal news agent that talks, learns and explains. In Etzioni, O., Müller, J. P., and Bradshaw, J. M., editors, *Proceedings of the 3rd International Conference on Autonomous Agents (Agents'99)*, pages 268–275, Seattle, WA, USA. ACM Press.

[Dempster et al., 1977] Dempster, M. M., Laird, N. M., and Jain, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society: Series B*, 39:1–38.

[Frank and Witten, 1998] Frank, E. and Witten, I. (1998). Generating accurate rule sets without global optimization. In *Proceedings of the 15th International Conference on Machine Learning*, pages 144–151. Morgan Kaufmann.

[Kobsa, 2001] Kobsa, A. (2001). Tailoring privacy to users' needs. In M.Bauer, Gmytrasiewicz, P., and Vassileva, J., editors, *User Modeling*, volume 2109 of *Lecture Notes in Artificial Intelligence*, pages 303–313. Springer, Berlin.

[McCallum and Nigam, 1998] McCallum, A. and Nigam, K. (1998). A comparison of event models for naive bayes text classification. In *Proceedings of the AAAI/ICML-98 Workshop on Learning for Text Categorization*, pages 41–48. AAAI Press.

[Mitchell, 1997] Mitchell, T. (1997). *Machine Learning*. McGraw-Hill, New York.

[Mooney and Roy, 2000] Mooney, R. J. and Roy, L. (2000). Content-based book recommending using learning for text categorization. In *Proceedings of the 5th ACM Conference on Digital Libraries*, pages 195–204, San Antonio, US. ACM Press, New York, US.

[Nigam et al., 2000] Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. M. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134.

[Orkin and Drogin, 1990] Orkin, M. and Drogin, R. (1990). *Vital Statistics*. McGraw-Hill, New York.

[Pazzani and Billsus, 1997] Pazzani, M. and Billsus, D. (1997). Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331.

[Quinlan, 1993] Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

[Schafer et al., 2001] Schafer, J., Konstan, J., and Riedl, J. (2001). E-commerce recommendation applications. *Data Mining and Knowledge Discovery*, 5(1/2):115–153.

[Sebastiani, 2002] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.

[Semeraro et al., 2000] Semeraro, G., Esposito, F., Fanizzi, N., and Ferilli, S. (2000). Interaction profiling in digital libraries through learning tools. In Borbinha, J. and Backer, T., editors, *Research and Advanced Technology for Digital Libraries*, volume 1923 of *Lecture Notes in Computer Science*, pages 229–238. Springer, Berlin.

[Semeraro et al., 2001] Semeraro, G., Ferilli, S., Fanizzi, N., and Abbattista, F. (2001). Learning interaction models in a digital library service. In M.Bauer, Gmytrasiewicz, P., and Vassileva, J., editors, *User Modeling*, volume 2109 of *Lecture Notes in Artificial Intelligence*, pages 44–53. Springer, Berlin.

[Tasso and Omero, 2002] Tasso, C. and Omero, P. (2002). *Personalization of web content: e-commerce, i-access, e-government (in Italian)*. Franco Angeli, Milano.

[Webb et al., 2001] Webb, G. I., Pazzani, M., and Billsus, D. (2001). Machine learning for user modeling. *User Modeling and User-Adapted Interaction*, 11:19–29.

[Witten and Frank, 1999] Witten, I. H. and Frank, E. (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers, San Francisco.

[Yao, 1995] Yao, Y. Y. (1995). Measuring retrieval effectiveness based on user preference of documents. *Journal of the American Society for Information Science*, 46(2):133–145.

Attribute	**Description**
User_id	Unique identifier of each user
Access_date	Identifies the date of the last access performed by the user
Connections_num	Total number of connections to the site performed by the user
Search_num<*CategoryName*>	Number of searches for a specific category
Search_freq<*CategoryName*>	Frequency of searches for a specific category
Purchase_num<*CategoryName*>	Number of purchases for a specific category
Purchase_freq<*CategoryName*>	Frequency of purchases for a specific category

Table 1: Description of the attributes used to represent examples. <CategoryName> denotes each one of the ten main book categories of the BOL Web site.

Category	Book descriptions	Books with annotation	Avg. annotation length	User Id
Computer & Internet	5414	4190 (77%)	42.39	User1
Fiction & literature	6099	3378 (55%)	35.54	User2
Travel	3179	1541 (48%)	28.29	User3
Business	5527	3668 (66%)	42.04	User4
SF, horror & fantasy	667	484 (72%)	22.33	User5
Total	20886	13261		

Table 2: Dataset information.

Category	Pr	Re	F	NDPM	Rs	E
Computer & internet	0.8500	0.5476	0.6660	0.3241	0.5499	0.3498
Fiction & literature	0.5971	0.7033	0.6459	0.4458	0.0676	0.3489
Travel	0.8100	0.8900	0.8481	0.3322	0.4683	0.2885
Business	0.7364	0.6800	0.7070	0.3741	0.3466	0.3576
SF, horror & fantasy	0.4695	0.7833	0.5871	0.3583	0.3970	0.4105
Avg.	0.6926	0.7209	0.6909	0.3670	0.3659	0.3611

Table 3: Results of the 10-fold cross validation.

User	Query	Pr		NDPM		Rs	
		COGITO	ITR	COGITO	ITR	COGITO	ITR
1	Java	0.50	0.90	0.594	0.423	-0.288	0.300
1	Graphics	0.30	0.70	0.465	0.328	0.156	0.490
1	Security	0.80	0.75	0.636	0.410	-0.412	0.278
2	Realism	0.35	0.50	0.421	0.400	0.258	0.329
2	romanticism	0.60	0.55	0.505	0.636	-0.053	-0.362
2	Science fiction	0.65	0.55	0.468	0.476	0.042	0.109
3	Islands	0.65	0.90	0.600	0.536	-0.288	-0.136
3	Guides	0.40	0.60	0.539	0.694	-0.130	-0.581
3	restaurants	0.30	0.35	0.505	0.415	0.037	0.338
4	Business manager	0.35	0.60	0.513	0.494	-0.074	0.018
4	enterprise solution	0.20	0.30	0.365	0.292	0.405	0.595
4	investment	0.50	0.70	0.547	0.605	-0.118	-0.312
5	s_king	0.30	0.60	0.589	0.197	-0.261	0.806
5	Space	0.10	0.40	0.447	0.184	0.178	0.839
5	King	0.70	1.00	0.550	0.326	-0.154	0.517
	Avg.	0.45	0.63	0.516	0.428	-0.047	0.215
	W=	130		-74		72	

Table 4: Results of the comparison between the COGITO and the ITR profiles.

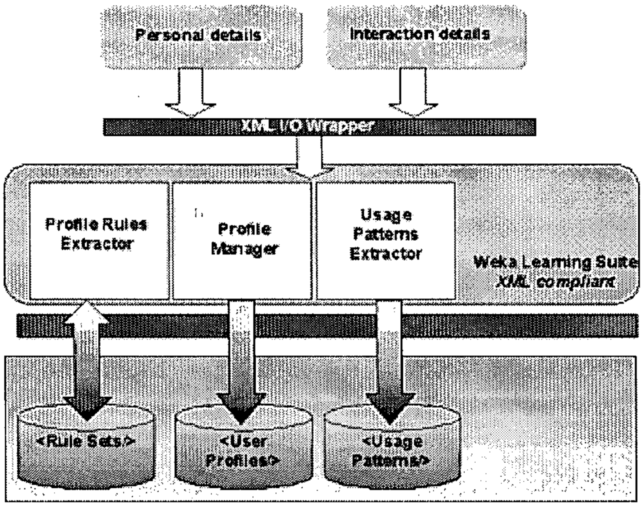
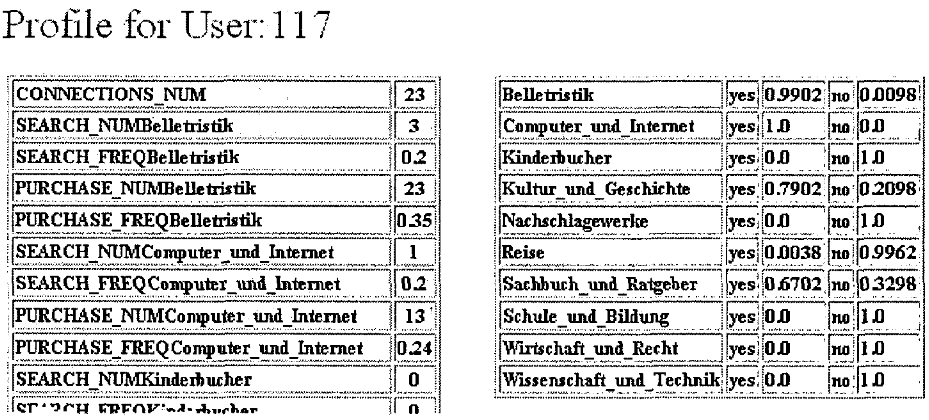


Figure 1: The Architecture of the Profile Extractor. In addition to the profiling sub-modules, the figure shows the *Usage Patterns Extractor*, that groups usage sessions in order to infer usage patterns exploited for understanding trends useful to further market studies.

There are 11 rules extracted for class **Kinderbücher**:

```
if search_number_Kinderbucher > 1.0
   and purchase_number_Schule_Und_Bildung <= 38.0
   and search_freq_Schule_Und_Bildung <= 0.26 then
   Class: yes
else if purchase_freq_Kinderbucher > 0.24
        and  purchase_number_Computer_Und_Internet <= 29.0
        and  purchase_number_Computer_Und_Internet > 5.0 then
   Class: yes
else if purchase_freq_Kinderbucher > 0.25
        and purchase_number_Schule_Und_Bildung <= 24.0
        and purchase_number_Kultur_Und_Geschichte <= 18.0 then
   Class: yes
else if search_num_Kinderbucher <= 1.0
        and search_number_Reise <= 3.0
        and search_freq_Nachschlagewerke <= 0.33
        and purchase_freq_Nachschlagewerke > 0.14
        and search_number_Reise <= 2.0
        and search_number_Sachbuch_Und_Ratgeber > 0.0 then
   Class: no
end if
...
Otherwise Class: no
```

Figure 2: An example of classification rules for the book category "Kinderbücher".

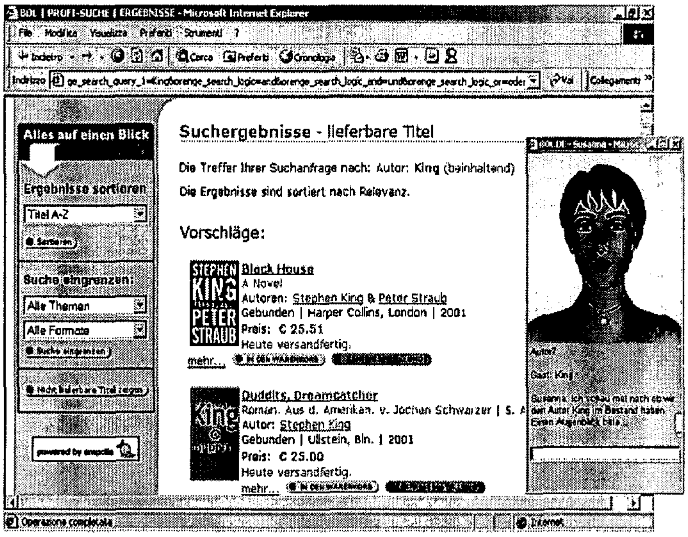


Profile for User: 117

CONNECTIONS_NUM	23
SEARCH_NUMBelletristik	3
SEARCH_FREQBelletristik	0.2
PURCHASE_NUMBelletristik	23
PURCHASE_FREQBelletristik	0.35
SEARCH_NUMComputer_und_Internet	1
SEARCH_FREQComputer_und_Internet	0.2
PURCHASE_NUMComputer_und_Internet	13
PURCHASE_FREQComputer_und_Internet	0.24
SEARCH_NUMKinderbucher	0
SEARCH_FREQKinderbucher	0

Belletristik	yes	0.9902	no	0.0098
Computer_und_Internet	yes	1.0	no	0.0
Kinderbucher	yes	0.0	no	1.0
Kultur_und_Geschichte	yes	0.7902	no	0.2098
Nachschlagewerke	yes	0.0	no	1.0
Reise	yes	0.0038	no	0.9962
Sachbuch_und_Ratgeber	yes	0.6702	no	0.3298
Schule_und_Bildung	yes	0.0	no	1.0
Wirtschaft_und_Recht	yes	0.0	no	1.0
Wissenschaft_und_Technik	yes	0.0	no	1.0

Figure 3: A COGITO coarse-grained profile.



Suchergebnisse - lieferbare Titel

Die Treffer Ihrer Suchanfrage nach: Autor: King (beinhaltend)

Die Ergebnisse sind sortiert nach Relevanz.

Vorschläge:

Black House
A Novel
Autoren: Stephen King & Peter Straub
Gebunden | Harper Collins, London | 2001
Preis: € 25.51
Heute versandfertig.
mehr...

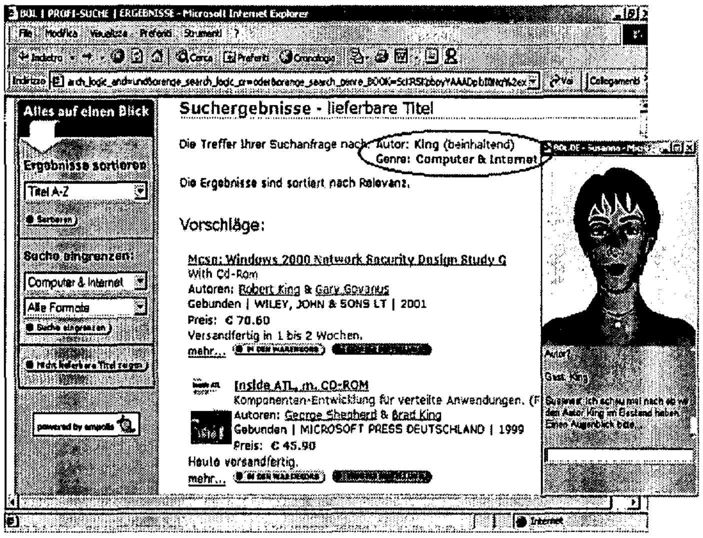
Duddits, Dreamcatcher
Roman. Aus d. Amerikan. v. Jochen Schwarzer | 5. A
Autor: Stephen King
Gebunden | Ullstein, Bln. | 2001
Preis: € 25.00
Heute versandfertig.
mehr...

Autor?

Gast: King

Susanna: Ich schau mal nach ob wir den Autor King im Bestand haben. Einen Augenblick bitte...

Figure 4: Susanna offers a long list of books belonging to several categories by authors whose last name is King.



Suchergebnisse - lieferbare Titel

Die Treffer Ihrer Suchanfrage nach: Autor: King (beinhaltend)
Genre: Computer & Internet

Die Ergebnisse sind sortiert nach Relevanz.

Vorschläge:

Mcsa: Windows 2000 Network Security Design Study G
With Cd-Rom
Autoren: Robert King & Gary Govanus
Gebunden | WILEY, JOHN & SONS LT | 2001
Preis: € 70.60
Versandfertig in 1 bis 2 Wochen.
mehr...

Inside ATL, m. CD-ROM
Komponenten-Entwicklung für verteilte Anwendungen. (F
Autoren: George Shepherd & Brad King
Gebunden | MICROSOFT PRESS DEUTSCHLAND | 1999
Preis: € 45.90
Heute versandfertig.
mehr...

Autor?

Gast: King

Susanna: Ich schau mal nach ob wir den Autor King im Bestand haben. Einen Augenblick bitte...

Figure 5: List of books by authors whose last name is King and that belong to the book category "Computer & Internet".

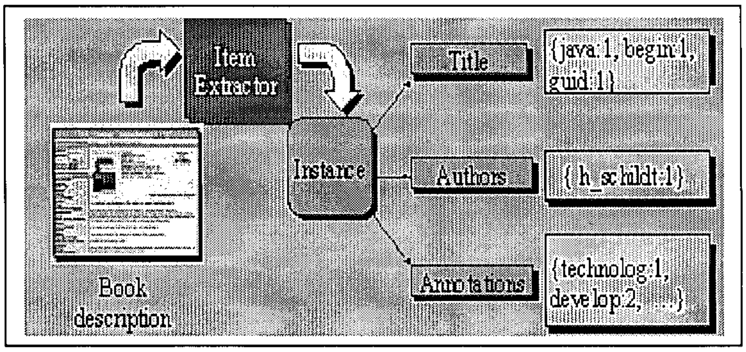
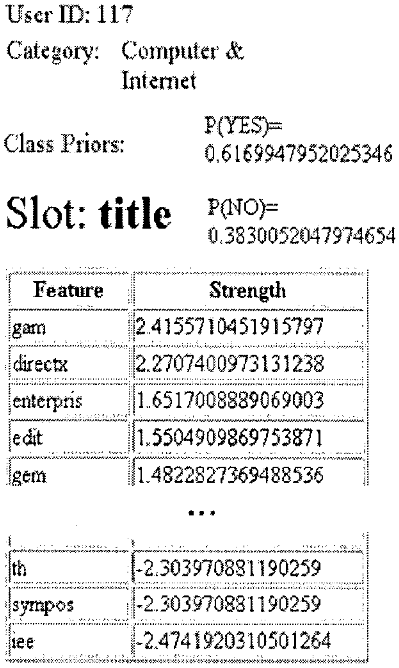


Figure 6: The BOW extraction process.



User ID: 117

Category: Computer & Internet

Class Priors: P(YES)= 0.6169947952025346

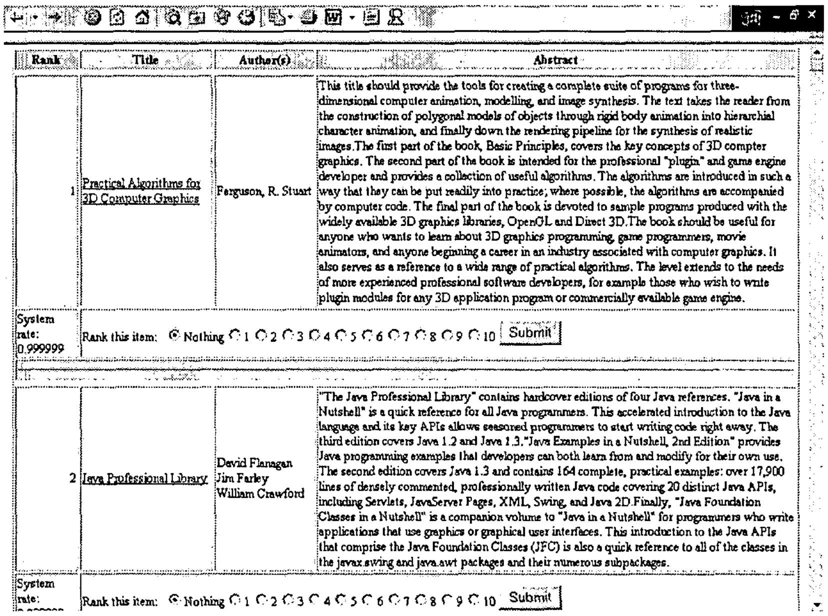
Slot: title P(NO)= 0.3830052047974654

Feature	Strength
gam	2.41557104519157797
directx	2.2707400973131238
enterpris	1.6517008889069003
edit	1.5504909869753871
gem	1.4822827369488536

...

th	-2.303970881190259
sympos	-2.303970881190259
iee	-2.4741920310501264

Figure 7: An example of ITR profile. The features are ranked according to a measure that indicates the discriminatory power of a word in classifying a book.



Rank	Title	Author(s)	Abstract
1	Practical Algorithms for 3D Computer Graphics	Ferguson, R. Stuart	This title should provide the tools for creating a complete suite of programs for three-dimensional computer animation, modelling, and image synthesis. The text takes the reader from the construction of polygonal models of objects through rigid body animation into hierarchial character animation, and finally down the rendering pipeline for the synthesis of realistic images.The first part of the book, Basic Principles, covers the key concepts of 3D compter graphics. The second part of the book is intended for the professional "plugin" and game engine developer and provides a collection of useful algorithms. The algorithms are introduced in such a way that they can be put readily into practice; where possible, the algorithms are accompanied by computer code. The final part of the book is devoted to sample programs produced with the widely available 3D graphics libraries, OpenGL and Direct 3D.The book should be useful for anyone who wants to learn about 3D graphics programming, game programmers, movie animators, and anyone beginning a career in an industry associated with computer graphics. It also serves as a reference to a wide range of practical algorithms. The level extends to the needs of more experienced professional software developers, for example those who wish to write plugin modules for any 3D application program or commercially available game engine.
System rate: 0.999999	Rank this item: ● Nothing ○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10 Submit		
2	Java Professional Library	David Flanagan Jim Farley William Crawford	"The Java Professional Library" contains hardcover editions of four Java references. "Java in a Nutshell" is a quick reference for all Java programmers. This accelerated introduction to the Java language and its key APIs allows seasoned programmers to start writing code right away. The third edition covers Java 1.2 and Java 1.3."Java Examples in a Nutshell, 2nd Edition" provides Java programming examples that developers can both learn from and modify for their own use. The second edition covers Java 1.3 and contains 164 complete, practical examples: over 17,900 lines of densely commented, professionally written Java code covering 20 distinct Java APIs, including Servlets, JavaServer Pages, XML, Swing, and Java 2D.Finally, "Java Foundation Classes in a Nutshell" is a companion volume to "Java in a Nutshell" for programmers who write applications that use graphics or graphical user interfaces. This introduction to the Java APIs that comprise the Java Foundation Classes (JFC) is also a quick reference to all of the classes in the javax.swing and java.awt packages and their numerous subpackages.
System rate:	Rank this item: ● Nothing ○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10 Submit		

Figure 8: An example of result set obtained by the ITR profile.

Informon—An Emergent Conscious Component

Anton P. Železnikar
Volaričeva ul. 8, Ljubljana, SI–1111
s51em@hamradio.si

Keywords: complexity, decomposition, emergentism, informational, informon, informonic consciousness, intention, learning curve, metatheory of informational, step curve of consciousness

Received: July 3, 2002

This article deals with a conscious entity being the building block of conscious systems. Such an informational entity is called informon, with its local and global function. The idea of informon as a conscious unit roots in the property of a sufficient complexity and learning capabilities, realized by different sorts of informational interpretation, formalism, general and metaphysicalistic decomposition concerning informon.

Emotional and cognitive informons show the complexity, intention, and capability of consciously, subconsciously, and self-consciously autonomous entities informing within a conscious system.

1 Understanding informon $\underline{\boldsymbol{\alpha}}$

Consciousness seems to be an informational phenomenon emerging within an individual (physical, biological, phenomenal) brain. This kind of conscious informing is grasped as an instantaneous informational process of temporary active attention concerning definite matters in informationally complex ways, coming into the foreground out of possible conscious background with conscious potentialities. The search for an realizable concept of consciousness concerns the possibility of design leading to something which could be called artificial, machine-like, or robotic consciousness.

To be conscious means to be involved informationally in a matter (entity, problem, process, event) of awareness, in an instantaneous way, in a moment of attention informationally concerning a distinct matter, that is, by an immediate concern that is spontaneous and transitory simultaneously, being a form of an informational event, happening as a matter of individual experience. This state of conscious possibility and possible experience concerning a determined or verbally named matter α is symbolized and informationally formalized by the notion of informon $\underline{\boldsymbol{\alpha}}$, published for the first time in German [9] and described before in [8].

To be more clear let's take the example of an emotion. In English, up to 2,500 names or name phrases for emotions can be distinguished. In Slovene, we can probably identify some hundreds names marking emotional states[1]. Taking a concrete emotion, for instance, named anger, informationally formalized by $\mathsf{a}_{\mathsf{anger}}$, we search for the meaning of this word in English. Using dictionaries, thesauri, encyclopedias, and our own imagination concerning anger, we can write a dissertation on anger. The meaning of anger is now determined by a collection of anger explaining sentences, paragraphs, psychological sketches, and the like. We understand that the collected meaning is in no way a final result and can be continued, refined, advanced, etc. The acquired meaning given by the linguistic collection concerning anger we call anger informon or informon, concerning anger, denoting it symbolically by $\underline{\mathsf{a}_{\mathsf{anger}}}$.

What do we have in mind by the obtained result of the concept? The name $\mathsf{a}_{\mathsf{anger}}$, a pure marker or basic informational operand concerning the English word anger, is now expanded or informationally propagated within the language to a meaning of anger, the still emerging informon denoted by $\underline{\mathsf{a}_{\mathsf{anger}}}$. In the complexity of language, this meaning is expressed by meanings of other emotional, cognitive, and other sorts of informons. The complexity becomes interweaved — informonically perplexed. In general, for such a situation, we can introduce the informon notations named α in a formula-dependent way,

$$\underline{\boldsymbol{\Phi}_{\underline{\boldsymbol{\alpha}}}} \rightleftharpoons \underline{\varphi_{\underline{\boldsymbol{\alpha}}}}\lfloor \alpha,\, \underline{\boldsymbol{\alpha}_1},\, \underline{\boldsymbol{\alpha}_2},\, \ldots,\, \underline{\boldsymbol{\alpha}_i},\, \ldots \rfloor$$

or, in a formula-system way,

$$\underline{\boldsymbol{\Phi}_{\underline{\boldsymbol{\alpha}}}} \rightleftharpoons (\alpha;\, \underline{\boldsymbol{\alpha}_1};\, \underline{\boldsymbol{\alpha}_2};\, \ldots;\, \underline{\boldsymbol{\alpha}_i};\, \ldots)$$

The third possibility would be to express the informational dependence of operands by the so-called informational concerning of the form

$$\underline{\boldsymbol{\Phi}_{\underline{\boldsymbol{\alpha}}}} \rightleftharpoons \underline{\boldsymbol{\alpha}}\lceil \alpha,\, \underline{\boldsymbol{\alpha}_1},\, \underline{\boldsymbol{\alpha}_2},\, \ldots,\, \underline{\boldsymbol{\alpha}_i},\, \ldots \rceil$$

The first expression is something we know as mathematical expression of a function or formula $\underline{\varphi_{\underline{\boldsymbol{\alpha}}}}$ depending on its operands (variables) α, $\underline{\boldsymbol{\alpha}_1}$, $\underline{\boldsymbol{\alpha}_2}$, $\ldots$, $\underline{\boldsymbol{\alpha}_i}$, $\ldots$. In case of an informational formula, parenthesis-like floor delimiters '$\lfloor$' and '$\rfloor$' are used instead of '(' and ')', respectively. The last parenthesis pair is used for delimiting subformulas occurring in informational formulas and, simultaneously, delimiting or enclosing the

[1] I believe that a study of this sort was not made until now. I tried something in this direction by translating known English terms for emotions, however, it happens that many distinguished English terms fell into the domain of one and the same Slovene term.

operands (elements) of a formula system $\underline{\Phi_{\underline{\alpha}}}$ as shown in the second expression. In the third expression, informon $\underline{\alpha}$ concerns informationally in a complex manner the name α and informons $\underline{\alpha_1}, \underline{\alpha_2}, \ldots, \underline{\alpha_i}, \ldots$ emerged during a system informing. It is essentially to stress that operands $\alpha, \underline{\alpha_1}, \underline{\alpha_2}, \ldots, \underline{\alpha_i}, \ldots$ certainly possess common operands and, in this way, become circularly structured, that is, inform circularly in one or another way. Detailed meanings of such expressions can be found in the study [8] and elsewhere in its references.

Let us show the examples of the discussed three informonic system notations $\underline{\Phi_{\underline{\alpha}}}$. Let the emotion of anger, $\mathfrak{a}_{\mathrm{anger}}$, inform dominantly in a situation within the conscious system, together with cognition concerning anger and some other involved emotions. All of these components are informonic (complex and consciously structured by themselves) and build up the informonic system of anger as the dominant intention. In the formula dependent way it means, according to the first kind of expression,

$$\begin{aligned}\underline{\Phi_{\underline{\mathfrak{a}_{\mathrm{anger}}}}} \rightleftharpoons \underline{\varphi_{\underline{\mathfrak{a}_{\mathrm{anger}}}}} \Big\lfloor \mathfrak{a}_{\mathrm{anger}},\ \underline{\mathfrak{c}_{\mathrm{cognition}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil}, \ldots, \\ \underline{\mathfrak{r}_{\mathrm{rage}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil},\ \underline{\mathfrak{s}_{\mathrm{sadness}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil},\ \underline{\mathfrak{d}_{\mathrm{depression}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil} \Big\rfloor\end{aligned}$$

In this expression components of the informonic formula $\underline{\varphi_{\underline{\mathfrak{a}_{\mathrm{anger}}}}}$ are linked by operators. In the second, formula system expression, components are linked through common operands, that is,

$$\begin{aligned}\underline{\Phi_{\underline{\mathfrak{a}_{\mathrm{anger}}}}} \rightleftharpoons \Big(\underline{\mathfrak{a}_{\mathrm{anger}}},\ \underline{\mathfrak{c}_{\mathrm{cognition}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil}, \ldots, \\ \underline{\mathfrak{r}_{\mathrm{rage}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil},\ \underline{\mathfrak{s}_{\mathrm{sadness}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil},\ \underline{\mathfrak{d}_{\mathrm{depression}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil} \Big)\end{aligned}$$

In the third, informational-concerning expression,

$$\begin{aligned}\underline{\Phi_{\underline{\mathfrak{a}_{\mathrm{anger}}}}} \rightleftharpoons \underline{\mathfrak{a}_{\mathrm{anger}}} \Big\lceil \underline{\mathfrak{c}_{\mathrm{cognition}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil}, \ldots, \\ \underline{\mathfrak{r}_{\mathrm{rage}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil},\ \underline{\mathfrak{s}_{\mathrm{sadness}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil},\ \underline{\mathfrak{d}_{\mathrm{depression}} \left\lceil \mathfrak{a}_{\mathrm{anger}} \right\rceil} \Big\rceil\end{aligned}$$

the system becomes circularly perplexed according to the dominating informon $\underline{\mathfrak{a}_{\mathrm{anger}}}$. It becomes evident that the three systems marked by $\underline{\Phi_{\underline{\mathfrak{a}_{\mathrm{anger}}}}}$ represent nothing other than the actual informon $\underline{\mathfrak{a}_{\mathrm{anger}}}$ within a complex conscious system.

2 Consciousness versus complexity, learning versus time, and decomposition versus emerging of informon

The main problem of informon concept is the implementation of its autonomous conscious function. In principle, each informational entity represented by an informational operand has the property to function consciously per se, to possess the conscious capability within its own informational organization. The hypothesis of the necessary complexity [2], exponential process of learning [3], and informational organization of emerging [8], offers a smart and believable approach to make informational entities intentionally conscious in an artificial way, that is, outside a natural biologically founded consciousness.

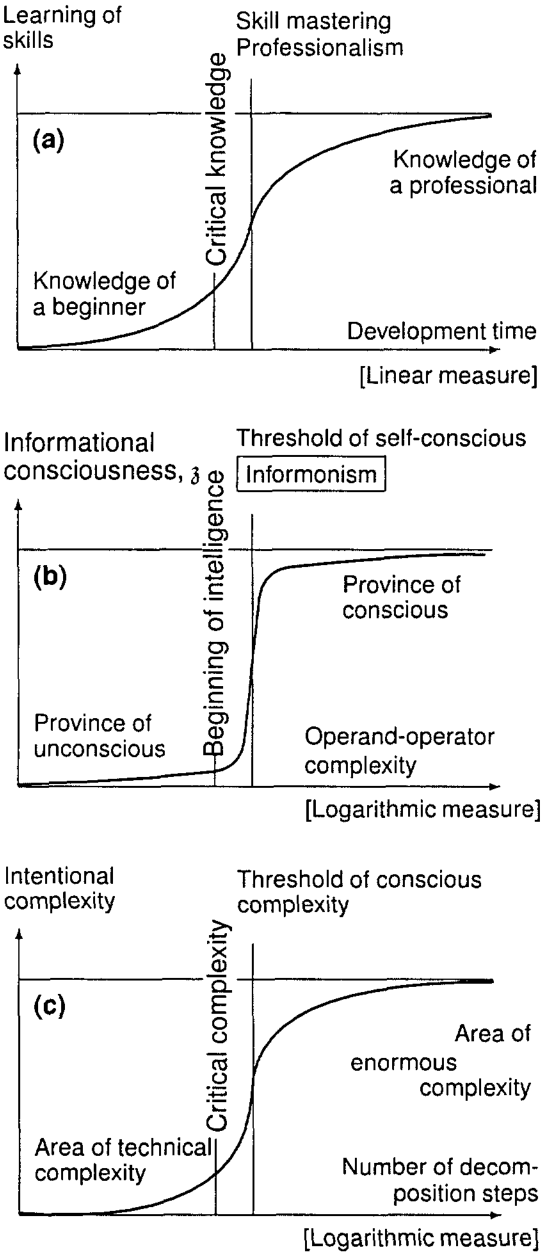


Figure 1: ***(a)*** *The exponential acceleration of an entity's knowledge through learning versus development time (Kurzweil [3] p. 34, the learning curve ∫.* ***(b)*** *The step curve ∫ of emerging of informational consciousness versus operand-operator complexity (Buttazzo [2]).* ***(c)*** *The complexity acceleration versus the number of informational decomposition steps (*Δ*,* $\mathfrak{M}, \mathfrak{I}$*, and other sorts of decomposition).*

Fig. 1 shows the learning versus time exponentialism (a) and the emerging of consciousness versus component complexity (b) being necessary for the occurrence of a conscious system. The learning curve ⁄in (a) shows how the state of a skill mastering advances through learning versus time from the knowledge of a beginner. In this part of the curve the acquiring of knowledge has an exponential nature when the beginner grasps a lot of new knowledge and begins to use it as his or her own experience. In a point of development, the critical knowledge is reached putting the beginner near the threshold of becoming the master of the learned skill. Finally, the skill mastering is reached, characterizing the skill of a professional, which then can be still improved in an advanced manner. The point of critical knowledge is a kind of the donkey bridge and, when crossed, the way to the professional side becomes open. One of the best examples of the learning curve is the exponential acceleration in computer technology, where complexity of components and their speed of operation advance exponentially through time. It is understood, that the critical knowledge of the complex technology is already reached, so the advancement to an artificial conscious system becomes possible through the next decades.

The step curve ⎽⎺in (b) is crucial for the occurrence of consciousness or conscious system. Human brain is an example of neuron and synapse complexity being sufficient for the real occurrence of consciousness in man. In primitive biological systems, the unconscious nervous activity takes care for the functioning of the necessary conditions of life. At some complexity of the neuronal system, intelligence can be observed being a manifest of the species successfully fitting to the environment. The state of the beginning of intelligence can be observed in machines using the technology of artificial intelligence (house equipment, today expert systems, and the like). As we see in Fig. 1 (b), this kind of intelligence is still far under the complexity needed for the most primitive form of consciousness. When complexity raises, a kind of jump to the conscious ability occurs. In man, some 10^{15} synapses and 10^{12} neurons constitute the conscious function being already substantially above the limit of consciousness occurrence.

Finally, the third curve (c) shows how a part of intentional complexity depends on the number of decomposition steps. The point of critical complexity, critical knowledge (a), and the beginning of intelligence (b) coincide. This coincidence is the condition for an informational system to become conscious, as the step curve (b) shows. The transition from unconscious to conscious happens along the "step" in curve (b). The other part of complexity might be conditioned by the number of physical components available for the machine in which a conscious system is embedded.

By the acquiring and emerging of knowledge the complexity of conscious system raises. Vice versa, the enlarged complexity accelerates the acquiring of knowledge. Under artificial or machine-like circumstances, artificial consciousness can widely surpass a biological consciousness in the brain. A silicon or quantum-technology brain will substantially surpass the biological brain in complexity, speed, interior and exterior communication (informational connectivity), and in (unforgettable) memory, being capable to accumulate everything happened to a conscious system in the past.

Functional and componetial complexity, building up the intentional informational complexity, remains one of the major requirements in the design and implementation of artificial consciousness. The informon as a component of conscious system needs the complexity in itself and in its environment.

3 To verbal and formal etymol- ogy of the word informon, informoron, and cogniton

For a physicist or an electronic engineer, the comparison between the electron concept and the informon concept may be instructive. For instance, electron is in no way a static, definite, or even transparent notion. On the contrary, electron is a dynamic phenomenon imagined in the brain of a physicist constructing the atom theory. It is not only a particle with mass and charge, but a moving particle within the atom organization and outside of the atom as an electrically charged particle. The invisible trajectories of electron within an atom concept exist just as a logical predisposition of the atomic model. Mainly, electron is characterized by its charge, mass, and the dynamics originating in the charge moving along an electro-magnetic field. The point of this story concerning electron is that its definition is in no way as simple as it would be seen at the first glance. The origin of the word electron is Greek and means amber.

Can we expect a simple story of an informon etymology after all? There will be several additions determining informon's nature in the world of informational. The word informon is a fusion of the Latin *informo* and the Greek *on* (being, entity). Translated into English, the meaning of *informon* is informational being or Being of the informational. Informational entity is the most general term to which the term informon belongs. It comes fore as the intentionally extremely complex entity connecting, interweaving, and in this manner including other informational entities and, in this way, informing and being informed consciously by itself. Informational entities, as they occur in conscious systems, are conscious, self-conscious, and subconscious as informational units. In this respect, informon brings a new meaning into the discourse of conscious structuring and organization, and understanding of its informational constitution. A conscious system—biological or artificial—can be imagined as an informonic organization, possessing initially a basic shell structure, complexly connected to various and numerous other entities in a circular way, ensuring the conscious maintaining and development of the involved informational entities.

Informing complexity	Name	Local informon	Global informon	Local informoron	Local cogniton
Operand's informing	α	$\underline{\alpha} \rightleftharpoons \left(\alpha; \underline{\mathcal{I}_{\underline{\alpha}}}; \underline{\mathcal{C}_{\underline{\alpha}}}; \underline{\mathcal{E}_{\underline{\alpha}}}\right)$ $\underline{\alpha} \rightleftharpoons (\alpha; \underline{\widehat{\alpha}}; \underline{\widetilde{\alpha}})$	$\underline{\widehat{\alpha}} \rightleftharpoons \left(\alpha; \underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}; \underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}\right)$	$\underline{\widetilde{\alpha}} \rightleftharpoons \left(\underline{\mathcal{I}_{\underline{\alpha}}}; \underline{\mathcal{C}_{\underline{\alpha}}}\right)$	$\underline{\widetilde{\widetilde{\alpha}}} \rightleftharpoons \underline{\mathcal{E}_{\underline{\alpha}}}$
Intentional informing	$\mathcal{I}_\alpha$	$\underline{\mathcal{I}_{\underline{\alpha}}} \rightleftharpoons$ $\left(\mathcal{I}_\alpha; \underline{\mathcal{I}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}; \underline{\mathcal{E}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}\right)$	$\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}} \rightleftharpoons$ $\left(\mathcal{I}_\alpha; \underline{\widehat{\mathcal{I}_{\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{E}_{\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}}}}\right)$	$\underline{\widetilde{\mathcal{I}_{\underline{\alpha}}}} \rightleftharpoons \left(\underline{\mathcal{I}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}\right)$	$\underline{\widetilde{\widetilde{\mathcal{I}_{\underline{\alpha}}}}} \rightleftharpoons \underline{\mathcal{E}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}$
Counter-informing	$\mathcal{C}_\alpha$	$\underline{\mathcal{C}_{\underline{\alpha}}} \rightleftharpoons$ $\left(\mathcal{C}_\alpha; \underline{\mathcal{I}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}; \underline{\mathcal{E}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}\right)$	$\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}} \rightleftharpoons$ $\left(\mathcal{C}_\alpha; \underline{\widehat{\mathcal{I}_{\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{E}_{\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}}}}\right)$	$\underline{\widetilde{\mathcal{C}_{\underline{\alpha}}}} \rightleftharpoons \left(\underline{\mathcal{I}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}\right)$	$\underline{\widetilde{\widetilde{\mathcal{C}_{\underline{\alpha}}}}} \rightleftharpoons \underline{\mathcal{E}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}$
Information-al embed-ding	$\mathcal{E}_\alpha$	$\underline{\mathcal{E}_{\underline{\alpha}}} \rightleftharpoons$ $\left(\mathcal{E}_\alpha; \underline{\mathcal{I}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}; \underline{\mathcal{E}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}\right)$	$\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}} \rightleftharpoons$ $\left(\mathcal{E}_\alpha; \underline{\widehat{\mathcal{I}_{\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{E}_{\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}}}}\right)$	$\underline{\widetilde{\mathcal{E}_{\underline{\alpha}}}} \rightleftharpoons \left(\underline{\mathcal{I}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}\right)$	$\underline{\widetilde{\widetilde{\mathcal{E}_{\underline{\alpha}}}}} \rightleftharpoons \underline{\mathcal{E}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}$

Table 1: *This table shows, how the initial (sublocal) components α, $\mathcal{I}_\alpha$, $\mathcal{C}_\alpha$, and $\mathcal{E}_\alpha$ of informon $\underline{\alpha}$ become local informons $\underline{\alpha}$, $\underline{\mathcal{I}_{\underline{\alpha}}}$, $\underline{\mathcal{C}_{\underline{\alpha}}}$, and $\underline{\mathcal{E}_{\underline{\alpha}}}$ and, in this sense, perform as individual informons by themselves. The similar is shown for the global informon.*

Nouns	Information, α	Informon, $\underline{\alpha}$	Informoron, $\underline{\widetilde{\alpha}}$	Cogniton, $\underline{\widetilde{\widetilde{\alpha}}}$
Adjectives	informational	informonic	informoronic	cognitonic
Adverbs	informationally	informonically	informoronically	cognitonically
Verbs	to inform	to informonize	to informoronize	to cognitonize
Participles	informing	informonizing	informoronizing	cognitonizing

Table 2: *An overview of meaning concerning the words information, informon, informoron, and cogniton and their English derivations.*

Etymologically, it can be useful to structure the informon into a more detail. For instance, it could be quite appropriate to distinguish the so-called informing-counterinforming subsystem and cognitive subsystem in the initial and further developed informon organization. For such a purpose, two new terms can be coined: *informoron* and *cogniton*, respectively. In this texture, informon $\underline{\alpha}$ is an informational fusion of the informon's name α, informon's informoron $\underline{\widetilde{\alpha}}$, and informoron's cogniton $\underline{\widetilde{\widetilde{\alpha}}}$, that is, a formula system $\underline{\alpha} \rightleftharpoons \left(\alpha; \underline{\widetilde{\alpha}}; \underline{\widetilde{\widetilde{\alpha}}}\right)$.

The informon comparison table Tab. 1 can be useful: it forces us to rethink the recursive definitions of informon and make the introduced symbolism transparent for the common use in the future. According to the table, in a concrete case, the informon $\underline{\alpha}$ components are informonically constituted as $\underline{\mathcal{I}_{\underline{\alpha}}} \rightleftharpoons \left(\mathcal{I}_\alpha; \underline{\mathcal{I}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}; \underline{\mathcal{I}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}; \underline{\mathcal{I}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}\right)$ (informon's informing), $\underline{\mathcal{C}_{\underline{\alpha}}} \rightleftharpoons \left(\mathcal{C}_\alpha; \underline{\mathcal{C}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}; \underline{\mathcal{C}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}\right)$ (informon's counterinforming), and $\underline{\mathcal{E}_{\underline{\alpha}}} \rightleftharpoons \left(\mathcal{E}_\alpha; \underline{\mathcal{E}_{\underline{\mathcal{I}_{\underline{\alpha}}}}}; \underline{\mathcal{E}_{\underline{\mathcal{C}_{\underline{\alpha}}}}}; \underline{\mathcal{E}_{\underline{\mathcal{E}_{\underline{\alpha}}}}}\right)$ (informon's informational embedding, called also cogniton). Evidently, the informon definition is recursive and can reach any reasonable informational depth of a situation decomposition.

In a similar way, this happens to the initially local components α, $\mathcal{I}_\alpha$, $\mathcal{C}_\alpha$, and $\mathcal{E}_\alpha$ (being named operands only), which in the framework of the global informon $\underline{\widehat{\alpha}}$ become global informons $\underline{\widehat{\alpha}}$, $\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}$, $\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}$, and $\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}$ as presented in Tab. 1.

Adequately to the local informorons and local cognitons in Tab. 1, global informorons can be defined in the form

$$\underline{\widehat{\widetilde{\alpha}}} \rightleftharpoons \left(\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}\right); \quad \underline{\widehat{\widetilde{\mathcal{I}_{\underline{\alpha}}}}} \rightleftharpoons \left(\underline{\widehat{\mathcal{I}_{\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}}}}\right);$$
$$\underline{\widehat{\widetilde{\mathcal{C}_{\underline{\alpha}}}}} \rightleftharpoons \left(\underline{\widehat{\mathcal{I}_{\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}}}}\right); \quad \underline{\widehat{\widetilde{\mathcal{E}_{\underline{\alpha}}}}} \rightleftharpoons \left(\underline{\widehat{\mathcal{I}_{\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}}}}; \underline{\widehat{\mathcal{C}_{\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}}}}\right)$$

and global cognitons in the fqrm

$$\underline{\widehat{\widetilde{\widetilde{\alpha}}}} \rightleftharpoons \underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}; \quad \underline{\widehat{\widetilde{\widetilde{\mathcal{I}_{\underline{\widehat{\alpha}}}}}}} \rightleftharpoons \underline{\widehat{\mathcal{E}_{\underline{\widehat{\mathcal{I}_{\underline{\widehat{\alpha}}}}}}}}; \quad \underline{\widehat{\widetilde{\widetilde{\mathcal{C}_{\underline{\widehat{\alpha}}}}}}} \rightleftharpoons \underline{\widehat{\mathcal{E}_{\underline{\widehat{\mathcal{C}_{\underline{\widehat{\alpha}}}}}}}}; \quad \underline{\widehat{\widetilde{\widetilde{\mathcal{E}_{\underline{\widehat{\alpha}}}}}}} \rightleftharpoons \underline{\widehat{\mathcal{E}_{\underline{\widehat{\mathcal{E}_{\underline{\widehat{\alpha}}}}}}}}$$

Global cognitons leave the question "What do they represent?" open. They certainly include a global understanding of the name (title, topic) α, its derivatives, and subnames, acquired through the informational decomposition propagation in global circumstances, for instance, including the meaning of α in different languages and image presentations and processing [5]. Such a sort of cogniton can emerge probably on a natual and an artificial level of consciousness, when the meaning material is assembled and systematically ordered and treated in a nervous system [6] and an informational machine.

To keep in mind the introduced symbols the following list with supplementary explanations could be helpful:

α	Name of entity, operand
$\underline{\alpha}$	Informon, as consciously organized complex entity in *local* or individual environment
$\underline{\tilde{\alpha}}$	Informoron, a complex component of informon, a subsystem of intentional and counterinformational part
$\underline{\tilde{\tilde{\alpha}}}$	Cogniton, a complex component of informon, a subsystem representing the cognitive part
$\underline{\hat{\alpha}}$	Global informon, as consciously organized complex entity in *global* environment, expanding over local or individual informational borders
$\underline{\hat{\tilde{\alpha}}}$	Global informoron, a complex component of global informon, a subsystem of intentional and counterinformational part
$\underline{\hat{\tilde{\tilde{\alpha}}}}$	Global cogniton, a complex component of global informon, a subsystem representing the cognitive part

Into the etymological view of informon additional words can enter, as shown in Tab. 2. The meaning of adjectives, adverbs, verbs, and participles must be understood in a pragmatic commonsense of the English language. For instance, informonic means to have an informational organization of informon. To informonize means to make an informational entity informon-like. Informoronic means to have an interior organization of informing and counterinforming simultaneously, etc.

4 Understanding consciousness in an innovative way

Usually, references concerning the topics of consciousness or mind root in some kind of traditional philosophy, for instance, such as Chalmers' conscious mind. In the last couple of years, questions touching problems of an artificial consciousness implementation come to the foreground (e.g., Buttazzo, 2001 [2]). The characteristic of references is a typical reductionism rooting on one side in an abstract philosophical discourse and, on the other side, in a too small number of properties pertaining to conscious systems. To such a conceptualism, usually a hierarchical organization of consciousness is studied and being disputed, rooting in the structure of brain and its functional locations.

For instance, the main question of the traditional consciousness studies remains if and when the design and implementation of an artificially conscious system would be possible. So, the experienced properties of conscious mind are rather very abstract concepts of unity, representation, supervenience, being in relation with, complexity, learning, and the like. A descriptive and exact formalism of conscious phenomena is practically not known, with few exceptions found in an exhaustive formalistic study of artificial consciousness in [8]. In this ongoing research some of the characteristic concepts or properties of conscious systems — biological or artificial — are systematically listed in Tab. 3.

The right column of the table is dedicated to the innovative philosophy, formalism, and methodology, all being relevant for the future artificial consciousness implementation. As the reader might observe, a substantial number of criteria for conscious system is identified — some of them by an entirely new approaches, for instance, how to generate, acquire, embed the meaning of solutions as a sort of results coming up in conscious systems. So, we can discuss some items of the table additionally and complemetarily.

The main point of this paper is to discuss the constitution of informon and show its functional (structural, organizational) sufficiency to operate or inform consciously, that is, as an independent conscious subsystem, coming into the conscious foreground upon the inner and environmental impulses, and putting it into the background, when other, for a particular informon relevant impulses, force other informons into the conscious foreground.

The concept of informonic consciousness does not reflect the physical and biological organization of the human brain with a specific and hierarchically structured architecture impacting and conditioning the information-functional specificity of human mind. It concentrates on informational possibility of conscious phenomena as they appear to an individual consciousness and can freely inform within a machine.

5 A concise definition of informon

5.1 The hypothetic background

Formal informational definition of informon needs a complex and environmentally perplexed definition of a formula system. Informon as an informational entity needs its verbal and formal definition. Verbally a lot concerning informon was said. What we need is a unique symbolism for informon $\underline{\alpha}$ rooting in its name α.

The name α, representing for instance a word, notion, concept, existent, imagined, etc., carries an intention of its meaning. The intentional means something which is already informonic in structure, organization, and propagation of α's meaning through and into informational environment (space):

- Structure of that which emerges out of the initial name α is informonic, e.g., basically metaphysicalistic in such or another way.

- Organization of the emerged is spontaneous and emergent in the propagation of the α's meaning. We usually say that the meaning of α arises in complexity, however in an intentionally consistent (predominant) way.

Item	Commonplace mind philosophy	Informational consciousness philosophy
(1)	Properties of conscious mind	Properties of informational consciousness
(2)	Unity of consciousness	I. systematism, informational formula systems
(3)	Consciousness representation: language, image, sound, etc.	I. operands (α), i. operators $(\models)$, i. formulas (φ), i.f. systems $\left(\varphi^{\circlearrowleft\|}, \Phi\right)$
(4)	Being in relation with other components	I. causality captured by i. formulas, common operands
(5)	Complexity of brain (neurons), mind	Complexity of i. formula systems: operands, operators
(6)	Phenomenal intentionality: in cognition, emotions, sensory domain	I. intentionality in meaning, goals, aims, i. stability, i. perseverance, i. orientation, ideology
(7)	Conscious spontaneity, occasioning, happening, being	I. unforeseeability, unpredictability, emergence of the informationally unexpected, possible
(8)	Direct and circular causality	I. causal relationship of i. entities
(9)	Conscious entity, event, process, experience	I. entity is: i. operand, i. operator, i. formula, i. formula system, emergence of meaning, understanding, expressed formally, generated by i. decomposition
(10)	Parallelism of conscious events, processes, experiences	I. parallelism of operands, operators, formulas, formula systems, represented formally
(11)	Conscious serialism, consequentialness, with simultaneous conscious parallelism of events	Non-circular and circular serial, reverse serial, biserial, split biserial i. formulas $\left(\varphi_{\triangleright}^{\triangledown}\right)$, and uniform and non-uniform i. formula systems $\left(\varphi_{\triangleright}^{\triangledown\|}, \Phi\right)$, where $\triangledown \in \{\lambda, \circlearrowleft\}$ and $\triangleright \in \{\rightarrow, \leftarrow, \rightleftarrows, (\rightarrow, \leftarrow)\}$
(12)	Conscious emergentism, experiential happening	I. arising of entities causally, unforeseeably, occasionally, happening accidentally, environmentally dependent
(13)	Conscious structure and organization, embodied in the brain architecture and its properties, depending on it	I. structure and organization, expressed by i. entities, informing continuously in an intentional, emotional, and cognitive way
(14)	Metaphysics of consciousness, the inner organization of conscious systems	I. metaphysicalism, with m. shell $\mathfrak{M}_{\triangleright}^{\circlearrowleft\|}\lceil\alpha\rceil$, where $\alpha \in \left\{\beta, \varphi_{\triangleright}^{\triangledown}, \varphi_{\triangleright}^{\triangledown\|}, \Phi\right\}$; β marks a primitive operand
(15)	Conscious ontogenesis	I. heaping, accumulating, enlarging of complexity, entity relatedness, causality, meaning
(16)	Conscious solving of problems	Producing of i. entity specific meaning, informationally fused in the solution f. systems
(17)	Events: conscious, sub-conscious, self-conscious, and not being conscious	Informing entities are operands, operators, formulas, and f. systems, representing conscious and other events, processes, and experiences
(18)	Conscious counterfactual relations	Counterinforming of i. entities, in decomposition $\mathfrak{M}_{\triangleright}^{\circlearrowleft\|}\lceil\alpha\rceil$, with counterinforming components $\mathfrak{C}_\alpha$ and $\mathfrak{c}_\alpha$
(19)	Conscious components, as subsystems of c.s., e.g., cognitive, emotional, attentional, behavioral, metastatic, etc.	Informons, informing in parallel: $\underline{\alpha_i} \rightleftharpoons \left(\alpha_i; \underline{\mathcal{I}_{\underline{\alpha_i}}}; \underline{\mathcal{C}_{\underline{\alpha_i}}}; \underline{\mathcal{E}_{\underline{\alpha_i}}}\right)$, where $i = 1, 2, \ldots$; α_i is the name, $\underline{\widetilde{\alpha}} \rightleftharpoons \left(\underline{\mathcal{I}_{\underline{\alpha_i}}}; \underline{\mathcal{C}_{\underline{\alpha_i}}}\right)$ is informoron, and $\underline{\widetilde{\widetilde{\alpha}}} \rightleftharpoons \underline{\mathcal{E}_{\underline{\alpha_i}}}$ is cogniton
(20)	Biological, individual consciousness	Artificial, computational, i. individual consciousness $\mathfrak{z}$
(21)	Physical embodiment of consciousness	I. embodiment in i. machines, robots, local and global webs, informationally accessible archives
(22)	Brains	I. machines, global web computing systems
(23)	Conscious system	Informonic systems, systems of informing informons
(24)	Formalism: exact, physicalistic, scientific, mathematical	I. formalism: embodied in i. axiomatism, informons, by operands, operators, formulas, and f. systems
(25)	Methodology: scientific and experimental	I. methodology: decomposition $(\Delta, \mathfrak{M})$, parenthesizing $(\mathfrak{P})$, schematizing $(\mathfrak{S})$, gestaltizing (Γ), rotation of operands $(\mathfrak{N})$, i. axiomatism, inferentialism
(26)	Authenticity of consciousness	Intentional i. emergentism, individualism, creativity, objectivism
(27)	Qualia	Individual, unique, sensory and metaphysical experience
(28)	Pure consciousness	Artificial consciousness as informon $\underline{\mathfrak{z}}$, concentrating on consciousness name $\mathfrak{z}$ as such

Agenda: c. — conscious, f. — formula, i. — informational, m. — metaphysicalistic, s. — system

Table 3: *A correspondence concerning the commonplace concepts and the informational concepts of consciousness. For details see [8].*

- The propagation of α's meaning comes into being by informational decomposition of that what emerged through previous processes of decomposition. Decomposition itself depends of the intentionality (meaning) of the name α, beginning from the initial shell of informon $\underline{\alpha}$, where the initial intention of the meaning is being captured.
- Decomposition of something is informational propagation of something's meaning through the informational space.

Hypothesis 1 (COMPLEXITY AND INITIAL SHELL) *An informational entity (operand, formula, or formula system) is said to inform consciously, if and only if*

1. *it possesses a sufficient amount of complexity, that is, sufficiently large or larger number of informational components (operands, formulas, formula systems), and*
2. *has an initial informonic structure of organization, that is, an informonic shell, from which the complexity can start according to the intention, given by the entity's name, e.g., α. An example of such a simple initial shell is informational metaphysicalism (decomposition) $\mathfrak{M}\lceil\alpha\rceil$.* □

Complexity is comprehended as substance (e.g., being material, brain-like) and spiritual (e.g., mental, phenomenal, mind-like, informational, informonic). Complexity emerges intentionally, as a consequence of intention in orientation, materialization, informational decomposition, meaning the propagation of informational intention in the complex informational space.

Hypothesis 2 (LEARNING AND DECOMPOSITION) *A conscious entity develops, that is, enlarges and advances its organization by*

- *learning, happening through the sensory system of the conscious system (e.g., nervous system including memory), and*
- *informational decomposition, meaning interpretation, deduced from intentional name and its semantic derivatives.* □

An informonic shell has, by definition, its intentionally informing part concerning the informon's name. Complexity, intentionality, learning, and decomposition are all emergent informational phenomena, being structured and organized spontaneously in a serial, biserial, parallel, and circular way of informing.

5.2 Definitions and a consequence concerning informon

We need a couple of definitions determining the concept of informon gradually, from the initial determination to its conscious informational function. The necessary conditions are, certainly: intentionality (informon's name α), complexity (in local and global informational circumstances), emerging of complexity by decomposition processes and, finally, the constitution of an informonic conscious system.

Definition 1 (LOCAL AND GLOBAL FORMULA SYSTEM) *Informon is a complex and perplexed local or global formula system, denoted by $\underline{\alpha}$ or $\widehat{\underline{\alpha}}$, respectively, with the operand name (argument) α, possessing a conscious structure of informational organization.* □

Definition 2 (LOCAL AND GLOBAL INFORMON OF INFORMONS) *A local informon $\underline{\alpha}$ informs in an informationally localized area, e.g., in an individual brain or machine and its sensory environment. A global informon $\widehat{\underline{\alpha}}$ is meant to inform in a global area up to the cosmological dimensions and, in this respect, represents an informonic system of other, informationally involved informonic systems. Both local and global informon are systems of informons.* □

Definition 3 (GENERAL INITIAL INFORMONIC DECOMPOSITION $\mathfrak{I}^{\circ\|}_{\triangleright}\lceil\alpha\rceil$) *Initial informonic decomposition of operand (name) α, called* informonic shell, *is, in its most general form, a circular biserial formula system, expressed schematically by*

$$\mathfrak{S}\left[\mathfrak{I}^{\circ\|}_{\triangleright}\lceil\alpha\rceil\right] \rightleftharpoons \begin{pmatrix} \alpha \models \underline{\mathcal{I}_{\underline{\alpha}}} \models \underline{\mathcal{C}_{\underline{\alpha}}} \models \underline{\mathcal{E}_{\underline{\alpha}}} \models \alpha; \\ \alpha \dashv \underline{\mathcal{I}_{\underline{\alpha}}} \dashv \underline{\mathcal{C}_{\underline{\alpha}}} \dashv \underline{\mathcal{E}_{\underline{\alpha}}} \dashv \alpha; \\ \alpha \models \underline{\mathcal{C}_{\underline{\alpha}}} \models \underline{\mathcal{I}_{\underline{\alpha}}} \models \underline{\mathcal{E}_{\underline{\alpha}}} \models \alpha; \\ \alpha \dashv \underline{\mathcal{C}_{\underline{\alpha}}} \dashv \underline{\mathcal{I}_{\underline{\alpha}}} \dashv \underline{\mathcal{E}_{\underline{\alpha}}} \dashv \alpha; \\ \alpha \models \underline{\mathcal{E}_{\underline{\alpha}}} \models \underline{\mathcal{C}_{\underline{\alpha}}} \models \underline{\mathcal{I}_{\underline{\alpha}}} \models \alpha; \\ \alpha \dashv \underline{\mathcal{E}_{\underline{\alpha}}} \dashv \underline{\mathcal{C}_{\underline{\alpha}}} \dashv \underline{\mathcal{I}_{\underline{\alpha}}} \dashv \alpha \end{pmatrix}$$

where $\triangleright \in \{\rightarrow, \leftarrow, \rightleftarrows, (\rightarrow, \leftarrow)\}$. The corresponding graph of this schema is presented in Fig. 2A. □

The scheme $\mathfrak{S}\left[\mathfrak{I}^{\circ\|}_{\triangleright}\lceil\alpha\rceil\right]$ covers the graph in Fig. 2A. We see how by considering this graph many different formula schemes can be formed. The graph corresponds exactly to the primitive formula system

$$\mathfrak{I}^{\circ\|'}_{\triangleright}\lceil\alpha\rceil \rightleftharpoons \begin{pmatrix} \alpha \models \underline{\mathcal{I}_{\underline{\alpha}}};\ \alpha \models \underline{\mathcal{C}_{\underline{\alpha}}};\ \alpha \models \underline{\mathcal{E}_{\underline{\alpha}}}; \\ \alpha \dashv \underline{\mathcal{I}_{\underline{\alpha}}};\ \alpha \dashv \underline{\mathcal{C}_{\underline{\alpha}}};\ \alpha \dashv \underline{\mathcal{E}_{\underline{\alpha}}}; \\ \underline{\mathcal{I}_{\underline{\alpha}}} \models \underline{\mathcal{C}_{\underline{\alpha}}};\ \underline{\mathcal{C}_{\underline{\alpha}}} \models \underline{\mathcal{E}_{\underline{\alpha}}};\ \underline{\mathcal{E}_{\underline{\alpha}}} \models \underline{\mathcal{I}_{\underline{\alpha}}}; \\ \underline{\mathcal{I}_{\underline{\alpha}}} \dashv \underline{\mathcal{C}_{\underline{\alpha}}};\ \underline{\mathcal{C}_{\underline{\alpha}}} \dashv \underline{\mathcal{E}_{\underline{\alpha}}};\ \underline{\mathcal{E}_{\underline{\alpha}}} \dashv \underline{\mathcal{I}_{\underline{\alpha}}} \end{pmatrix}$$

As expressed by the general form of informonic decomposition $\mathfrak{I}^{\circ\|'}_{\triangleright}\lceil\alpha\rceil$, particular cases are the following: serial informonic decomposition $\mathfrak{I}^{\circ\|'}_{\rightarrow}\lceil\alpha\rceil$, reverse serial informonic decomposition $\mathfrak{I}^{\circ\|'}_{\leftarrow}\lceil\alpha\rceil$, (proper) biserial informonic decomposition $\mathfrak{I}^{\circ\|'}_{\rightleftarrows}\lceil\alpha\rceil$, and split biserial informonic decomposition $\mathfrak{I}^{\circ\|'}_{\rightarrow,\leftarrow}\lceil\alpha\rceil$.

The informonic organization presented in Fig. 2A is understood as the maximal form of basic (initial) informonic

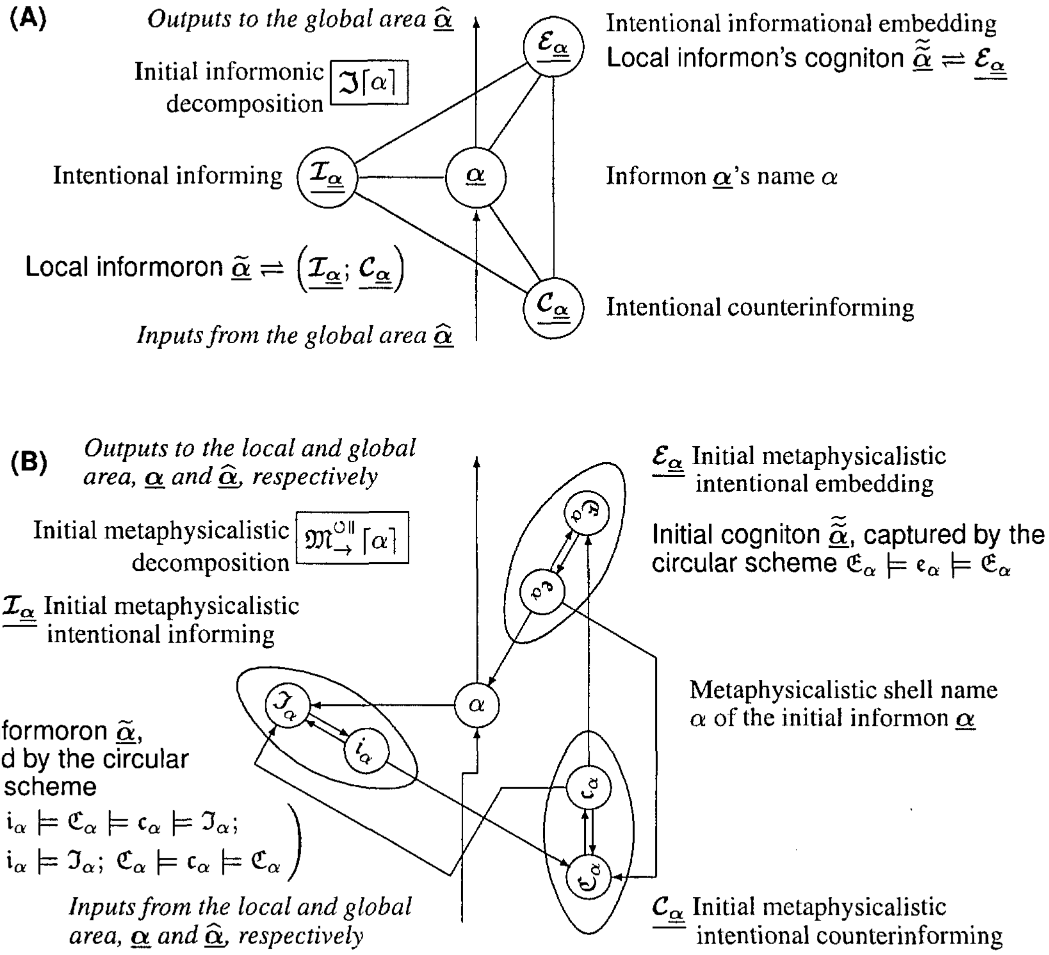
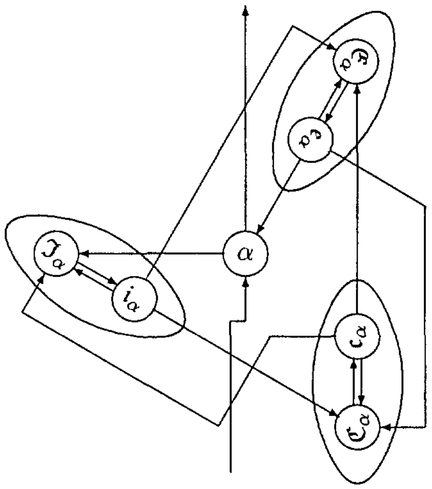

(A)
Outputs to the global area $\hat{\underline{\alpha}}$
Initial informonic decomposition $\boxed{\mathfrak{J}[\alpha]}$
Intentional informing
Local informoron $\tilde{\underline{\alpha}} \rightleftharpoons (\mathcal{I}_{\underline{\alpha}}; \mathcal{C}_{\underline{\alpha}})$
Inputs from the global area $\hat{\underline{\alpha}}$
$\mathcal{E}_{\underline{\alpha}}$
$\mathcal{I}_{\underline{\alpha}}$
$\underline{\alpha}$
$\mathcal{C}_{\underline{\alpha}}$
Intentional informational embedding
Local informon's cogniton $\tilde{\tilde{\underline{\alpha}}} \rightleftharpoons \mathcal{E}_{\underline{\alpha}}$
Informon $\underline{\alpha}$'s name α
Intentional counterinforming
(B)
Outputs to the local and global area, $\underline{\alpha}$ and $\hat{\underline{\alpha}}$, respectively
Initial metaphysicalistic decomposition $\boxed{\mathfrak{M}^{\circ\|}_{\rightarrow}[\alpha]}$
$\mathcal{I}_{\underline{\alpha}}$ Initial metaphysicalistic intentional informing
Initial informoron $\tilde{\underline{\alpha}}$, captured by the circular system scheme
$\begin{pmatrix} \mathfrak{I}_\alpha \models \mathfrak{i}_\alpha \models \mathfrak{C}_\alpha \models \mathfrak{c}_\alpha \models \mathfrak{I}_\alpha; \\ \mathfrak{I}_\alpha \models \mathfrak{i}_\alpha \models \mathfrak{I}_\alpha;\ \mathfrak{C}_\alpha \models \mathfrak{c}_\alpha \models \mathfrak{C}_\alpha \end{pmatrix}$
Inputs from the local and global area, $\underline{\alpha}$ and $\hat{\underline{\alpha}}$, respectively
$\mathcal{E}_{\underline{\alpha}}$ Initial metaphysicalistic intentional embedding
Initial cogniton $\tilde{\tilde{\underline{\alpha}}}$, captured by the circular scheme $\mathfrak{E}_\alpha \models \mathfrak{e}_\alpha \models \mathfrak{E}_\alpha$
Metaphysicalistic shell name α of the initial informon $\underline{\alpha}$
$\mathcal{C}_{\underline{\alpha}}$ Initial metaphysicalistic intentional counterinforming
$\mathfrak{E}_\alpha$
$\mathfrak{e}_\alpha$
$\mathfrak{I}_\alpha$
$\mathfrak{i}_\alpha$
α
$\mathfrak{C}_\alpha$
$\mathfrak{c}_\alpha$

Figure 2: ***(A)*** *The possible graph structure presents a maximal biserial form, called also the bicircular informational supervenience of informing, counterinforming, and informational embedding [8].* ***(B)*** *Serial metaphysicalistic decomposition of α is an example of the initial shell of informon, developing into the local and global informonic area.*

structure. Also, connections between the components of the graph are multitudinous and biserial. What is the minimum form of informon still guarantying the development from an initial informonic state to the conscious informing? The only condition of the informonic graph could be that all the components are circularly linked, thus, some of the connections sketched in Fig. 2A may not occur. Different forms of informational metaphysicalism satisfy this criterion [7, 8, 9]. The simplest form would be the initial serial loop with $\alpha, \mathcal{I}_\alpha, \mathcal{C}_\alpha$, and $\mathcal{E}_\alpha$, that is, a circular causality concerning all the informon's components.

The graph in Fig. 2B shows the initial metaphysicalistic variant of informonic structure marked by $\mathfrak{M}^{\circ\|}_{\rightarrow}$. One can see how informonic components α, $\mathcal{I}_{\underline{\alpha}}$, $\mathcal{C}_{\underline{\alpha}}$, and $\mathcal{E}_{\underline{\alpha}}$ are configured and connected within the graph. By a slightly modified graph to Fig. 2B, of the form



an intentionally stronger controlled initial cogniton is obtained. Intention $\mathfrak{i}_\alpha$ informs the cognitive informing $\mathfrak{E}_\alpha$ being directly involved into the emergence of the resulting cognition $\mathfrak{e}_\alpha$. The graph is structured in a stronger way than the graph $\mathfrak{G}\left[\mathfrak{M}_{\to}^{\circ\|}\lceil\alpha\rceil\right]$ in Fig. 2B. The primitive initial informon for the stronger graph, as presented above, can be expressed transparently by the system of subsystems of basic transitions corresponding to informonic componetns, that is,

$$\underline{\alpha'} \rightleftharpoons$$

$$\left(\alpha;\begin{pmatrix}\alpha \models \mathfrak{I}_\alpha;\\ \mathfrak{I}_\alpha \models \mathfrak{i}_\alpha;\\ \mathfrak{I}_\alpha \dashv \mathfrak{i}_\alpha;\\ \mathfrak{i}_\alpha \models \mathfrak{C}_\alpha;\\ \mathfrak{i}_\alpha \models \mathfrak{E}_\alpha;\\ \mathfrak{c}_\alpha \models \mathfrak{I}_\alpha\end{pmatrix};\begin{pmatrix}\mathfrak{i}_\alpha \models \mathfrak{C}_\alpha;\\ \mathfrak{C}_\alpha \models \mathfrak{c}_\alpha;\\ \mathfrak{C}_\alpha \dashv \mathfrak{c}_\alpha;\\ \mathfrak{c}_\alpha \models \mathfrak{E}_\alpha;\\ \mathfrak{e}_\alpha \models \mathfrak{C}_\alpha\end{pmatrix};\begin{pmatrix}\mathfrak{c}_\alpha \models \mathfrak{E}_\alpha;\\ \mathfrak{E}_\alpha \models \mathfrak{e}_\alpha;\\ \mathfrak{E}_\alpha \dashv \mathfrak{e}_\alpha;\\ \mathfrak{e}_\alpha \models \mathfrak{C}_\alpha;\\ \mathfrak{i}_\alpha \models \mathfrak{E}_\alpha;\\ \mathfrak{e}_\alpha \models \alpha\end{pmatrix}\right)$$

representing, transparently,

$$\underline{\alpha'} \rightleftharpoons \left(\alpha; \underline{\mathcal{I}'_{\underline{\alpha'}}}; \underline{\mathcal{C}'_{\underline{\alpha'}}}; \underline{\mathcal{E}'_{\underline{\alpha'}}}\right)$$

Definition 4 (CONSCIOUS SYSTEM) *A conscious system $\underline{\mathfrak{z}}$ is a system of informonic components $\underline{\mathfrak{z}_i}$, $i = 1, 2, \ldots$, where some of the components, corresponding to a situation and time, represent the contents being currently in the conscious foreground (attention).* □

Consequence 1 (CONSCIOUSNESS OF INFORMONS) *An informational entity (formula system) named β informs consciously if and only if it is initially structured in an intentional way according to Defs. 1—4, and if it, through sufficient number of decomposition steps, became structured and informationally organized to a sufficient degree of complexity. In this case, the named entity β has reached the informonic, that is, conscious organization $\underline{\beta}$.* □

Proof 1 Although the proof of Cons. 1 is evident, its proving interpretation might be truly useful. How did the initial name β expand or arrive to informon $\underline{\beta}$? At the beginning, the first step toward the informonizing of β is the initial, the so-called informonic shell decomposition of β, marked by $\mathfrak{I}\lceil\beta\rceil$ (see, for instance, Fig. 2A or Fig. 2B). By the initial decomposition $\mathfrak{I}\lceil\beta\rceil$, the initial informon $\underline{\beta} \rightleftharpoons \mathfrak{I}\lceil\beta\rceil$ is coming into existence. This decomposition gives to $\underline{\beta}$ the necessary intentionally informing, intentionally counterinforming (emotional), and intentionally embedding (cognitive) organization of the shell. In the next procedures of decomposition, the inner components of the initial informon will propagate and expand into the environmental, intentionally related tissue of the informational space. Thus, the complexity of $\underline{\beta}$ will rise and, gradually, will reach the complexity, being necessary for $\underline{\beta}$'s conscious behavior. Defs. 1—4 concerning $\underline{\beta}$ will gradually, by decompositions considering the $\underline{\beta}$'s intention, structure and organize in a complex way the informon $\underline{\beta}$ for its conscious function.

To stress, decomposition considers learning and, with $\underline{\beta}$ as the meaning structure and organization of intentional meaning concerning the name β, the acquired knowledge is accumulated as the $\underline{\beta}$'s experience up to the moment, when in a situation, $\underline{\beta}$ enters into the occurring conscious domain. Virtually, this proves the value of the consequence. □

5.3 A new philosophy with informonic consciousness

Informon is an informational unit being in the foreground of the instantaneous consciousness. Consciousness is nothing else than a momentary happening among informons, some of them coming to the conscious surface (attention). In a time interval, consciousness is understood to be a system of actively informing informons, that come into conscious existence, that emerge from the unconscious background of consciousness into the conscious existence; they emerge as momentary conscious entities, out of the subconscious domain, where they inform, and depend emergently and crucially on the temporary sensory situation. Consciousness is an informonic theater in which informons play their informational roles for the observing and informationally acting conscious system. In this view, conscious system happens as a momentary sequence of groups of informons marching through the time of conscious experience. Consciousness is experienced as a column of informon groups in the understanding of past, present, and future situations. In this sort of grasping, consciousness is just an informon by itself and to itself. The consciousness informon $\underline{\mathfrak{z}}$ understands currently active informons $\underline{\mathfrak{z}_1}, \underline{\mathfrak{z}_2}, \ldots$ and specific informons $\underline{\mathfrak{z}_1}, \underline{\mathfrak{z}_2}, \ldots$ understand (are aware of) consciousness $\underline{\mathfrak{z}}$.

Each informon is a conscious entity per se. As such, an informon consciously grasps other informons and itself. For instance, $\underline{\mathfrak{a}_{\text{anger}}}$ grasps consciously itself and other, the informon informationally accompanying informons as, for instance, $\underline{\mathfrak{s}_{\text{sadness}}}$, $\underline{\mathfrak{e}_{\text{embarrassment}}}$, $\underline{\mathfrak{s}_{\text{surprise}}}$, $\underline{\mathfrak{h}_{\text{hate}}}$, $\underline{\mathfrak{p}_{\text{plot}}}$, etc. In such an angry conscious happening, the informational interplay of the listed and other informons takes place. Thus, anger or any other emotional or cognitive conscious components can pull into the conscious orchestration several other emotional and cognitive components. In a moment of conscious happening, a distinct orchestra of informons plays it informational prelude into the next conscious happening.

The distinction to the common understanding of consciousness is in the ability of conscious informing of each single informon together with other informons. Consciousness $\underline{\mathfrak{z}}$ is just a specific informon $\underline{\mathfrak{c}_{\text{consciousness}}}$ and nothing more. In the framework of this understanding, informon $\underline{\alpha}$ means a specific name α, propagated into the possible meaning of α, concerning intentionally and attentionally the entire conscious system. As we know, $\underline{\alpha}$ is constituted by system complexity, α-intentionality, learning or informedness coming from other informons, and the own ability of decomposition or interpretation accelerating the complexity of the momentary conscious situation.

Now, let's say the said about a conscious system again by other words, with the aim, to acquire the additional clarity in informonic understanding of consciousness. The informonic concept does not stress so much the so-called hierarchically organized system being characteristic for the functioning of the human brain (informational machine). Intentionalities of complexly linked informons make the spontaneous and circular informing of the current conscious matter possible.

Consciousness about something emerges as an informational overlapping of informons, being currently relevant for a specific situation and time. A conscious event is constituted by the currently informing informons, as they occur spontaneously in a distinct time interval. The consciousness about something consists of yet consciously informing informons, transiting from a current situation to the next situation, where the next informons are coming to the conscious surface, that is, in the conscious foreground. In such a happening of conscious events, the hierarchy of the informationally relevant is not so much in the game as the informons' intentions and their interweavedness with current situations and time.

This kind of conscious mind comes fore as a spontaneous sequence of conscious events, that is, informons, depending on interior and exterior circumstances, occurring randomly but, certainly, in an intentionally unforeseeable and unpredictable way. Each individual consciousness about something confronts itself with a spontaneous stream of conscious, sub-conscious, self-conscious, and unconscious events, with an existing and entirely new experience, with learned and just learning facts, objectivities, irrationalities, beliefs, attention, cognition, and emotions. Such a conscious happening is offered through an informational overlapping and time-transiting of informons, constituting the feeling, unwinding, and revealing of consciousness about something.

6 Substructuring informon schematically, graphically, and meaningly

Capturing the inner organization of informon, its syntactic (formula-like) and semantic (perenthesis-setting) structuring, can be useful for informational decompositions, starting from the beginning informon's shell components. Decomposition of a current informon development of the form $\underline{\alpha}$ into a more sophisticated and more complex form of the emerging entity, now in a new form $\underline{\underline{\alpha}}$, hides the basic informon's intentional orientation, given to it by the initial meaning, at that time an informon's pure shell form $\underline{\alpha} \rightleftharpoons \left(\alpha; \underline{\mathcal{I}_{\underline{\alpha}}}; \underline{\mathcal{C}_{\underline{\alpha}}}; \underline{\mathcal{E}_{\underline{\alpha}}}\right)$, characterized intentionally by the meaning of its name α.

The emerging of informon is reflected in the initial (starting) pure informon's shell, at that time a $\mathfrak{I}\lceil\alpha\rceil$, and, after some decompositional steps, advanced to a more developed meaning of α, then $\underline{\alpha}$ and, finally, to the currently developed form $\underline{\alpha}$, in which the maximal amount of meaning was gathered (accumulated) up to now. This meaning, $\underline{\alpha}$, is the semantic representation of the name α, structured schematically out of the graphical representation $\mathfrak{G}\lceil\underline{\alpha}\rceil$. Informational schemes form a formula scheme system $\mathfrak{S}\lceil\Phi\rceil$, which can finally be parenthesized, $\mathfrak{P}\lceil\mathfrak{S}\lceil\Phi\rceil\rceil$, so the final and precise form of meaning concerning α is obtained, however, merely up to now.

New and new decompositions of the obtained informon's situation are only parts of an infinitesimal procedure, approaching to a more and more precise meaning of α, however never reaching a final or exact meaning, but coming near to a virtually final meaning as close as possible [7].

In this respect, an informon reflects informationally to some extent the slogan *One in all and all in one* (OA&AO). An informon propagates into the informational space gathering a specific (intentional) meaning concerning the informon's name. Using the graph of an informon, $\mathfrak{G}\lceil\underline{\alpha}\rceil$, the OA&AO problem can be made more transparent. First, let's remind that a system graph $\mathfrak{G}\lceil\Phi\rceil$ is equivalently described by the primitive system Φ'. This yields $\mathfrak{G}\lceil\Phi\rceil \rightleftharpoons \Phi'$. Second, we introduce a relation of graphical inclusion, $\prec$, with the meaning,

$$\left(\mathfrak{G}\lceil\underline{\alpha_1}\rceil \prec \mathfrak{G}\lceil\underline{\alpha_2}\rceil\right) \rightleftharpoons \left(\underline{\alpha'_1} \subset \underline{\alpha'_2}\right)$$

This means that all the primitive transitions of system $\underline{\alpha'_1}$ enter into the primitive system $\underline{\alpha'_2}$. Precisely, by a primitive transition $\alpha \models \beta$, the operand α, the subscribed operator $\models$ (e.g., $\models_{\mathrm{subscript}}$), and the operand β is meant.

The way from system $\Phi \rightleftharpoons (\varphi_1, \varphi_2, \ldots, \varphi_{n_\Phi})$ to the corresponding graph $\mathfrak{G}\lceil\Phi\rceil$, that is, to Φ', leads via the de-parenthesizing of system formulas ($\mathfrak{S}\lceil\Phi\rceil$) and, then, via the primitive partition of formula schemes into primitive system Φ'. An original formula $\varphi_i \in \Phi$ of length ℓ_{φ_i} (number of operators in φ_i), has been dissolved in the formula scheme $\mathfrak{S}\lceil\varphi_i\rceil$ and definitively lost in the possibility of $\frac{1}{\ell_{\varphi_i}}\binom{2\ell_{\varphi_i}}{\ell_{\varphi_i}}$ differently parenthesized formulas. Further, in the transition from the system scheme $\mathfrak{S}\lceil\Phi\rceil$ to Φ', the formula schemes are definitively lost. From a primitive system Φ' (graph $\mathfrak{G}\lceil\Phi\rceil$), $\prod_{i=1}^{n_\Phi}\frac{1}{\ell_{\varphi_i}}\binom{2\ell_{\varphi_i}}{\ell_{\varphi_i}}$ different formula systems can be derived for each reasonable n_Φ (a complete overlapping which covers the graph).

Let's mark the informing all (cosmos) by $\mathfrak{a}_{\mathrm{all}}$, a part of the all by $\mathfrak{p}_{\mathrm{part}}\lceil\mathfrak{a}_{\mathrm{all}}\rceil$, and the informational shell encircling the part of the all by $\mathfrak{s}_{\mathrm{shell}}\left\lceil\mathfrak{p}_{\mathrm{part}}\lceil\mathfrak{a}_{\mathrm{all}}\rceil\right\rceil$. Using the language of graphs, the following can be said:

$$\begin{aligned}
&\mathfrak{G}\lceil\underline{\alpha}\rceil \prec \mathfrak{G}\lceil\mathfrak{a}_{\mathrm{all}}\rceil;\\
&\mathfrak{G}\lceil\underline{\alpha}\rceil \prec \mathfrak{G}\left\lceil\mathfrak{p}_{\mathrm{part}}\lceil\mathfrak{a}_{\mathrm{all}}\rceil\right\rceil;\ \mathfrak{G}\left\lceil\mathfrak{p}_{\mathrm{part}}\lceil\mathfrak{a}_{\mathrm{all}}\rceil\right\rceil \prec \mathfrak{G}\lceil\mathfrak{a}_{\mathrm{all}}\rceil;\\
&\mathfrak{G}\lceil\underline{\alpha}\rceil \prec \mathfrak{G}\left\lceil\mathfrak{s}_{\mathrm{shell}}\left\lceil\mathfrak{p}_{\mathrm{part}}\lceil\mathfrak{a}_{\mathrm{all}}\rceil\right\rceil\right\rceil;\\
&\qquad\qquad \mathfrak{G}\left\lceil\mathfrak{s}_{\mathrm{shell}}\left\lceil\mathfrak{p}_{\mathrm{part}}\lceil\mathfrak{a}_{\mathrm{all}}\rceil\right\rceil\right\rceil \prec \mathfrak{G}\left\lceil\mathfrak{p}_{\mathrm{part}}\lceil\mathfrak{a}_{\mathrm{all}}\rceil\right\rceil
\end{aligned}$$

The interpretation of this situation is presented topologically [7] in Fig. 3. Namely, expressed by the corresponding

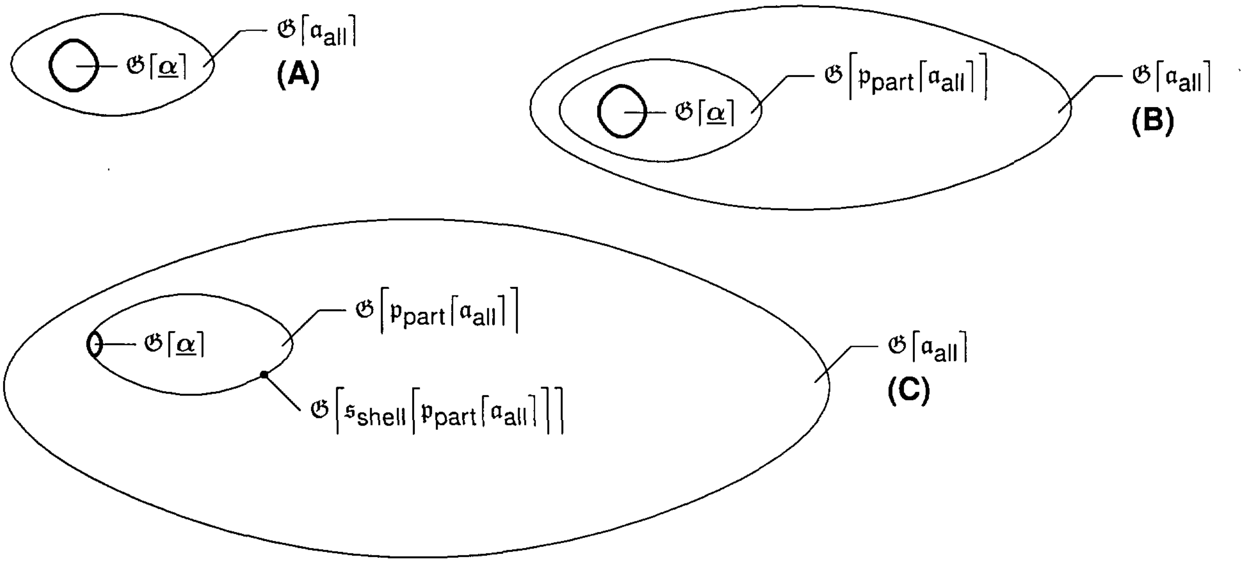

$\mathfrak{G}\lceil\mathfrak{a}_{\text{all}}\rceil$
$\mathfrak{G}\lceil\underline{\alpha}\rceil$
(A)
$\mathfrak{G}\lceil\mathfrak{p}_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil\rceil$
$\mathfrak{G}\lceil\underline{\alpha}\rceil$
$\mathfrak{G}\lceil\mathfrak{a}_{\text{all}}\rceil$
(B)
$\mathfrak{G}\lceil\mathfrak{p}_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil\rceil$
$\mathfrak{G}\lceil\underline{\alpha}\rceil$
$\mathfrak{G}\lceil\mathfrak{s}_{\text{shell}}\lceil\mathfrak{p}_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil\rceil\rceil$
$\mathfrak{G}\lceil\mathfrak{a}_{\text{all}}\rceil$
(C)

Figure 3: *The solution as an informon's meaning, that is, the up-to-the-minute developed informon—the one—represented graphically:* ***(A)*** *The informon as a possibility of the all.* ***(B)*** *The informon as a possibility of a part of the all.* ***(C)*** *The informon as a possibility of a shell surrounding a part of the all.*

primitive formula systems, there is

$$\begin{aligned}
&\underline{\alpha'} \subset \mathfrak{a}'_{\text{all}};\\
&\underline{\alpha'} \subset \mathfrak{p}'_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil;\ \mathfrak{p}'_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil \subset \mathfrak{a}'_{\text{all}};\\
&\underline{\alpha'} \subset \mathfrak{s}'_{\text{shell}}\left\lceil\mathfrak{p}_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil\right\rceil;\\
&\qquad\qquad \mathfrak{s}'_{\text{shell}}\left\lceil\mathfrak{p}_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil\right\rceil \subset \mathfrak{p}'_{\text{part}}\lceil\mathfrak{a}_{\text{all}}\rceil
\end{aligned}$$

This set-theoretical presentation helps to make the graphical situation in Fig. 3 definitely transparent.

The meaning of something means to have an informational expression for something called the informational solution on something. For instance, in searching a solution of something α, the at-hand solution is a sort of decomposition, in general, $\Delta\lceil\alpha\rceil$. However, out of a formula φ or formula system Φ in which α occurs, the solution concerning α can be explicitly expressed. Let $\mathfrak{P}$ mark the formula parenthesizing (setting of parenthesis pairs), $\mathfrak{R}$ the operand rotation in a circular formula, $\mathfrak{S}$ the schematizing (deleting the parenthesis pairs) of a formula, and $\overset{\circ}{\varphi}_{\triangleright}\lfloor\alpha_1,\ldots,\alpha_{i-1},\alpha_i,\alpha_{i+1},\ldots,\alpha_n\rfloor$ a circular formula. In this case it is possible to solve formula $\overset{\circ}{\varphi}_{\triangleright}\lfloor\alpha_1,\ldots,\alpha_{i-1},\alpha_i,\alpha_{i+1},\ldots,\alpha_n\rfloor$ on each of its operands $\alpha_1,\ldots,\alpha_{i-1},\alpha_i,\alpha_{i+1},\ldots,\alpha_n$, that is, to get n different solutions, expressing the meaning of each particular operand. Then, for a solution, using the operand rotation principle, there is

$$\begin{aligned}
&\overset{\circ}{\varphi}_{\triangleright}\lfloor\alpha_i,\alpha_{i+1},\ldots,\alpha_n,\alpha_1,\ldots,\alpha_{i-1}\rfloor \rightleftharpoons\\
&\quad\mathfrak{P}\left\lceil\mathfrak{R}\left\lceil\mathfrak{S}\left\lceil\overset{\circ}{\varphi}_{\triangleright}\lfloor\alpha_1,\ldots,\alpha_{i-1},\alpha_i,\alpha_{i+1},\ldots,\alpha_n\rfloor\right\rceil\right\rceil\right\rceil;\\
&i = 1,2,\ldots,n
\end{aligned}$$

It is understood that the solution upon operand α_1 is already $\overset{\circ}{\varphi}_{\triangleright}\lfloor\alpha_1,\ldots,\alpha_{i-1},\alpha_i,\alpha_{i+1},\ldots,\alpha_n\rfloor$, obtained by an operand α_1 decomposition, for instance, etc.

A solution upon an operand can be obtained from a complex formula system taking into account all its formulas. That what comes out of such a consideration are informons $\underline{\alpha'_1},\ldots,\underline{\alpha'_{i-1}},\underline{\alpha'_i},\underline{\alpha'_{i+1}},\ldots,\underline{\alpha'_n}$.

7 The pure informon

The pure informon, $\underline{\mathfrak{i}_{\text{informon}}}$, is the searching, propagating, and determining the meaning of the newly coined and introduced word *informon* in the informational space. By the acquired meaning, the concept of informon is fortified and made familiar and understood in communities of researchers, scientists, and publicists. For instance, the new word can enter into an English or other dictionary and be regularly explained together with its etymology.

Additionally, the concept of informon can be grasped formally by informational formalism. In fact, at the beginning, we concentrate on a name (phrase, named informational entity) marking it by α. What will be the informon named α and how will it develop informationally? According to the discussed definitions, initially,

$$\underline{\mathfrak{i}_{\text{informon}}\lceil\alpha\rceil} \rightleftharpoons \begin{pmatrix} \mathfrak{i}_{\text{informon}}\lceil\alpha\rceil;\\ \underline{\underline{\mathcal{I}_{\mathfrak{i}_{\text{informon}}\lceil\alpha\rceil}}};\\ \underline{\underline{\mathcal{C}_{\mathfrak{i}_{\text{informon}}\lceil\alpha\rceil}}};\\ \underline{\underline{\mathcal{E}_{\mathfrak{i}_{\text{informon}}\lceil\alpha\rceil}}} \end{pmatrix}$$

In general, an initially decomposed informon, e.g., $\mathfrak{I}\lceil\mathfrak{i}_{\text{informon}}\lceil\alpha\rceil\rceil$, attains the required complexity through the linkage to other informons. For the informonic components of the pure informon there is, evidently, in case of intentional component,

$$\underline{\mathcal{I}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}} \rightleftharpoons \begin{pmatrix} \mathcal{I}_{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}; \\ \underline{\mathcal{I}_{\underline{\mathcal{I}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}}; \\ \underline{\mathcal{C}_{\underline{\mathcal{I}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}}; \\ \underline{\mathcal{E}_{\underline{\mathcal{I}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}} \end{pmatrix}$$

in case of counter-intentional component,

$$\underline{\mathcal{C}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}} \rightleftharpoons \begin{pmatrix} \mathcal{C}_{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}; \\ \underline{\mathcal{I}_{\underline{\mathcal{C}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}}; \\ \underline{\mathcal{C}_{\underline{\mathcal{C}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}}; \\ \underline{\mathcal{E}_{\underline{\mathcal{C}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}} \end{pmatrix}$$

and in case of cognitive component,

$$\underline{\mathcal{E}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}} \rightleftharpoons \begin{pmatrix} \mathcal{E}_{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}; \\ \underline{\mathcal{I}_{\underline{\mathcal{E}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}}; \\ \underline{\mathcal{C}_{\underline{\mathcal{E}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}}; \\ \underline{\mathcal{E}_{\underline{\mathcal{E}_{\underline{\mathfrak{i}_{\mathrm{informon}}\lceil\alpha\rceil}}}}} \end{pmatrix}$$

The next, being extremely interesting, is the intentional contents of each of informonic components. Beside the informons' names, the informing of intentional contents concerns informons and their components. The general properties of each informon and its components are the following:

1. the existence of the name-α-specific intention $\mathfrak{i}_{\mathrm{intention}}\lceil\alpha\rceil$, the name-$\alpha$-specific counter-intention $\mathfrak{c}_{\mathrm{counter\text{-}intention}}\lceil\alpha\rceil$, and the name-$\alpha$-specific intentional cognition $\mathfrak{i}_{\mathrm{intentional\_cognition}}\lceil\alpha\rceil$;

2. the development of the name-α-specific intentional complexity $\mathfrak{c}_{\mathrm{complexity}}\lceil\mathfrak{i}_{\mathrm{intention}}\lceil\alpha\rceil\rceil$, the name-$\alpha$-specific counter-intentional complexity $\mathfrak{c}_{\mathrm{complexity}}\lceil\mathfrak{c}_{\mathrm{counter\text{-}intention}}\lceil\alpha\rceil\rceil$, and the name-$\alpha$--specific complexity of intentional cognition $\mathfrak{c}_{\mathrm{complexity}}\left\lceil\mathfrak{i}_{\mathrm{intentional\_cognition}}\lceil\alpha\rceil\right\rceil$; and

3. the emerging of the name-α-specific intentional decomposition $\mathfrak{J}\lceil\mathfrak{i}_{\mathrm{intention}}\lceil\alpha\rceil\rceil$, the name-$\alpha$-specific counter-intentional decomposition $\mathfrak{J}\lceil\mathfrak{c}_{\mathrm{counter\text{-}intention}}\lceil\alpha\rceil\rceil$, and the name-$\alpha$-specific complexity of intentional cognition $\mathfrak{J}\left\lceil\mathfrak{i}_{\mathrm{intentional\_cognition}}\lceil\alpha\rceil\right\rceil$.

Within this circle of emergence, the intention is refined, stepping into greater details, and complexity is enlarged, being more and more interweaved into the informational space. The consequence is the enriching of meaning concerning the informon's name α, that is, the name-specific intentionality $\mathfrak{i}_{\mathrm{intention}}\lceil\alpha\rceil$. For an informon $\underline{\underline{\alpha}} \rightleftharpoons \left(\alpha; \underline{\mathcal{I}_{\underline{\alpha}}}; \underline{\mathcal{C}_{\underline{\alpha}}}; \underline{\mathcal{E}_{\underline{\alpha}}}\right)$, there is, evidently,

$$\underline{\mathcal{I}_{\underline{\alpha}}} \rightleftharpoons \begin{pmatrix} \mathfrak{i}_{\mathrm{intention}}\lceil\alpha\rceil;\ \mathfrak{c}_{\mathrm{complexity}}\lceil\mathfrak{i}_{\mathrm{intention}}\lceil\alpha\rceil\rceil; \\ \mathfrak{J}\lceil\mathfrak{i}_{\mathrm{intention}}\lceil\alpha\rceil\rceil;\ \ldots \end{pmatrix}$$

$$\underline{\mathcal{C}_{\underline{\alpha}}} \rightleftharpoons \begin{pmatrix} \mathfrak{c}_{\mathrm{counter\text{-}intention}}\lceil\alpha\rceil; \\ \mathfrak{c}_{\mathrm{complexity}}\lceil\mathfrak{c}_{\mathrm{counter\text{-}intention}}\lceil\alpha\rceil\rceil; \\ \mathfrak{J}\lceil\mathfrak{c}_{\mathrm{counter\text{-}intention}}\lceil\alpha\rceil\rceil;\ \ldots \end{pmatrix}$$

$$\underline{\mathcal{E}_{\underline{\alpha}}} \rightleftharpoons \begin{pmatrix} \mathfrak{i}_{\mathrm{intentional\_cognition}}\lceil\alpha\rceil; \\ \mathfrak{c}_{\mathrm{complexity}}\left\lceil\mathfrak{i}_{\mathrm{intentional\_cognition}}\lceil\alpha\rceil\right\rceil; \\ \mathfrak{J}\left\lceil\mathfrak{i}_{\mathrm{intentional\_cognition}}\lceil\alpha\rceil\right\rceil;\ \ldots \end{pmatrix}$$

8 Conclusion

The concept of informon calls for a different, in some way new understanding of consciousness or, precisely, conscious system. It comes close to that what a conscious observer experiences on the own consciousness, its happening in a time slice and in the sequence of conscious time slices. It enables a straightforward reasoning in the direction of artificially conscious systems constituted by informons. Informon seems to be finally just an informational entity depending on complexity, intention, learning, and decomposition concerning meaning of the intentional name. The required complexity seems to be a problem of future technological development, and the remaining necessary requirements are already in the visible scope of mastering them philosophically and technically.

After all, a meme [1] seems to be nothing else than a specific self-replicating informonic entity[2] in individual brains which breaks down the reasonable immunity of conscious systems against an informational excess, exaggeration, plethora, or surfeit. By the philosophy of informon, a concrete meme can be studied as an authentically conscious sort of informon, in an innovative general and meaning-specific way. E.g., totalitarian memes have informed as politically local informons.

Regarding the building block of conscious system, called informon, it might be reasonable to understand it as a consequence of macroscopic quantum nature of consciousness—which is pointed out in Peruš [5], as well as in Raković [6], from where it follows that informons might be related to quantum eigenstates of the conscious macroquantum sysrem. The whole story should be then related closely to quantum algebra, and naturally explains the nonlocal (holistic) aspect of consciousness.

References

[1] BLACKMORE, S. 1999. *The Meme Machine.* Oxford University Press. New York.

[2] An informon informs in a self-replicating way by informational decomposition considering the informon's intention.

[2] BUTTAZZO, G.C. 2001. *Artificial consciousness: Utopia or real possibility?* IEEE Computer 34:7:24–30.

[3] KURZWEIL, R. 1999. *The Age of Spiritual Machines. When Computers Exceed Human Intelligence.* Penguin Books. New York.

[4] MORAVEC, H. 1999. *Robot. Mere Machine to Transcendent Mind.* Oxford University Press. New York.

[5] PERUŠ, M. 2001. *Image processing and becoming conscious of its results.* Informatica 25:575–592.

[6] RAKOVIĆ, D. 2002. *Hopfield-like quantum associative neural networks and (quantum) holistic psychosomatic implications.* B. Reljin, Ed.: Proc. 6-th NEUREL. IEEE Yugoslavia Section. Belgrade.

[7] ŽELEZNIKAR, A.P. 1998. *Topological informational spaces.* Informatica 22:287–308.

[8] ŽELEZNIKAR, A.P. 2002. *An Introduction to Artificial Consciousness. An Informational Approach, Formalism, and Implementation.* Available free in PDF/Acrobat at location http:// www.artifico.org.

[9] ŽELEZNIKAR, A.P. 2002. *Informon—ein bewußter Bauteil des Bewußtseins.* Grundlagenstudien aus Kybernetik und Geisteswissenschaft —Humankybernetik 43 (in press).

A List Scheduling Heuristic for Allocating the Task Graph to Multiprocessors

Janez Brest and Viljem Žumer
University of Maribor
Faculty of Electrical Engineering and Computer Science
Smetanova 17, 2000 Maribor, Slovenia
E-mail: janez.brest@uni-mb.si, http://marcel.uni-mb.si/janez

Keywords: parallel processing, compiler, static scheduling

Received: July 2, 2002

In this paper we propose a new static scheduling algorithm for allocating the task graph without communication costs to fully connected multiprocessors. A global comparison is carried out for the proposed algorithm and three reported scheduling algorithms. The proposed algorithm outperforms the previous algorithms in terms of the generated schedule length using Standard Task Graph set.

1 Introduction

To efficiently execute a program on a multiprocessor system [11, 19, 20, 8, 18], it is essential to solve a minimum execution time multiprocessor scheduling problem [16, 13, 14, 2, 4, 5], which determines how to assign a set of tasks to processors and in what order those tasks should be executed to obtain the minimum execution time. The tasks can then be scheduled to the processors for execution by using a suitable scheduling algorithm, static in compile-time or dynamic in run-time [7, 9, 3]. The optimal static scheduling, except for a few highly simplified cases, is an NP-complete problem. Thus, heuristic approaches are generally sought to tackle the problem. Traditional static scheduling algorithms attempt to minimize the schedule length through iterative local minimization of the start times of individual tasks. On the other hand for example the Dynamic Level Scheduling (DLS) algorithm dynamically selects tasks during the scheduling process [15]. As optimal scheduling of tasks is a strong NP-hard problem, many heuristic algorithms have been introduced in the literature [6].

In this paper we proposed a low time complexity multiprocessor static scheduling algorithm called MCP/CLR without communication costs, which is based on critical path (CP) algorithm, such as, for example, the MCP [21] algorithm. It generates high quality scheduling solutions.

The remaining paper is organized as follows: In the next section, we present a brief overview of various approaches that have been proposed for the DAG scheduling problem. In Section 3, we present the proposed algorithm, and discuss its design principles. We present the experimental results in Section 4, and conclude the paper with some final remarks in Section 5.

2 The Multiprocessor Scheduling Problem

In static scheduling, a parallel program is presented by a directed acyclic graph (DAG) [19]. In a DAG, $G = (V, E)$, V is a set of v nodes, representing the tasks, and E is a set of e directed edges, representing the communication messages. Edges in a DAG are directed and, thus, capture the precedence constraints among the tasks. The cost of node n_i, denoted as $w(n_i)$, represents the computation cost of the task. The cost of the edge, emerges from the source node n_i and incidents on the destination node n_j, denoted by c_{ij}, represents the communication cost of the message. The source node of an edge is called a parent node, while the destination node is called a child node. A node with no parent is called an entry node and a node with no child is called an exit node. A node can only start execution after it has gathered all of the messages from its parent nodes. The *b-level* of a node is the length (sum of the computation costs only) of the longest path from this node to an exit node. The *t-level* of a node is the length of the longest path from an entry node to this node (excluding the cost of this node).

The objective of scheduling is to minimize the schedule length, which is defined as the maximum finish time of all the nodes, by properly assigning tasks to processors such that the precedence constraints are preserved.

The existing scheduling algorithms are classified into four categories by Ahmad and Kwok [2, 14]:

1. Bounded Number of Processors (BNP) Scheduling: A BNP algorithm schedules a DAG to a limited number of processors directly. The processors are assumed to be fully connected without any regard to link contention and scheduling of messages. The proposed algorithm belongs to this class.

2. Unbounded Number of Clusters (UNC) Scheduling: An UNC algorithm schedules a DAG to an unbounded

number of clusters. The clusters generated by these algorithms may be mapped onto the processors using a separate mapping algorithm. These algorithms assume the processors to be fully connected.

3. Arbitrary Processor Network (APN) Scheduling: An APN algorithm performs scheduling and mapping on an architecture in which the processors are connected via a network topology. An APN algorithm also explicitly schedules communication messages on the network channels, taking care of the link contention factor.

4. Task-Duplication-Based (TDB) Scheduling: A TDB algorithm duplicates tasks in order to reduce the communication overhead. Duplication, however, can be used in any of the other three classes of algorithms.

For our purpose, we will compare the proposed algorithm with three other BNP scheduling algorithms.

In a traditional scheduling algorithm, the scheduling list is statically constructed before node allocation begins, and, more importantly, the sequencing in the list is not modified.

The Earliest Task First (ETF) algorithm [10] uses static node priorities and assumes only a bounded number of processors [16, 17]. The High Level First with Estimated Time (HLFET) algorithm [1] assigns the nodes in a DAG to the processors, level by level.

Similar to the ETF and HLFET algorithms, the Modified Critical Path (MCP) algorithm [21] constructs a list of tasks before the scheduling process starts. The MCP algorithm uses the ALAP (As-Late-As-Possible) start time of a node as the scheduling priority. The MCP algorithm first computes the ALAP times of all the nodes, then constructs a list of nodes in ascending order of ALAP times. Ties are broken by considering the ALAP times of the children of a node. The MCP algorithm then schedules the nodes on the list one by one so that a node is scheduled to a processor that allows the earliest start time using the insertion approach. The MCP algorithm looks for an idle time slot for a given node. The algorithm is briefly described in Figure 1 [21, 16, 14]. The complexity of the MCP algorithm is $O(v^2 \log v)$.

3 The Heuristic Algorithm

In this section we discuss some of the principles used in the design of proposed algorithm. To minimize the final schedule length, we select a node as it is selected in the MCP algorithm. At each step of the scheduling process, the first node is removed from the list of nodes (the list of nodes is sorted in increasing lexicographical order of the latest possible start times) and it is scheduled to a processor. While we are able to identify a selected node, we still need a method to select an appropriate processor for scheduling that node into the most suitable idle time slot. At each step, the algorithm needs to find the most suitable processor which contains the most suitable place in time for a selected node.

```
(1) Compute the ALAP time of each node.
(2) For each node, create a list which consists of the
    ALAP times of the node itself and all its children
    in descending order.
(3) Sort these lists in ascending lexicographical or-
    der. Create a node list according to this order.
Repeat
(4) Schedule the first node in the node list to a pro-
    cessor that allows the earliest execution, using the
    insertion approach.
(5) Remove the node from the node list.
Until the node list is empty.
```

Figure 1: The MCP algorithm.

The MCP algorithm schedules the selected node to a processor that allows for the earliest start time. The proposed algorithm has another processor selection criteria and they are described as follows.

3.1 The MCP/CLR Algorithm

```
Build_ALAP();
Sort_ALAP();
// v is number of tasks
for (i = 0; i < v; i++)
{
   t_i = EST(ALAP(n_i));
   if a processor j exists where SL_j(i) ≤ t_i
   then
      schedule node n_i to a processor j where
            SL_j(i) − t_i is minimal
   else
      schedule node n_i to a processor that
      allows the earliest execution
}
```

Figure 2: The MCP/CLR algorithm.

The function *Build_ALAP*() computes the ALAP time of each node and creates a list, which consists of the ALAP times of the node itself and all its children in descending order. Function *Sort_ALAP*() sorts these lists in ascending lexicographical order as in the MCP algorithm.

Assumed that, in the scheduling process there are already scheduled $i-1$ nodes. Next selected node is n_i. $SL_j(i)$ is the schedule length of the step i of the scheduling process on the processor j. The MCP/CLR (*MPC/Close-Left-Right*) algorithm (see Fig. 2) tries to find a processor j for the selected node n_i. It is needed to distinguish two cases of the processor selection step. If a processor exists, say j, which satisfy that $SL_j(i)$ is less or equal to the earliest start time (EST) of the selected node n_i, our algorithm assigns the selected node n_i to the processor j with the smallest value $SL_j(i) - t_i$. Otherwise it assigns the selected node n_i to a processor that allows the earliest execution (like the MCP algorithm), using non-insertion approach. The complexity of the MCP/CLR algorithm is $O(v^2 \log v)$, too.

3.2 Scheduling Example

In this section, we present an example to demonstrate the operation of the proposed algorithm using the task graph shown in Fig. 3. The task graph was drawn using the Graphlet Tool (*http://www.fmi.uni-passau.de/Graphlet*). The schedules of the algorithms are shown in Fig. 4. The entry and exit node are dummy. The MCP algorithm creates a list of edges and schedules the task graph onto the multiprocessor machine with 2 processors (processing elements) in the order: $n_1, n_2, n_5, n_3, n_8, n_7, n_4, n_{11}, n_{10}, n_9, n_6, n_{12}$. The HLFET and MCP/CLR schedule the nodes in the same order as the MCP algorithm. The ETF algorithm schedules the nodes in the order: $n_1, n_2, n_5, n_3, n_4, n_7, n_8, n_{10}, n_6, n_9, n_{11}, n_{12}$. The order of nodes n_4, n_7, n_8 and the processor selection during the scheduling process, have caused different schedules of the task graph, and therefore also different schedule lengths.

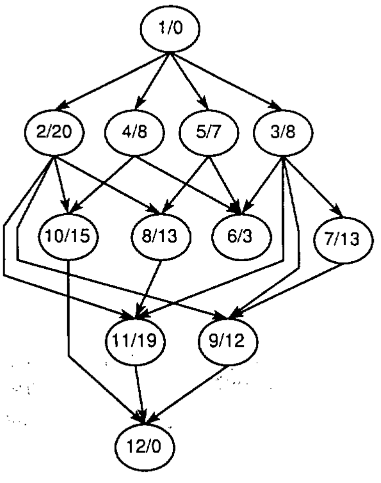


Figure 3: An example of a task graph with 12 nodes.

4 Results

In this section, we present the performance results of the proposed algorithm and compare them with the results of the HLFET, ETF and MCP algorithms.

We have implemented the scheduling algorithms on a SUN workstation using C/C++. They were evaluated by using a Standard Task Graph set: *http://www.kasahara.elec.waseda.ac.jp/schedule/*. The Standard Task Graph set has 900 task graphs with 50 to 2700 tasks.

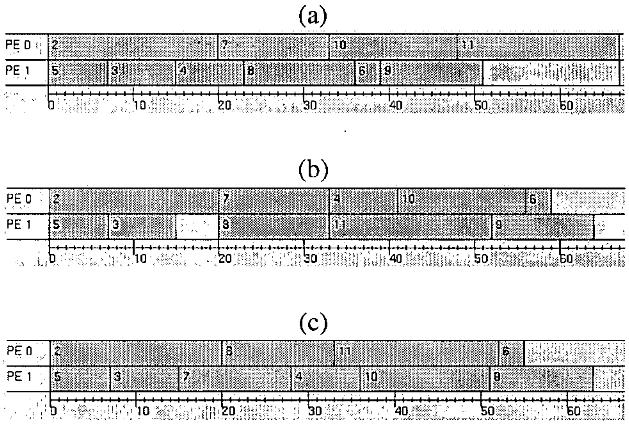


Figure 4: The schedules of the task graph on Fig. 3 generated by: (a) ETF algorithm (schedule length = 67 time units); (b) HLFET and MCP algorithms (schedule length = 64 time units); and (c) MCP/CLR algorithm (schedule length = 63 time units).

Table 3: Number of times the optimal schedule is found, and a global error

Algorithm	Optimal schedule	%	Global error
ETF	33	12.94	5577
HLFET	50	19.61	3189
MCP	57	22.53	1531
MCP/CLR	167	65.49	257

The results obtained in our experiments are shown in Table 1. The second and third columns indicate the name of the task graph instance and number of nodes, respectively. In next four columns results of the schedule length for the all of algorithms are shown, respectively. In the last column the optimal schedule length value is shown. If the optimal schedule is found, the schedule length value is boldface. For some problem instances, the optimal schedule length is not known.

In order to rank all the algorithms in terms of the schedule lengths, we made a global comparison [17]. We observed the number of times each algorithm performed better, worse or the same compared to each of the other algorithms. This comparison is presented in Fig. 5, where some boxes have the left and the right side. Each left side of the box compares two algorithms – the algorithm on the left side and the algorithm on the top. Each left side of the box contains three numbers preceded by ">", "<", and "=" signs which indicate the number of times the algorithm on the left performed better, worse, or the same, respectively, compared to the algorithm shown on the top. Each comparison is based on the total of 300 task graphs. Each right side of the box contains the number of times when one of algorithms, the algorithm on the left side or the algorithm on the top, find the optimal schedule length. Optimal sched-

Table 1: Schedule results of 50 task graph instances

	Graph	#Nodes	ETF	HLFET	MCP	MCP/CLR	Optimum
1	proto000.stg	452	**537**	**537**	**537**	**537**	537
2	proto001.stg	473	1191	1179	1179	**1178**	1178
3	proto002.stg	499	357	363	355	343	341
4	proto003.stg	164	**556**	**556**	**556**	**556**	556
5	proto004.stg	457	267	238	234	222	—
6	proto005.stg	404	758	749	**742**	**742**	742
7	proto006.stg	273	171	154	149	142	—
8	proto007.stg	499	492	**489**	**489**	**489**	489
9	proto008.stg	399	578	582	579	572	571
10	proto009.stg	438	**625**	**625**	**625**	**625**	625
11	proto010.stg	539	351	338	338	**334**	334
12	proto011.stg	759	513	496	494	484	—
13	proto012.stg	939	1804	1795	1795	**1793**	1793
14	proto013.stg	799	698	688	685	682	681
15	proto014.stg	636	523	520	516	509	—
16	proto015.stg	712	513	513	501	**491**	491
17	proto016.stg	641	1016	1026	1022	1006	—
18	proto017.stg	722	487	475	473	463	—
19	proto018.stg	730	704	706	704	701	700
20	proto019.stg	617	683	682	674	668	667
21	proto020.stg	1104	1523	1514	1511	1505	1504
22	proto021.stg	1145	644	632	616	**605**	605
23	proto022.stg	1189	1625	1620	1617	1610	1609
24	proto023.stg	1353	1619	1628	1624	1614	1612
25	proto024.stg	1218	1295	1291	1289	1283	1281
26	proto025.stg	1258	1193	1198	1194	1191	1188
27	proto026.stg	1239	1509	1502	1501	**1500**	1500
28	proto027.stg	1055	2003	2001	2001	2001	2000
29	proto028.stg	1424	1538	1506	1506	**1504**	1504
30	proto029.stg	1341	845	**830**	**830**	**830**	830
31	proto280.stg	1668	2821	2809	2806	**2800**	2800
32	proto281.stg	1622	977	970	921	897	896
33	proto282.stg	1793	3131	3131	3127	**3123**	3123
34	proto283.stg	1591	2479	2429	2428	2425	2422
35	proto284.stg	1703	4499	4458	4447	**4444**	4444
36	proto285.stg	1766	1333	1304	1285	**1268**	1268
37	proto286.stg	1703	945	940	930	**918**	918
38	proto287.stg	1615	1423	1422	1381	1360	1353
39	proto288.stg	1672	1938	1944	1935	1926	1925
40	proto289.stg	1642	1968	1922	1917	**1915**	1915
41	proto290.stg	2133	1453	1450	1438	**1428**	1428
42	proto291.stg	2122	2611	2605	2597	**2591**	2591
43	proto292.stg	2333	8022	8012	8011	**8009**	8009
44	proto293.stg	2089	1516	1477	1474	**1472**	1472
45	proto294.stg	2014	1336	1307	1273	1259	1257
46	proto295.stg	2168	1349	1329	1321	**1318**	1318
47	proto296.stg	2162	2589	2590	2575	**2563**	2563
48	proto297.stg	2136	7708	7710	7710	**7703**	7703
49	proto298.stg	2399	2492	2476	2474	**2471**	2471
50	proto299.stg	2205	1171	1153	1147	1142	1141

Table 2: Schedule length with respect to the optimal solution

	Quality of the solution (Error)	ETF	HLFET	MCP	MCP/CLR
1	0% (optimum)	33	50	57	167
2	< 5%	178	182	195	88
3	5% - 10%	30	21	3	0
4	10% - 15%	11	2	0	0
5	15% - 20%	3	0	0	0
6	Optimum not known	45	45	45	45
	Total	300	300	300	300

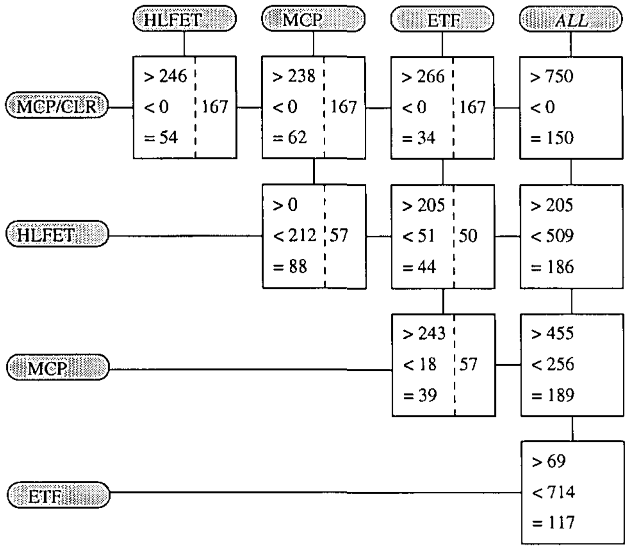


Figure 5: A global comparison of four algorithms in terms of better, worse, and equal performance.

ule lengths are known for 255 of all 300 task graphs. They were computed on a parallel machine using the ISH algorithm [13, 12]. For example, the MCP/CLR algorithm performed better than the MCP algorithm in 238 cases, never performed worse, and performed the same in 62 cases. The MCP/CLR algorithm or the MCP algorithm or both of them found optimal solution of the schedule length in 167 cases. An additional box for each algorithm compares that algorithm with all other algorithms combined.

The experimental results of the quality of the schedule length are summarized in Table 2. For example, the MCP/CLR algorithm found the optimal schedule length in 167 cases and, additionally, the solution within 5% in 88 cases.

Table 3 shows number of times the algorithm has found the optimal schedule, and global error which is defined as difference between the sum of all the optimal schedule values and the sum of all the schedule values generated by the algorithm.

It can be noticed that the proposed MCP/CLR algorithm outperformed three other well known algorithms. Based on these experiments, all the algorithms can be sorted in the following order: MCP/CLR, MCP, HLFET and ETF. The same order of the MCP and ETF algorithms can be found in [17], where communications are also assumed among the tasks.

5 Conclusion

This paper presents the static task scheduling algorithm which can schedule directed acyclic graphs (DAGs) with a complexity of $O(v^2 \log v)$, where v is the number of tasks in the DAG. The algorithm schedules the tasks and it is suitable for the graphs with arbitrary computation and without communication costs, and is applicable to the system with homogeneous fully connected processors. The performances of the proposed algorithm has been observed by comparing it with other existing bounded number of processor (BNP) scheduling algorithms in terms of the schedule length.

References

[1] T. L. Adam, K. M. Chandy, and J. R. Dickson. A comparison of list schedules for parallel processing systems. *Communications of the ACM*, 17(12):685–690, December 1974.

[2] I. Ahmad and Y.-K. Kwok. On parallelizing the multiprocessor scheduling problem. *IEEETPDS: IEEE Transactions on Parallel and Distributed Systems*, 10, 1999.

[3] J. Brest, V. Žumer, and M. Ojsteršek. Dynamic scheduling on a network heterogeneous computer system. *LNCS 1557*, pages 584–585, 1999.

[4] J. Brest and V. Žumer. A Performance Evaluation of List Scheduling Heuristics for Task Graphs without Communication Costs. Proceedings of the International Workshop on Parallel Processing (ICPP'00), pages 421–428, 2000.

[5] J. Brest, J. Jejčič, A. Vreže and V. Žumer. An Approximation Algorithm for the Static Task Schedul-

ing on Multiprocessors. VECPAR'2000 4th International Meeting on Vector and Parallel Processing, Vol. 1, pages 46–56, 2000.

[6] D. Darbha and D. P. Agrawal: Optimal scheduling algorithm for distributed-memory machines. *IEEETPDS: IEEE Transactions on Parallel and Distributed Systems*, 9, 1998.

[7] M. M. Eshagian, editor. *Heterogeneous Computing*. Artech House, Inc., Norwood, MA 02062, ISBN 0-89006-552-7, 1996.

[8] I. Foster. *Designing and Building Parallel Programs*. Addison-Wesley, ISBN 0-201-57594-9, 1995.

[9] E. Haddan. Load Balancing and Scheduling in Network Heterogeneous Computing. In M. M. Eshagian, editor, *Heterogeneous Computing*, pages 224–276, Norwood, MA 02062, ISBN 0-89006-552-7, 1996. Artech House, Inc.

[10] J. J. Hwang, Y.-C. Chow, F. D. Anger, and C.-Y. Lee. Scheduling precedence graphs in systems with interprocessor communication times. *SIAM Journal on Computing*, 18(2):244–257, April 1989.

[11] K. Hwang and Z. Xu. *Advanced Computer Architecture: Technology, Architecture, Programming*. McGraw-Hill, New York, 1998.

[12] H. Kasahara, H. Honda, and S. Narita. Parallel processing of near fine grain tasks using static scheduling on OSCAR (optimally scheduled advanced multiprocessor). In IEEE, editor, *Proceedings, Supercomputing '90: November 12–16*, pages 856–864. IEEE Computer Society Press, 1990.

[13] H. Kasahara and S. Narita. Practical multiprocessor scheduling algorithms for efficient parallel processing. *IEEE Trans. on Computers*, 33(11):1023, November 1984.

[14] Y.-K. Kwok. *High-Performace Algorithms for Compile-Time Scheduling of Parallel Processors*. PhD thesis, The Hong Kong University of Science and Technology, 1997.

[15] Y.-K. Kwok and I. Ahmad. FASTEST: A practical low-complexity algorithm for compile-time assignment of parallel programs to multiprocessors. *IEEE Transactions on Parallel and Distributed Systems*, 10(2):147–159, February 1999.

[16] Y.-K. Kwok and I. Ahmad. Parallel program scheduling technique. In Buyya Raykumar, editor, *High Performance Cluster Computing: Architectures and Systems*. Prentice Hall - PTR, NJ, USA, 1999.

[17] Y.-K. Kwok and I. Ahmad. Dynamic critical-path scheduling: An effective technique for allocating task graphs to multiprocessors:. *IEEE Transactions on Parallel and Distributed Systems*, 7(5):506–521, May 1996.

[18] M. Quinn. *Parallel Computing: Theory and Practice*. McGraw-Hill, 1994.

[19] B. Raykumar, editor. *High Performance Cluster Computing: Architectures and Systems*. Prentice Hall - PTR, NJ, USA, 1999.

[20] B. Wilkinson and M. Allen. *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*. Prentice-Hall, Englewood Cliffs, NJ 07632, USA, 1998.

[21] M.-Y. Wu and D. D. Gajski. Hypertool: A programming aid for message-passing systems. *IEEE Transactions on Parallel and Distributed Systems*, 1(3):330–343, July 1990.

JOŽEF STEFAN INSTITUTE

Jožef Stefan (1835-1893) was one of the most prominent physicists of the 19th century. Born to Slovene parents, he obtained his Ph.D. at Vienna University, where he was later Director of the Physics Institute, Vice-President of the Vienna Academy of Sciences and a member of several scientific institutions in Europe. Stefan explored many areas in hydrodynamics, optics, acoustics, electricity, magnetism and the kinetic theory of gases. Among other things, he originated the law that the total radiation from a black body is proportional to the 4th power of its absolute temperature, known as the Stefan–Boltzmann law.

The Jožef Stefan Institute (JSI) is the leading independent scientific research institution in Slovenia, covering a broad spectrum of fundamental and applied research in the fields of physics, chemistry and biochemistry, electronics and information science, nuclear science technology, energy research and environmental science.

The Jožef Stefan Institute (JSI) is a research organisation for pure and applied research in the natural sciences and technology. Both are closely interconnected in research departments composed of different task teams. Emphasis in basic research is given to the development and education of young scientists, while applied research and development serve for the transfer of advanced knowledge, contributing to the development of the national economy and society in general.

At present the Institute, with a total of about 700 staff, has 500 researchers, about 250 of whom are postgraduates, over 200 of whom have doctorates (Ph.D.), and around 150 of whom have permanent professorships or temporary teaching assignments at the Universities.

In view of its activities and status, the JSI plays the role of a national institute, complementing the role of the universities and bridging the gap between basic science and applications.

Research at the JSI includes the following major fields: physics; chemistry; electronics, informatics and computer sciences; biochemistry; ecology; reactor technology; applied mathematics. Most of the activities are more or less closely connected to information sciences, in particular computer sciences, artificial intelligence, language and speech technologies, computer-aided design, computer architectures, biocybernetics and robotics, computer automation and control, professional electronics, digital communications and networks, and applied mathematics.

The Institute is located in Ljubljana, the capital of the independent state of **Slovenia** (or S♡nia). The capital today is considered a crossroad between East, West and Mediterranean Europe, offering excellent productive capabilities and solid business opportunities, with strong international connections. Ljubljana is connected to important centers such as Prague, Budapest, Vienna, Zagreb, Milan, Rome, Monaco, Nice, Bern and Munich, all within a radius of 600 km.

In the last year on the site of the Jožef Stefan Institute, the Technology park "Ljubljana" has been proposed as part of the national strategy for technological development to foster synergies between research and industry, to promote joint ventures between university bodies, research institutes and innovative industry, to act as an incubator for high-tech initiatives and to accelerate the development cycle of innovative products.

At the present time, part of the Institute is being reorganized into several high-tech units supported by and connected within the Technology park at the Jožef Stefan Institute, established as the beginning of a regional Technology park "Ljubljana". The project is being developed at a particularly historical moment, characterized by the process of state reorganisation, privatisation and private initiative. The national Technology Park will take the form of a shareholding company and will host an independent venture-capital institution.

The promoters and operational entities of the project are the Republic of Slovenia, Ministry of Science and Technology and the Jožef Stefan Institute. The framework of the operation also includes the University of Ljubljana, the National Institute of Chemistry, the Institute for Electronics and Vacuum Technology and the Institute for Materials and Construction Research among others. In addition, the project is supported by the Ministry of Economic Relations and Development, the National Chamber of Economy and the City of Ljubljana.

Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
Tel.:+386 1 4773 900, Fax.:+386 1 219 385
Tlx.:31 296 JOSTIN SI
WWW: http://www.ijs.si
E-mail: matjaz.gams@ijs.si
Contact person for the Park: Iztok Lesjak, M.Sc.
Public relations: Natalija Polenec

CONTENTS OF *Informatica* Volume 26 (2002) pp. 1–439

Papers

Rajalingham, K., D. Chadwick & B. Knight. 2002. Efficient methods for checking integrity: A structured spreadsheet engineering methodology. Informatica 26:181–189.

Raković, D. & M. Dugić. 2002. A critical note on the role of the quantum mechanical "collapse" in quantum modeling of consciousness. Informatica 26:85–90.

Ramani, A., S. Vhora & S. Sanyal. 2002. The next generation Internet protocol. Informatica 26:27–45.

Sarkar, D. & P.K. Das. 2002. On mirroring, connected component labelling and topological properties of images enclosed as minimized boolean function. Informatica 26:17–25.

Savanović, A., D. Gabrijelčič, B. Jerman Blažič & S. Karnouskos. 2002. An active networks security architecture. Informatica 26:211–221.

Semeraro, G., M. Degemmis & P. Lops. 2002. User profiling to support Internet customers: What do you want to buy today? Informatica 26:407–418.

Spalka, A., A.B. Cremers & H. Langweg. 2002. Trojan horse attacks on software for electronic signatures. Informatica 26:191–203.

Tai, S.-C., C.-C. Wang & C.-S. Yu. 2002. Visual secret sharing watermarking for digital image. Informatica 26:381–388.

Taniar, D., Y. Jiang, K.H. Liu & C.H.C. Leung. 2002. Parallel aggregate-join query processing. Informatica 26:321–332.

Taweel, A. & P. Brereton. 2002. Developing software across time zones: An exploratory empirical study. Informatica 26:333–344.

Wang, C., Y. Wang & F. Zhang. 2002. An anonymous mobile agents scheme for secure web transaction over the Internet. Informatica 26:291–297.

Wei, B., D. Liu & X. Wang. 2002. Analysis of AES S-box with Walsh spectrum. Informatica 26:259–262.

Wu, H., F. Bao & R.H. Deng. 2002. Cryptanalysis of some hash functions based on block ciphers and codes. Informatica 26:255–258.

Železnikar, A.P. 2002. Informon—An emergent conscious component. Informatica 26:419–431.

Editorials

Bergadano, F. & C.-K. Wu. 2002. Cryptology and network Security. Informatica 26:245–248.

Welzer, T. & L. Strous. 2002. IFIP–TC 11. Informatica 26:103–104.

Professional Societies

Jožef Stefan Institute. Ljubljana, Slovenia. 2002. Informatica 26:101,243,345,439.

INFORMATICA

AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS

INVITATION, COOPERATION

Submissions and Refereeing

Please submit three copies of the manuscript with good copies of the figures and photographs to one of the editors from the Editorial Board or to the Contact Person. At least two referees outside the author's country will examine it, and they are invited to make as many remarks as possible directly on the manuscript, from typing errors to global philosophical disagreements. The chosen editor will send the author copies with remarks. If the paper is accepted, the editor will also send copies to the Contact Person. The Executive Board will inform the author that the paper has been accepted, in which case it will be published within one year of receipt of e-mails with the text in Informatica LaTeX format and figures in .eps format. The original figures can also be sent on separate sheets. Style and examples of papers can be obtained by e-mail from the Contact Person or from FTP or WWW (see the last page of Informatica).

Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the Contact Person.

QUESTIONNAIRE

□ Send Informatica free of charge

□ Yes, we subscribe

Please, complete the order form and send it to Dr. Rudi Murn, Informatica, Institut Jožef Stefan, Jamova 39, 1111 Ljubljana, Slovenia.

Since 1977, Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automation and other related areas. In its 16th year (more than five years ago) it became truly international, although it still remains connected to Central Europe. The basic aim of Informatica is to impose intellectual values (science, engineering) in a distributed organisation.

Informatica is a journal primarily covering the European computer science and informatics community - scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board.

Informatica is free of charge for major scientific, educational and governmental institutions. Others should subscribe (see the last page of Informatica).

ORDER FORM - INFORMATICA

Name: ....

Title and Profession (optional): ....

....

Home Address and Telephone (optional): ....

....

Office Address and Telephone (optional): ....

....

E-mail Address (optional): ....

Signature and Date: ....

Informatica WWW:

http://ai.ijs.si/informatica/
http://orca.st.usm.edu/informatica/

Referees:

Witold Abramowicz, David Abramson, Adel Adi, Kenneth Aizawa, Suad Alagić, Mohamad Alam, Dia Ali, Alan Aliu, Richard Amoroso, John Anderson, Hans-Jurgen Appelrath, Iván Araujo, Vladimir Bajič, Michel Barbeau, Grzegorz Bartoszewicz, Catriel Beeri, Daniel Beech, Fevzi Belli, Simon Beloglavec, Sondes Bennasri, Francesco Bergadano, Istvan Berkeley, Azer Bestavros, Andraž Bežek, Balaji Bharadwaj, Ralph Bisland, Jacek Blazewicz, Laszlo Boeszoermenyi, Damjan Bojadžijev, Jeff Bone, Ivan Bratko, Pavel Brazdil, Bostjan Brumen, Jerzy Brzezinski, Marian Bubak, Davide Bugali, Troy Bull, Leslie Burkholder, Frada Burstein, Wojciech Buszkowski, Rajkumar Bvyya, Netiva Caftori, Particia Carando, Robert Cattral, Jason Ceddia, Ryszard Choras, Wojciech Cellary, Wojciech Chybowski, Andrzej Ciepielewski, Vic Ciesielski, Mel Ó Cinnéide, David Cliff, Maria Cobb, Jean-Pierre Corriveau, Travis Craig, Noel Craske, Matthew Crocker, Tadeusz Czachorski, Milan Češka, Honghua Dai, Bart de Decker, Deborah Dent, Andrej Dobnikar, Sait Dogru, Peter Dolog, Georg Dorfner, Ludoslaw Drelichowski, Matija Drobnič, Maciej Drozdowski, Marek Druzdzel, Marjan Družovec, Jozo Dujmović, Pavol Ďuriš, Amnon Eden, Johann Eder, Hesham El-Rewini, Darrell Ferguson, Warren Fergusson, David Flater, Pierre Flener, Wojciech Fliegner, Vladimir A. Fomichov, Terrence Forgarty, Hans Fraaije, Hugo de Garis, Eugeniusz Gatnar, Grant Gayed, James Geller, Michael Georgiopolus, Michael Gertz, Jan Goliński, Janusz Gorski, Georg Gottlob, David Green, Herbert Groiss, Jozsef Gyorkos, Marten Haglind, Abdelwahab Hamou-Lhadj, Inman Harvey, Jaak Henno, Marjan Hericko, Elke Hochmueller, Jack Hodges, Doug Howe, Rod Howell, Tomáš Hruška, Don Huch, Simone Fischer-Huebner, Alexey Ippa, Hannu Jaakkola, Sushil Jajodia, Ryszard Jakubowski, Piotr Jedrzejowicz, A. Milton Jenkins, Eric Johnson, Polina Jordanova, Djani Juričič, Marko Juvancic, Sabhash Kak, Li-Shan Kang, Ivan Kapustøk, Orlando Karam, Roland Kaschek, Jacek Kierzenka, Jan Kniat, Stavros Kokkotos, Fabio Kon, Kevin Korb, Gilad Koren, Andrej Krajnc, Henryk Krawczyk, Ben Kroese, Zbyszko Krolikowski, Benjamin Kuipers, Matjaž Kukar, Aarre Laakso, Les Labuschagne, Ivan Lah, Phil Laplante, Bud Lawson, Herbert Leitold, Ulrike Leopold-Wildburger, Timothy C. Lethbridge, Joseph Y-T. Leung, Barry Levine, Xuefeng Li, Alexander Linkevich, Raymond Lister, Doug Locke, Peter Lockeman, Matija Lokar, Jason Lowder, Kim Teng Lua, Ann Macintosh, Bernardo Magnini, Andrzej Małachowski, Peter Marcer, Andrzej Marciniak, Witold Marciszewski, Vladimir Marik, Jacek Martinek, Tomasz Maruszewski, Florian Matthes, Daniel Memmi, Timothy Menzies, Dieter Merkl, Zbigniew Michalewicz, Gautam Mitra, Roland Mittermeir, Madhav Moganti, Reinhard Moller, Tadeusz Morzy, Daniel Mossé, John Mueller, Jari Multisilta, Hari Narayanan, Jerzy Nawrocki, Rance Necaise, Elzbieta Niedzielska, Marian Niedq'zwiedziński, Jaroslav Nieplocha, Oscar Nierstrasz, Roumen Nikolov, Mark Nissen, Jerzy Nogieć, Stefano Nolfi, Franc Novak, Antoni Nowakowski, Adam Nowicki, Tadeusz Nowicki, Daniel Olejar, Hubert Österle, Wojciech Olejniczak, Jerzy Olszewski, Cherry Owen, Mieczyslaw Owoc, Tadeusz Pankowski, Jens Penberg, William C. Perkins, Warren Persons, Mitja Peruš, Stephen Pike, Niki Pissinou, Aleksander Pivk, Ullin Place, Gabika Polčicová, Gustav Pomberger, James Pomykalski, Dimithu Prasanna, Gary Preckshot, Dejan Rakovič, Cveta Razdevšek Pučko, Ke Qiu, Michael Quinn, Gerald Quirchmayer, Vojislav D. Radonjic, Luc de Raedt, Ewaryst Rafajlowicz, Sita Ramakrishnan, Kai·Rannenberg, Wolf Rauch, Peter Rechenberg, Felix Redmill, James Edward Ries, David Robertson, Marko Robnik, Colette Rolland, Wilhelm Rossak, Ingrid Russel, A.S.M. Sajeev, Kimmo Salmenjoki, Pierangela Samarati, Bo Sanden, P. G. Sarang, Vivek Sarin, Iztok Savnik, Ichiro Satoh, Walter Schempp, Wolfgang Schreiner, Guenter Schmidt, Heinz Schmidt, Dennis Sewer, Zhongzhi Shi, Mária Smolárová, Carine Souveyet, William Spears, Hartmut Stadtler, Olivero Stock, Janusz Stokłosa, Przemysław Stpiczyński, Andrej Stritar, Maciej Stroinski, Leon Strous, Tomasz Szmuc, Zdzislaw Szyjewski, Jure Šilc, Metod Škarja, Jiři Šlechta, Chew Lim Tan, Zahir Tari, Jurij Tasič, Gheorge Tecuci, Piotr Teczynski, Stephanie Teufel, Ken Tindell, A Min Tjoa, Vladimir Tosic, Wieslaw Traczyk, Roman Trobec, Marek Tudruj, Andrej Ule, Amjad Umar, Andrzej Urbanski, Marko Uršič, Tadeusz Usowicz, Romana Vajde Horvat, Elisabeth Valentine, Kanonkluk Vanapipat, Alexander P. Vazhenin, Jan Verschuren, Zygmunt Vetulani, Olivier de Vel, Valentino Vranić, Jozef Vyskoc, Eugene Wallingford, Matthew Warren, John Weckert, Michael Weiss, Tatjana Welzer, Lee White, Gerhard Widmer, Stefan Wrobel, Stanislaw Wrycza, Janusz Zalewski, Damir Zazula, Yanchun Zhang, Ales Zivkovic, Zonling Zhou, Robert Zorc, Anton P. Železnikar

EDITORIAL BOARDS, PUBLISHING COUNCIL

Informatica is a journal primarily covering the European computer science and informatics community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor from the Editórial Board can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the list of referees. Each paper bears the name of the editor who appointed the referees. Each editor can propose new members for the Editorial Board or referees. Editors and referees inactive for a longer period can be automatically replaced. Changes in the Editorial Board are confirmed by the Executive Editors.

The coordination necessary is made through the Executive Editors who examine the reviews, sort the accepted articles and maintain appropriate international distribution. The Executive Board is appointed by the Society Informatika. Informatica is partially supported by the Slovenian Ministry of Science and Technology.

Each author is guaranteed to receive the reviews of his article. When accepted, publication in Informatica is guaranteed in less than one year after the Executive Editors receive the corrected version of the article.

Volume 26 Number 4 December 2002 ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics