Herwig Unger, Kyandoghere Kyamaky, and Janusz Kacprzyk (Eds.)

Autonomous Systems: Developments and Trends

Studies in Computational Intelligence, Volume 391

Editor-in-Chief

Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw
Poland
E-mail: kacprzyk@ibspan.waw.pl

Herwig Unger, Kyandoghere Kyamaky,
and Janusz Kacprzyk (Eds.)

Autonomous Systems: Developments and Trends

Springer

Editors

Prof. Dr.-Ing. habil. Herwig Unger
Fernuniversität Hagen
Fakultät für Mathematik und Informatik
Universitätsstr. 27
58084 Hagen
Germany
Phone: +49 2331 9871155
Fax: +49 2331 987353
E-mail: herwig.unger@FernUni-Hagen.de

Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw
Poland
E-mail: kacprzyk@ibspan.waw.pl

Univ.-Prof. Dr.-Ing. Kyandoghere Kyamakya
Alpen Adria Universität Klagenfurt
Institut für Intelligente Systemtechnologien
(Smart System Technologies)
Universitätsstraße 65-67
9020 Klagenfurt am Wörthersee
Austria
E-mail: kyandoghere.kyamakya@uni-klu.ac.at

Foreword

The Workshops on Autonomous Systems emanated from a gathering with the doctoral students of just three chairs at Fernuniversitt in Hagen, which we organise twice per year for a number of years now. Their purpose is to discuss on-going research and to create a community spirit. Furthermore, they serve as a means of structuring the students' research processes.

It is not suitable to conduct such a meeting on one's own premises, because there are too many disturbances and one cannot really escape from the daily routine. For its research-related events Advanced Study Institutes and Advanced Research Workshops NATO thus used to require in the corresponding guideline to carry them out in relative seclusion. Following this approach, in spring we meet in Schloss Dagstuhl, the Leibniz Centre for Informatics, which fulfills these requirements.

To change locations, we looked for another secluded place for the autumn meetings. Comparing offers of a large number of facilities, some of which just a few kilometers down the road, revealed that the most economic solution was to fly to the Mediterranean island of Mallorca with a low-cost airline, and to take advantage of special offers holding for the last week of the season before hotels close for the winter. After a first try, we found a hotel in an isolated bay, but still easily accessible by bus from Palma — in other words: we found relative seclusion. In this hotel, the workshop takes place now for the third time. The workshop has grown and matured in several respects. The doctoral students presenting their work do not come from a single university anymore, but from three. Besides them and their supervisors, also other scientists became interested in the event and contribute to its programme. Following the model of Advanced Study Institutes, they are available on the premises for relaxed, informal discussions outside the formal sessions. Finally, with the co-sponsorship of Gesellschaft fr Informatik, the German Computer Society, and this surprisingly comprehensive volume of contributions published by Springer-Verlag the workshop turned into a visible scientific event.

Besides its educational and scientific purpose, the workshop in 2011 is dedicated to the celebration of my 60th birthday. Moving into the grandfather age, I am particularly happy that many participants, several of whom are former doctoral students of mine, come with their families including children and even two newly born babies.

So, I cordially thank all persons having organised and contributed to this workshop, I appreciate the willingness of Gesellschaft fr Informatik to act as co-sponsor and of Springer-Verlag to publish this nice book.

I do wish all participants a good time, that they have some fun and interesting conversations.

Hagen, June 2011   Wolfgang Halang

Programme Committee

Laudatio in Honour of
Prof. Dr. Dr. Wolfgang A. Halang

by Prof. Dr. Matjaž Colnarič

It was with my greatest pleasure that I accepted the invitation from the organisers of this conference to prepare a few words in honour of Prof. Dr. Dr. Wolfgang A. Halang's 60th birthday. I have known him for twenty three years and I was his first PhD student. Since then, we have done a lot of work and spent many beautiful hours together.

In the mid 80's of the previous century, long after finishing my master's degree, when I was doing a lot of industrial research and was already lecturing for some years at the university, I realised that I should have started with my doctoral studies long ago. I read some papers and got a feeling that my experience from our previous work was pretty much in line with what I read about real-time. I tried to define a thesis but, being stubborn and picky about the domain, I did not find much support in my environment. So, I gathered the courage and submitted a paper on one of our applied projects to SERTA, an international conference in Cirencester. Surprising for me at that time, it was accepted and I gathered funds to attend it. I was looking for somebody to help me with my thesis there and I ran into a person who was immediately ready to talk to me and, shockingly quickly, agreed to supervise my work. That is how I met Prof. Halang and became his first doctoral student.

Prof. Dr. Dr. Wolfgang A Halang received a doctorate in mathematics from Ruhr Universityät Bochum in 1976, and a second doctorate in computer science from Universityät Dortmund in 1980. During and immediately after his studies, he was a part time employee at Coca Cola GmbH in Essen for nine years. After his second dissertation, he has spent a couple of years travelling the world, lecturing and researching for two years at the King Fahd University of Petroleum and Minerals in Dhahran, Saudi Arabia (1986-87) and another year at the University of Illinois at Urbana Champaign in US (1987-88). When he returned home, he led the Department of Control Engineering at Bayer AG in Leverkusen and Dormagen before he was appointed to the Chair for Application Oriented Computing Science and head of the Department of Computing Science at the University of Groningen in the Netherlands in June 1989. Since October 1991, he has held the Chair of Information Technology at the Faculty of Electrical Engineering at Fern Universityät Hagen in Germany, where he also served as the Dean in the years 2002-2006.

His research interests comprise all major areas of real time systems and domains related to them. He wrote a large number of pier-reviewed journal and conference papers, and authored or co-authored a number of books and book chapters. He has organised important and successful scientific conferences, of which I would particularly like to mention the Workshop on Real-Time Programming, which is probably the oldest series of scientific meetings in the domain of computer control. I am happy to report that this series will be revived as a track of the newly organised triennial conference CESCIT (Conference on Embedded Systems, Computer Intelligence and Telecommunications) in April 2012 in Würzburg, where he will serve as the Honorary Chair.

On the national level, Professor Halang is chairing the Expert Committee on Real-Time Systems (Fachausschuss Echzeitsysteme) of German Informatics Society (Gesellschaft für Informatik) and is one of the driving forces of the PEARL-Meeting and Real-time Systems Conference (Pearl Taggung and Echtzeitsysteme).

He has served as an official in international professional organisations. For two terms, he was chairman of the Coordination Committee on Computers, Cognition and Communication of IFAC, the most important world association of control professionals. For a number of years, he was European editor of the important specialised International Journal of Real-Time Systems, published first by Kluwer, then by Elsevier.

He is a mentor to young researchers in the broader domain of real-time and safety related systems; he supervised a number of doctoral students. He is also renowned for choosing his candidates not only from German universities, but also from other countries, of which China, Poland and also Slovenia prevail. With institutions in these countries he actively maintains successful professional collaboration.

The list of Dr. Halang's scientific and professional engagements does not end here. I apologise for my lack of knowledge of all his involvement, especially in the German national professional community.

Everybody who knows Prof. Halang will probably agree with me that he is not just another professor or scientist, but an interesting and unique person with some very specific personal characteristics. In the following, I would like to point out some of them, which I find that represent him the most. I am sure Professor Halang will not mind if I respectfully share some fragments of my recollections of times and events spent with him, just to illustrate some of these characteristics.

The scientific work of Professor Halang is characterised by *innovation*. For example: In his early works, he introduced an innovative asymmetrical multiprocessor architecture for dedicated functions and its implementation. During our collaboration on my doctoral thesis, I often came to a point where I could not find any reasonable solution. His very frequent advice in such cases was: "Querdenken! Do not think the way everybody else does, try to consider things from an unusual aspect."

On the other hand, through his rich experience and broad knowledge, he often shows that a lot of problems, which eventually emerge, have already been dealt with before, possibly in another form. Often the solutions can still be reused, at least as a basis for implementation in new technologies. Especially in informatics, things often reappear in circles, like multiprocessing, distributed processing, clusters, clouds, etc.

Furthermore, he is "allergic" to papers dealing with scheduling. As he often states, the scheduling problems have essentially been solved in the early 1970s. Many of the authors of later papers only use this domain in order to exhibit their perfection in creating problems and proving correct their useless solutions as well as their inability to take delight in other sophisticated academic theoretical domains. Let me add my personal experience here: I once had to review a scheduling paper. I struggled through the formal descriptions of algorithms, theorems, proofs and lemmas until the authors concluded that their solution almost meets the performance of random scheduling.

One of his significant personal characteristics is his *frankness*. I cannot imagine anyone saying Prof. Halang was not sincere to him or her. That was not always good for him and did not help him to make a lot of convenient friends. For example: In general, he is not very fond of our friends over the Atlantic. This is mainly because of their attitude towards the rest of the world, at least in our domain. He does not hide his opinion and, in return, they have often ignored him. Although he is very precise in correct use of English, which became lingua franca of science and technology, we heard him joking that English is an underdeveloped ancient dialect of German.

Once he decides on a certain reasonable and good solution, it is usually not easy to persuade him to use any other, less perfect one, just because it is easier. It was the same when our collaboration on my doctoral thesis began.

At that time, in the late 1980s, he has just returned to Europe and was stationed in Groningen, in the Netherlands. Since he was reluctant to use the traditional phone and mail communication, our first technical problem was to establish an electronic communication channel; electronic mail then was not very common nor usual or reliable, especially in our part of Europe. The problem was that we were using VAX VMS operating system at our university, which was totally incompatible with Unix on Sun workstations used in Groningen, which also pertains to communication protocols. The addresses on his side were formed using a proprietary-standard UUCP (Unix-to-Unix communication protocol) by "bangs" or exclamation marks (in the form of !site!machine!user). After some shaky and more or less unsuccessful attempts, two gateways have been written for us; one residing in Groningen and another one in Belgrade, converting addressing standards between European Academic Network EAN (X400) used by us, and the Unix protocol. The solution was definitely not easy, but after some initial hiccups, it worked excellent and saved us a lot of trouble in communication. I mailed him weekly reports on my research and he commented on them and proofread the final versions. Eventually we had lot of materials ready for editing my thesis and joint publications afterwards.

If I would have to decide what is, in my opinion, his ultimate maxim, I would not hesitate. *Simplicity.* During our intensive collaboration, he was never satisfied with

any of my sophisticated solutions, regardless of how perfectly they have complied with the requirements. Of course there was always a good reason for that simple solutions are easy to understand, prove and believe.

He always likes to use nice quotations from wise men, especially in the context of praising simplicity. Allow me to repeat some of them, which always still come to mind whenever I run into a nice and scientifically looking, overly complex, sophisticated and publishable solution to a simple problem:

1. Simple solutions are the most difficult ones: they require high innovation and complete intellectual penetration of issues.
2. Progress is the road from the primitive via the complicated to the simple. (both by Dr. Kurt Biedenkopf, professor and former Prime Minister of the State of Saxony, 1991 and 1994, respectively)
3. I also remember him making jokes especially on the account of informatics, citing Edsger Dijkstra's statement from 1989 that "It is time to unmask the computing community as a Secret Society for the Creation and Preservation of Artificial Complexity" or, as Professor Halang better puts it, Complicatedness.

Related to the simplicity, he always advocates *minimalism*. For example, he always travels lightly, never with luggage that would have to be checked in. Many of us know his famous little old brown suitcase, which is always large enough for his luggage, regardless of the season and the duration of the travel. And his plastic bag with fluid toilet utensils with the average volume of a couple of millilitres, with which he likes to impress airport security people so much.

Once at Graz airport in Austria, my usual departure site, going to one of our joint trips to Hong Kong, I realised that I had forgotten my reading glasses. They offered me a pair that was about one centimetre wide and was kept in a cartridge of the shape and size of a larger body-thermometer box in the duty-free shop. When he saw it, he said, with admiration and even slight envy: "This really is minimalistic".

Another one of his very personal attitudes is his reluctance to spend any resources on unnecessary expenses, like taxes, postal or bank fees, expensive flight tickets or excessive conference fees. Let me be very clear that he is far from being avaricious. He just does not want to get ripped off, if not absolutely necessary.

In 1996, I received a DAAD grant as a young academician for a visit to Hagen, where we worked on our joint publications. In order to minimize the expenses, Prof. Halang and his family very generously hosted me in their home for the whole month and made me feel as a real member of their family along with his devoted wife Berta and their children Andrea and Mathias. One morning he had to leave for a short trip by train and asked me to drive him to the train station with his car. Unfortunately, I got caught by stationary radar violating a speed limit of 20 km/h for 5 km/h due to a road reconstruction. After a few months, he told me what happened: he received a photo and a fine ticket for 20 DEM. He sent the letter back saying that it can be clearly seen from the photo that he was not the one driving the car. The driver was a guest academician from abroad whose identity he is not willing to reveal. He does not think he is responsible for the fine, but will pay it if they insist. The letter never came back.

He studied and, I am sure, still follows the tax legislative thoroughly and takes tax declaration very seriously. He once said his goal was to make it like his neighbour dentist, who never pays taxes and even gets some money returned. Once his tax declaration was not accepted by a clerk at the tax office. He complained, referring to certain articles in the law, but without success. Of course he did not give up; he wrote to the finance minister, saying that his employees were incompetent of reading their own legislation. His complaint was granted.

I am sure many people could add a lot more of his personal characteristics, events and anecdotes but this is how he is permanently residing in my personal pleasant recollections. Dear Professor Halang, we wish you good health and happiness among your dearest. All of us gathered here to celebrate your sixtieth anniversary are looking forward to many coming years of friendship, to spending a lot of nice hours together, meeting in beautiful places and, if really necessary, successful collaboration in our common domains!

Contents

Part III: Modeling

Part IV: Self Organisation

Part V: Nonlinear Systems

Part VI: Decentralized Networks

Part VII: Decentralized Systems

Struggle for Temporal Predictability of Processors for Real-Time Environments, Revisited 20 Years after

Matjaž Colnarič

Abstract. In this contribution, a short overview of our early collaboration with Prof. Halang is given. Our joint research is briefly described and some results presented. First, prerequisite considerations are given, then the asymmetrical architecture exhibiting a dedicated processor for operating system kernel functions is outlined. Its design is further elaborated, divided in several parts. Then, an early implementation with microcontrollers and a dedicated processor for parallel architectures is shown, together with the programming constructs supporting tasking and temporal operations. It is interesting that the main ideas are still actual and are open to further elaboration with the modern technology, especially because the fully predictable processors for embedded applications are still not available.

1 Introduction

In the late 80's I first met Professor Halang on conferences dealing with real-time, first in SERTA Cirencester in 1989, then at ICECCS in Como, Italy. After some discussions, and without the need for much persuasion, he agreed to help me select the topic, to form the thesis and to supervise my work towards my doctorate which I have defended in my home University of Maribor. So, I became, as far as I know, his first PhD student.

At that time he has just returned to Europe and was stationed in Groningen in The Netherlands. After solving some electronic communication problems, our virtual collaboration started. During my studies we met couple of times in Groningen (later in Hagen), in Maribor, on the island of Krk in Croatia and also regularly at various conferences. He provided me with reading material, at that time still mainly in paper; I remember that on my first visit, to carry home the photocopied materials, I had to

Matjaž Colnarič
University of Maribor, Faculty of El. Eng. and Comp. Sci.,
Smetanova 17, 2000 Maribor Slovenia
e-mail: `colnaric@uni-mb.si`

buy an extra suitcase from the money I got from University of Groningen for giving a lecture to his students.

After a couple of years of intensive collaboration, in 1992 he persuaded me that I should finally write and submit my thesis, and that we can continue with further work afterwards; my thesis has been defended on the holiday of Whitsun in June 1992. After that, a fruitful period of almost 40 joint publications started, from which some of the earliest and most significant results, published in various journals, are collected in this paper.

2   Prerequisites

At the time when our collaboration started, Professor Halang has already in detail elaborated his idea of an asymmetrical multiprocessor architecture, which has been first published in a comprehensive internal report at Urbana Champaign [9]. The idea followed the pattern which could be found in the nature (e.g., operation of the human brain) or in organisation of work of a manager and his/her secretary: the latter takes over the overhead work, intercepts most interruptions by phone or visitors, prepares schedules of important events that need to be attended, and reminds him of them. This way, the manager is to the highest possible extent isolated from interruptions, allowing him to work in peace in order to keep his deadlines.

On the other side, in our laboratory we have at that time done a number of practical industrial applications in control, through which we have learnt that temporal determinism and predictability indeed is an important feature of control computer systems, which has not been elaborated enough at that time, in spite of the fact that certain other publications, e.g. [10], [11]] etc. have also helped to pave the way for real-time systems research to become an independent and broadly recognised discipline.

At that time, solid-state technology was advancing in giant steps. Unprecedented density of transistors on a single chip was enabled. On the other hand, in the beginning, similar to the case of the better known "software gap", the design methods did not follow the development and the potential processing power could not be fully utilised by sophisticated processor features. This was a time of wild ideas of co-locating memory and processors, implementations of co-processors, intelligent memories, etc. Thus, technological pre-requisites were given for new approaches and concepts in designing processor and computer architectures for real-time environments.

In the conventional architectures employed for computer control applications, a number of undesirable properties could be identified [3], which made a system's temporal behaviour unpredictable or/and difficult to be estimated. A common counterproductive characteristic of their design was excessive complexity, which made their verification difficult or impossible, e.g., processor architectures with their pipelines, caches, etc. enabling parallel operation of internal structures, direct memory access, delaying the instructions execution, data transfer protocol with random collision detection, etc.

The most severe obstacle in determinism and thus predictability of the process execution, however, were interrupts and exceptions. By definition, they introduce delays in execution of the processes, what makes the worst case execution time (WCET) estimation impossible. This became the ultimate topic in our research. Other features were explored mainly to support the design of applications and to solve additional problems jeopardising the predictability (e.g., language issues, exception handling, etc.).

3 The Proposed Architecture

In the traditional computer architecture the operating system is running on the same processor(s) as the application software. In response to any occurring event, the context is switched, system services are performed, and scheduling is done. Although it is very likely that the same process will be resumed, a lot of performance is wasted and temporal determinism and predictability disabled, by superfluous overhead. This suggests employing a parallel processor, to carry out the operating system services.

The computer system to cope with that is hierarchically organised. In Fig. 1 it is shown that it consists of the task processor and the kernel processor, which are fully separated from each other.
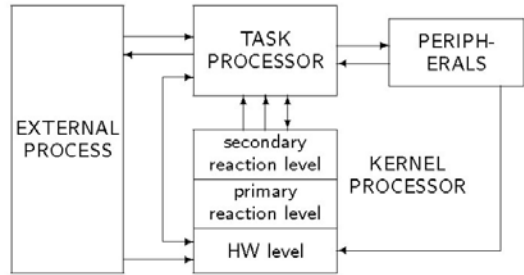
Fig. 1 Conceptual diagram of the architecture

Instead of interrupting the execution of process code in the task processor(s), the interrupt signals were introduced to the lowest level of the kernel (co-)processor where they were decoded, associated processes identified and scheduling done. Apart of that, also temporal events (processes scheduled on certain time, delays, and periodic events) and explicit task activations were served. Upon that, scheduling was done and eventually the task processing pre-empted, if necessary.

Upon this high-level architecture an implementation was designed, mainly focusing at the task processor, which was the main topic of my doctoral thesis. Unfortunately, since practical implementation was at that time not feasible, it was only verbally validated. The somewhat more detailed outlines are shown in Fig. 2. For the fully detailed implementation, we here refer to [2].

The task processor consisted of the task control and the data processing units. The task of the former was to fetch the instructions from the program memory and
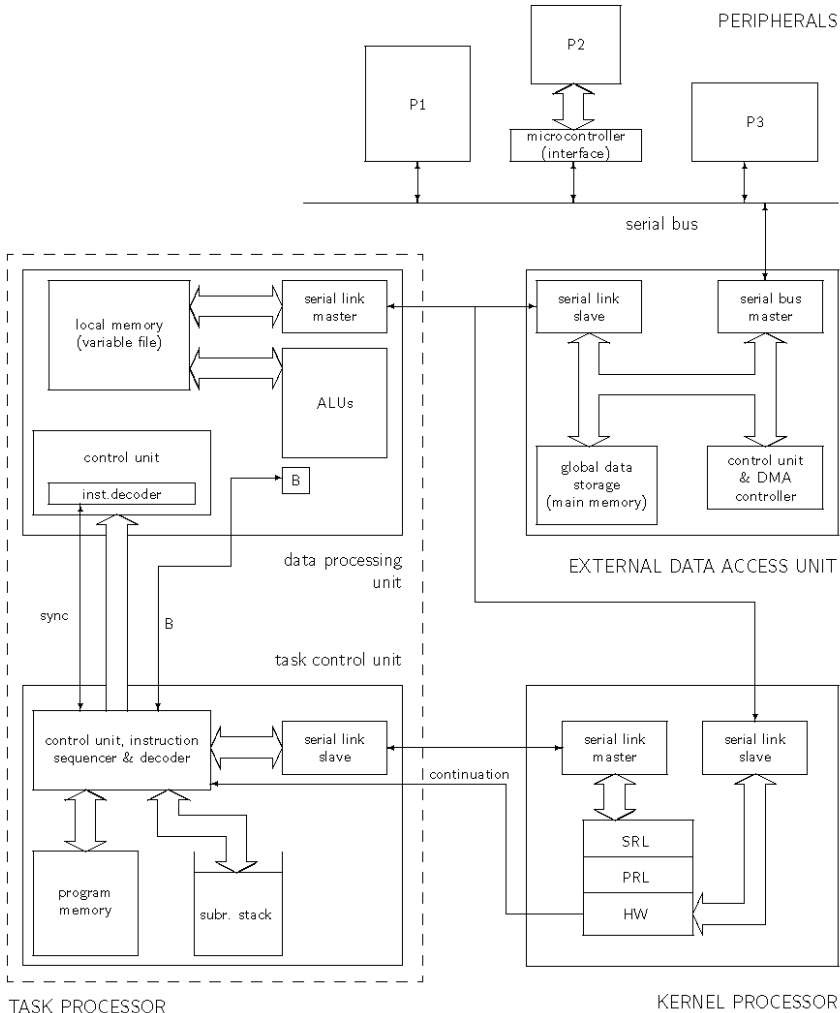
Fig. 2 More details about the design of the architecture

perform their pre-decoding to distinguish between the flow control and the data processing instructions. The flow control instructions (branches, subroutine calls and returns, and the wait instruction), which have an impact on the program counter only, are executed. To allow for conditional branches, a status flag (B) is accessible from the data processing unit. For the administration of subroutine return addresses a list of stacks is provided. The data processing instructions are forwarded via the parallel link to the data processing unit where they are executed in a sequential way.

The local variables are kept in a specially designed dual port local memory called "variable file". Part of it is directly addressable, holding constants (0, 1, and other

specific ones) and local variables, whose number is usually limited. Over the variable file also external data (variables residing in external memories and peripheral data) can be accessed by indirect addressing (locations holding their indirect addresses). To allow for that, various data locations were mapped to the data processing unit by the external data access unit.

All unites described here were designed to the level of digital ports, or even transistors in the case of the dual port local memory. The details were given in the doctoral thesis [2].

4 Implementation in Late 90's

At that time, the only possibility to rigorously implement these ideas would be the physical design in silicon by producing its mask and manufacturing the chips what was obviously not possible. Instead, in the following years, in the newly formed Laboratory for Real-Time Systems in Maribor, an implementation of the overall architecture using carefully selected processors was done, which is described in detail in [6].
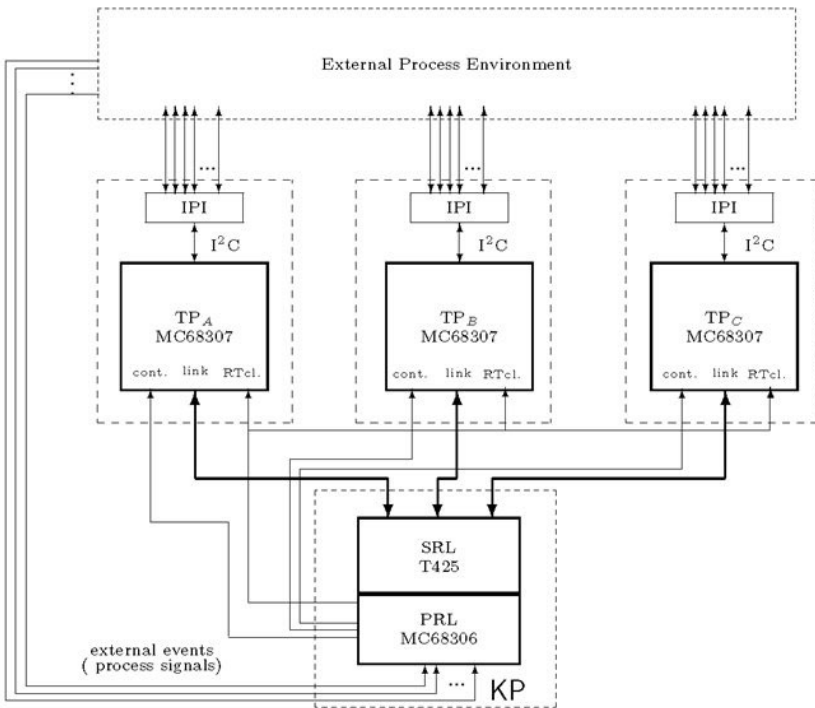
Fig. 3 Implementation of the kernel co-processor architecture

The outline of the design is given in the Fig. 3. The basic idea has been extended to three task processors, implemented using Motorola's microcontrollers MC68307. Kernel processor was built from two processors, Inmos Transputer T425 and Motorola microcontroller MC68306. Communication of the task processors and the kernel processor was done in point-to-point manner using the transputer's high speed serial links.

For this architecture, operating system has been designed, based on the task state transition diagram shown in Fig. 4, which is also in more detail given in [6].

The system communicated with the environment through intelligent peripheral interfaces (IPI) which were attached to the task processors by a simple I^2C serial bus. In the latter, task processes were executed and were only interrupted on request from the kernel processor. Events from the environment were fed to the lower level of the kernel processor (MC68306) which also kept track of time schedules and other task activation commands. Apart of that, real-time clock is kept there.

In [5] the operating system supporting the architecture has been described. The rather standard task state transition diagram is given in Fig. 4.
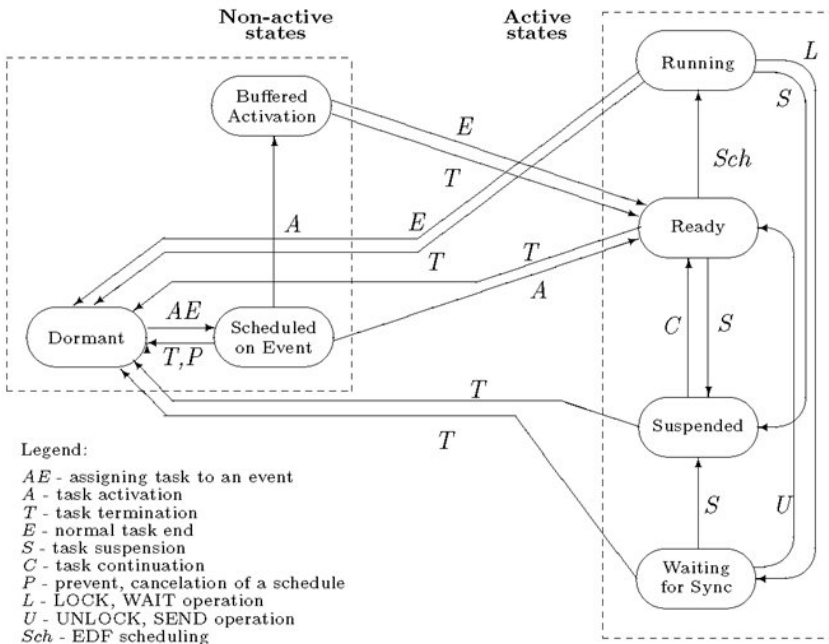
Fig. 4 State transition diagram of the operating system

Scheduling support

<non_time_schedule> ::= WHEN <non_time_event> { OR <non_time_event> }
<non_time_event> ::= <interrupt_id> | <shared_variable> [<rel_op> <exp>]
<simple_time_schedule> ::= AT <time_exp> | AFTER <dur_exp>
<periodical_time_schedule>::=[<simple_time_schedule>]EVERY<dur_exp>[DURING<dur_exp>]
<time_schedule> ::= <simple_time_schedule> | <periodical_time_schedule>
<combined_schedule> ::= <non_time_schedule> OR <time_schedule>
<schedule> ::= <non_time_schedule> | <time_schedule> | <combined_schedule>

Task activation

<task_activation> ::= [<schedule>] <task_activation_statement>
<task_activation_statement> ::= ACTIVATE <task_id> DEADLINE IN <dur_exp>

Task termination

<task_termination> ::= [<schedule>] <task_termination_statement>
<task_termination_statement> ::= TERMINATE [<task_id>]

Task prevention

<task_prevention> ::= [<schedule>] <task_prevention_statement>
<task_prevention_statement> ::= PREVENT [<task_id>]

Task suspension

<task_suspension> ::= [<schedule>] <task_suspension_statement>
<task_suspension_statement> ::= SUSPEND [<task_id>]

Task continuation and resumption

<task_continuation> ::= [<schedule>] <task_continuation_statement>
<task_continuation_statement> ::= CONTINUE <task_id> DEADLINE IN <dur_exp>
<task_resume> ::= [EXACTLY] <schedule> <task_resume_statement>
<task_resume_statement> ::= RESUME [<task_id>] [DEADLINE IN <dur_exp>]
    TIMEOUT <dur_exp> [ONTIMEOUT <statement>]

Normal task end

<task_end> ::= <task_end_statement>
<task_end_statement> ::= END

Synchronisation constructs and critical region concept

<send_statement> ::= SEND <signal_id>
<wait_statement> ::= WAIT <signal_id> TIMEOUT <dur_exp> [ONTIMEOUT <statement>]
ENTER <semaphore_id> [NP] TIMEOUT <dur_exp> <statement>{<statement>}
    [ONTIMEOUT <statement>{<statement>}] LEAVE

Explicit assertion of execution time

DURING <dur_exp> [NP] DO <statement>{<statement>}
    [ONTIMEOUT <statement>{<statement>}] FIN

Status or value acquisition constructs

<task_state_statement> ::= TSTATE <task_id>
<sync_state_statement> ::= SSTATE <sema_id>
<get_RTC_statement> ::= NOW

Interrupt enabling and disabling constructs

<interrupt_enable_statement> ::= ENABLE <interrupt_id>
<interrupt_disable_statement> ::= DISABLE <interrupt_id>

Fig. 5 Summary of scheduling and other OS related commands

There are two less standard features worth mentioning:

- Buffered activation is two-state task activation: a task is first scheduled for activation upon a temporal condition, and, after this condition is fulfilled, it must wait for a certain signal from environment.
- Waiting for Sync: a task can be suspended to wait for a temporal synchronisation.

Based on the asymmetrical distributed architecture and the pertaining programming features, timing analysis of the task programs has been elaborated in [14]. In this paper, an idea of automated syntax tree analysis and execution time measurement vie background debugging mode of Motorola's microcontroller ColdFire has been implemented, coping with pessimism of WCET estimation.

It is here to be mentioned that this architecture had some draw-backs. Most notably there was a mismatch of the distributed architecture connecting task processors and peripherals, while the interrupts were directly wired to the kernel processor which prevented local distribution of the units. This was coped with in the next versions of the architecture which will not be given in this paper but can be found in [1].

Last but not least, programming of the real-time applications has also been elaborated. In Fig. 5 a list of scheduling commands to be included in the programming language in order to support the architectural particularities is given. This was a part of the doctoral thesis [12] of Dr. Domen Verber, co-supervised by Prof. Halang and myself.

Apart of that, in this work an attempt of inclusion of terms of object oriented approach in a real-time programming language has been given. Although the dynamic object oriented approach appeared problematic, the tasks, peripherals, etc. has been presented in a similar way as objects, although they were ordinary static features. The syntax, however, could be adopted, providing a convenient and modern way of programming.

An interesting attempt of using a subset of a real-time programming language PEARL for specifying hardware and software features of a control system has been presented in [7]. This work was further conducted by another joint doctoral student, Dr. Roman Gumzej in his thesis [8].

Apart of the mentioned papers and topics, a number of other related ideas have also been dealt with in joint research and publications, like dealing with exceptions [4], attempts of establishing benchmarks for real-time control systems [15] and others.

Finally, our collaboration and also some later improvements have been rounded up our in our joint monograph [1] published by Springer Verlag in the series Advances in industrial control. The authors received the full recognition when we noted that it is available on pirate repositories.

5   Later Development in the Field of Predictability of Processors for Real-Time Applications

It is interesting that microprocessors with fully predictable behaviour still do not exist. However, the awareness of the fact that at least pessimistic estimation of worst

case execution time should be possible led to implementation of some important features of modern processors which are meant for embedded applications. For example, digital signal processors exhibit good temporal predictability. Their features that jeopardise the predictability can often be to a large extent, although not fully, disabled, e.g., cache memories can sometimes be shut off. Also, the pipeline operation has been elaborated in the sense that its breaking has minimum impact on the temporal delays of the tasks. This is done by sophisticated strategies of branch predictions and handling, and by backing-up the contents of the higher stages of the pipelines.

Similar considerations can be noted also in the domain of general purpose processors. Some families in parallel provide real-time versions, like, e.g., ARM Cortex^{TM}-R real-time processors for deeply embedded systems with demanding real-time response constraints. The latter, however, are considered more in the "soft" and less in the "hard" sense. The impact of architectural features is in the case of GP processors much more notable than in the field of digital signal processors. This is because of more intensive employment of architectural means for boosting their average performance at the cost of worst case behaviour.

6 Further Work

It is interesting to mention, that the general and overall ideas elaborated in our common works from early 90-ies appear to be still usable after twenty years. In the Laboratory for real-time systems we have lately done some further work in implementation of the kernel processor. We started with the FPGA implementation of the EDF scheduling algorithm together with run-time schedulability analysis. We are proud to mention that by parallel implementation of the algorithm we managed to reach $O(1)$ complexity of scheduling of a single newly arrived task into the existing list [13].

Further, in process is FPGA implementation of other essential features that are needed in a simple operating system for embedded applications. The idea is to implement the full OS kernel processor in FPGA in order to support temporal predictability and high speed of its operation. First versions are operational, the next step is its practical employment in a complete simple embedded system.

If this is successful, in preparation is an attempt of FPGA implementation of the parts of the task processor, based on the one from [2], of course adapted to the novel technologies. It is our goal to implement several units on the same device together with the kernel processor.

References

1. Colnaric, M., Verber, D., Halang, W.A.: Distributed embedded control systems: improving dependability with coherent design. In: Advances in Industrial Control. Spinger, London (2008)
2. Colnarič, M.: Predictability of temporal behaviour of hard real-time systems. Ph.D. thesis, University of Maribor (1992)

3. Colnarič, M., Halang, W.A.: Architectural support for predictability in hard real-time systems. Control Engineering Practice 1(1), 51–59 (1993); ISSN 0967–0661

4. Colnarič, M., Halang, W.A.: Exception handling and predictability in hard real-time systems. In: Proceedings of the 12th International Conference on Computer Safety, Reliability and Security SAFECOMP 1993, Poznań – Kiekrz, Poland. Springer, Heidelberg (1993); ISBN: 0-387-19838-5

5. Colnarič, M., Halang, W.A., Tol, R.M.: Hardware supported hard real-time operating system kernel. Microprocessors and Microsystems 18(10), 141–9331 (1994); ISSN 0141-9331

6. Colnarič, M., Verber, D., Gumzej, R., Halang, W.A.: Implementation of embedded hard real-time systems. International Journal of Real-Time Systems 14(3), 293–310 (1998)

7. Colnarič, M., Verber, D., Halang, W.A.: A Real-Time Programming Language as a Means to Express Specifications. Control Engineering Practice 5(7), 975–983 (1997); ISSN 0967-0661

8. Gumzej, R.: Vzporedno načrtovanje arhitekture vgrajenih sistemov in njena validacija (parallel design of architecture of embedded systems and its validation, in slovene). Ph.D. thesis, University of Maribor (1999)

9. Halang, W.A.: Definition of an auxiliary processor dedicated to real– time operating system kernels. Tech. Rep. UILU-ENG-88-2228 CSG-87, University of Illinois at Urbana Champaign (1988)

10. Liu, C., Layland, J.: Scheduling algorithms for multiprogramming in a hard real-time environment. Journal of the ACM 20(1), 46–61 (1973)

11. Stankovic, J.A.: Misconceptions about real-time computing. IEEE Computer 21(10), 10–19 (1988)

12. Verber, D.: Object orientation in hard real-time system development. Ph.D. thesis, University of Maribor, Slovenia (1999)

13. Verber, D.: Co-processor for microkernel os services. In: Proceedings of the IFAC World Congress, IFAC, Milan, Italy (2011)

14. Verber, D., Colnarič, M., Halang, W.A.: Programming and time analysis of hard real-time applications. Control Engineering Practice 4(10), 1427–1434 (1996); ISSN 0967-0661

15. Halang, W.A., Gumzej, R., Colnaric, M., Druzovec, M.: Measuring the performance of real-time systems. Real-Time Systems Journal 18(1), 59–68 (2000)

A New Approach for Choosing the Most Appropriate Fuzzy Ranking Algorithm for Solving MADM Problems

Fahimeh Ramezani and Jie Lu

Abstract. There are many fuzzy ranking algorithms available to solve multi-attribute decision making (MADM) problems. Some are more suitable than others for particular decision problems. This paper proposes a new method for choosing the most appropriate fuzzy ranking algorithm for solving MADM problems based on the type and number of attributes and the number of alternatives, considering the least time consumption and the least computation for ranking alternatives. In addition, we develop a software to simulate three main fuzzy ranking algorithms: SAW, Negi, and Chen and Hwang (Chen and Hwang 1992). This software can be used in any MADM decision support system.

1 Introduction

Many multi-attribute decision-making (MADM) models which apply various ranking algorithms for solving different types of decision problems have been developed (Hwang and Yoon 1981; Zeleny 1982; Colson and de Bruyn 1989; Dyer et al. 1992; Stewart 1992) and they are widely used in ranking the finite number of alternatives with respect to multiple attributes. In some MADM problems with a short list of alternatives, the ranking outcome produced by some MADM methods may not differ (Belton 1986; Karni et al. 1990). However, in some MADM problems which need all the alternatives, or a subset of them, different methods often produce inconsistent rankings for the same problem (Voogd 1983; Zanakis et al. 1998). If the number of alternatives increases or if the alternatives perform in a similar, the inconsistency of ranking outcome of MADM methods will increase (Olson et al. 1995).

Fahimeh Ramezani · Jie Lu
School of Software, Faculty of Engineering and Information Technology,
University of Technology, Sydney, P.O. Box 123, Broadway, NSW 2007, Australia
e-mail: {Fahimeh.Ramezani,Jie.Lu}@uts.edu.au

Because there are a large variety of decision problems solvable by various MADM methods, the selection of a MADM method for a given problem has been regarded as a MADM problem itself. Along this line of research, method selection procedures have been proposed based on the comparison between the characteristics of the decision problem and the distinct features of available methods (Yeh 2002). Zanakis et al. (1998) compared similarities and differences in the behavior of eight ranking algorithms for solving the MADM problem by their performance measurement. Their simulation experiment evaluated these algorithms under different number of alternatives, criteria and distributions. Their performance measures show that final results are affected by these three factors in that order, and if the number of alternatives increases, the algorithms tend to produce similar final weights but dissimilar rankings, and more rank reversals. However, like other studies, the appropriateness or validation of the algorithms considered is not dealt with in their study. Yeh (2002) proposed a new validation approach to validate the ranking outcome of three ranking algorithms (SAW, MEW and TOPSIS) by examining their ability to reflect the decision information embedded in the problem data set, via sensitivity analysis of attribute weights. To choose the best ranking algorithms, according to this approach, decision makers (DM) should apply all these algorithms to rank their alternatives and also do comprehensive computations to validate them, finally choosing the result of one of them. Considering most MADM problems involve both quantitative and qualitative assessment attributes and require fuzzy ranking algorithms, and most fuzzy ranking algorithms individually have the problem of cumbersome computations (Chen and Hwang 1992), even using one of these algorithms for solving large size fuzzy MADM problems is not reasonable and applying three of them in such problems seems to be incorrect. Therefore, DMs need to choose the most appropriate ranking algorithms for their MADM problems before starting computations.

All this considered, determining the most suitable ranking algorithm for a given MADM problem which involves the least time consumption and the least computation remains an open issue. To this end, this paper proposes a new method for choosing the most appropriate fuzzy ranking algorithm for solving a given MADM problem, based on the type and number of attributes and the number of alternatives, considering the least computation and time consumption for ranking alternatives. Three fuzzy ranking algorithms: (1) SAW, (2) Negi's fuzzy ranking algorithm, and (3) Chen and Hwang fuzzy ranking algorithm, are considered in this paper because these algorithms can be applied for solving both MADM and FMADM problems with both crisp and fuzzy numbers. Moreover, despite some algorithms like AHP, these three algorithms give DMs an opportunity to assess attributes individually.

In the following sections, we first explain three considered fuzzy ranking algorithms in Section 2. In Section 3, we present a new method for choosing the most suitable fuzzy ranking algorithm for solving given MADM problems. Finally, in Section 4 we describe a software which has been developed to simulate these three fuzzy ranking algorithms and compare their time consumptions.

2   Feature Analysis of Three Main Fuzzy Ranking Algorithms

In this section, three fuzzy ranking algorithms: "SAW", "Negi's algorithm" and "Chen and Hwang algorithm" for solving the MADM problem are explained briefly.

2.1   Simple Additive Weighting Algorithm (SAW)

The SAW algorithm is known as the weighted sum algorithm and is probably the best known and most widely used MADM method (Hwang and Yoon, 1981). The basic logic of SAW is to obtain a weighted sum of the performance ratings of each alternative over all attributes (Fishburn 1967; MacCrimmon 1968). With a normalized decision matrix (r_{ij}) and a weight vector (w_j), the overall preference value of each alternative (U_i) is obtained by:

$$U_i = \sum_{j=1}^{n} w_j \cdot r_{ij} \qquad (1)$$

The greater the value (U_i) the more preferred the alternative (A_i). Research results have shown that the linear form of trade-offs between attributes used by the SAW algorithm produces extremely close approximations to complicated nonlinear forms, while being far easier to use and understand (Hwang and Yoon 1981).

This algorithm can be appropriate for solving problems with triangle or trapezoidal fuzzy numbers, otherwise the other algorithm are more appropriate (Chen and Hwang 1992).

2.2   Chen and Hwang Fuzzy Ranking Algorithm for Solving MADM Problems

Considering fuzzy approaches for solving MADM problems, Chen and Hwang (1992) realized that all approaches have two defects: (1) Most of them have the problem of cumbersome computations. As a result, none of them is suitable for solving problems with more than ten alternatives associated with more than ten attributes. This drawback certainly limits their applicability to real world problems. (2) Some of these approaches require that the elements in the impact matrix be presented in a fuzzy format, even though they are crisp in nature.

In fuzzy MADM methods, qualitative numbers convert to fuzzy numbers. Bass and Kwakernak (1977), Bonissone (1982), Chen (1988), Efstathiou and Rajkovic (1979), Efstathiou and Tong (1982), Kerre (1982) and Wenstop (1976) have proposed scales for converting qualitative numbers to fuzzy numbers, and then applied the fuzzy MADM method for solving MADM problems. Chen and Hwang (1992) identified a new scale with a combination of previous scales and proposed a new approach for solving fuzzy MADM problems which is less complex and needs less computation. For instance, they claim that applying their scale makes solving a 50*50 matrix as simple as solving a 5*5 matrix. In addition, MADM

problems can be meaningfully and efficiently solved in a fuzzy environment. The basic assumption of the Chen and Hwang approach is that the MADM problem may contain fuzzy and crisp data. This fuzzy MADM approach is composed of two phases. The first phase which consists of two major steps:

1. Linguistic-term conversion which transforms the impact value into a fuzzy set if they are verbal terms.
2. Conversion from a fuzzy set to a crisp value set where all the fuzzy sets are assigned crisp scores. The results of this phase produce a new impact matrix which only contains numeric data.

In the second phase, a classical MADM method can be utilized to determine the ranking order of alternatives. In this section, the procedure of fuzzy impact transformation is described. Introduction of this MADM method will be presented in this paper.

2.2.1 Linguistic-Term Conversion

A numerical approximation system is proposed to systematically transform linguistic terms to their corresponding fuzzy set. The transformation requires eight conversion scales, as shown in Figure 1.

The conversion scales are proposed by synthesizing and modifying the work of Baas and Kwakernak (1977), and Bonissone (1982). In the procedure of the linguistic-term conversion, the principle is to simply select a scale figure that contains all the verbal terms given by the DM and use the membership function set for that figure to represent the meaning of the verbal terms.

2.2.2 Conversion from a Fuzzy Set to a Crisp Value

The second step of a fuzzy impact transformation is to convert the fuzzy set to crisp scores. In this section, a modified L-R scoring approach based on Jain's (1976, 1977) and Chen's (1985) works is introduced. The crisp core of a fuzzy set M is obtained as follows.

Assume a maximizing fuzzy set and a minimizing fuzzy set which are defined as:

$$\mu_{\max}(x) = \begin{cases} x, & 0 \le x \le 1 \\ 0, & otherwise \end{cases} \tag{2}$$

$$\mu_{\min}(x) = \begin{cases} 1-x, & 0 \le x \le 1 \\ 0, & otherwise \end{cases} \tag{3}$$

The right score of M can be determined using:

$$\mu_R(M) = \sup_x \left[\mu_M(x) \wedge \mu_{\max}(x) \right] \tag{4}$$

Similarly, the left score of M can be determined using:

$$\mu_L(M) = \sup_x \left[\mu_M(x) \wedge \mu_{\min}(x) \right] \tag{5}$$

Given the left and right scores of *M*, the total score of *M* can be calculated using:

$$\mu_T(M) = \left[\mu_R(M) + 1 - \mu_L(M) \right] / 2 \tag{6}$$
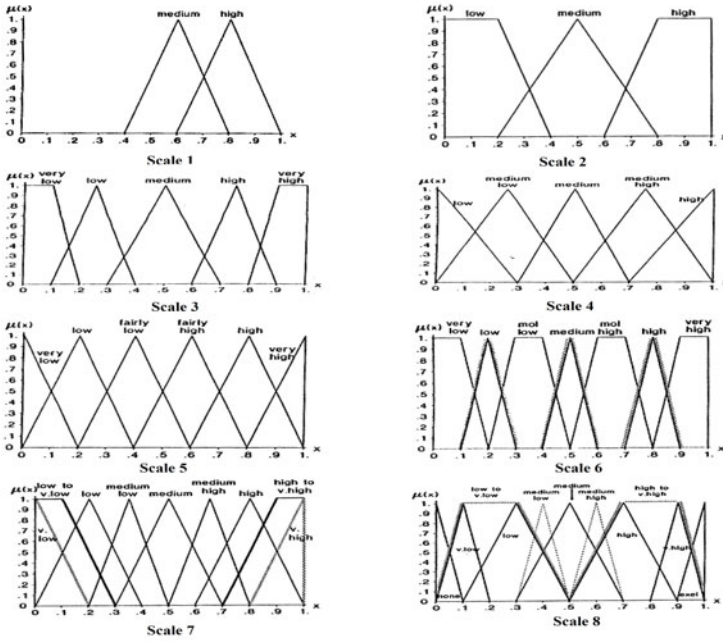
Fig. 1 Scale 1-Scale 8 of transformation

2.2.3 Chen and Hwang Algorithm for Solving MADM Problems

Step1: For each fuzzy attribute, the linguistic expressions are first transformed into a fuzzy set.

Step 2: Crisp scores should be assigned to the fuzzy sets using Equations 4,5,6.

Step 3: In this step, a matrix with fuzzy numbers is converted to a matrix with crisp numbers. Then the alternatives can be ranked by applying any of the MADM methods. In this algorithm, the TOPSIS algorithm is chosen for ranking options.

2.3 Negi's Fuzzy Ranking Algorithm for Solving MADM Problems

Negi believes that every MADM problem with fuzzy numbers can be solved by a classical MADM algorithm with fuzzy numbers which are assumed to be trapezoidal fuzzy numbers. He believes that when we can apply a classical MADM algorithm to solve MADM problems easily, there is no need to apply the Chen & Hwang algorithm for solving MADM problems.

A known MADM method should be applied to explain the Negi's approach. Assume $D=[x_{ij}]_{m*n}$ is a decision matrix where x_{ij} are fuzzy or crisp numbers. If x_{ij} is a fuzzy number, it shows as a trapezoidal number as: $x_{ij}=(a_{ij}, b_{ij}, c_{ij}, d_{ij})$ and the fuzzy weights are assumed as $w_j=(\alpha_j, \beta_j, \gamma_j, \sigma_j)$ (Chen and Hwang 1992).

2.3.1 Negi's Algorithm for Solving MADM Problems

Step1: For given decision matrix $D=[x_{ij}]_{m*n}$, calculate the normalized (unit free) matrix $D^*=[x_{ij}]_{m*n}$. The normalized fuzzy value r_{ij} is defined as:

$$r_{ij} = x_{ij}/x_j^* \quad \forall j \qquad \text{If } j \text{ is a benefit attribute} \qquad (7)$$

$$r_{ij} = x_j^-/x_{ij} \quad \forall j \qquad \text{If } j \text{ is a cost attribute} \qquad (8)$$

When x_{ij} is a crisp number, its corresponding r_{ij} should be a crisp number. When x_{ij} is a fuzzy number, its corresponding r_{ij} should be a fuzzy number. Where x_{ij} is the attainment of attribute j by alternative i is assumed to be a trapezoidal number and:

$x_{ij}=(a_{ij}, b_{ij}, c_{ij}, d_{ij})$, $x_j^-=(a_j^-, b_j^-, c_j^-, d_j^-)$ and $x_j^*=(a_j^*, b_j^*, c_j^*, d_j^*)$

Step 2: Compute the weighted normalized decision matrix $V=[v_{ij}]_{m*n}$. the weighted normalized fuzzy value v_{ij} is defined as:

$$v_{ij} = r_{ij} \cdot w_j \quad \forall i, j \qquad (9)$$

When r_{ij} and w_j are crisp numbers, their corresponding v_{ij} should be a crisp number. When r_{ij} and w_j are fuzzy numbers, their corresponding v_{ij} should be a fuzzy number. Where w_j is the importance (weight) of attribute assumed to be a trapezoidal number.

Step 3: Determine the positive ideal A^* and the negative ideal A^- solutions:

$$\begin{aligned} A^* &= [v_1^*, \ldots, v_n^*] \\ A^- &= [v_1^-, \ldots, v_n^-] \end{aligned} \qquad (10)$$

Where $v_j^*=\max_i v_{ij}$ and $v_j^-=\min_i v_{ij}$. To find v_j^* and v_j^- we need to compare m fuzzy set v_{ij} ($i=1,2,..,m$). To do this comparison, Negi applied Lee and Li's (1988) method where:

$$M_u(v_{ij}) = \frac{\left(-a_{ij}^2 - b_{ij}^2 + c_{ij}^2 + d_{ij}^2 - a_{ij}b_{ij} + c_{ij}d_{ij}\right)}{\left(3(-a_{ij} - b_{ij} + c_{ij} + d_{ij})\right)} \qquad (11)$$

Lee and Li (1988) pointed out that human intuition would favor a fuzzy number with a higher mean value and at the same time, a lower spread.

Step 4: The separation measures are obtained (distance to positive ideal and negative ideal solutions) for each alternative as:

$$S_i^* = \sum_j D_{ij}^* \qquad i = 1, \ldots, m \tag{12}$$

$$S_i^- = \sum_j D_{ij}^- \qquad i = 1, \ldots, m \tag{13}$$

Where D_{ij}^* and D_{ij}^- are calculated as follows for crisp numbers:

$$D_{ij}^* = \left| v_{ij} - v_j^* \right| \tag{14}$$

$$D_{ij}^- = \left| v_{ij} - v_j^- \right| \tag{15}$$

and where D_{ij}^* and D_{ij}^- are calculated as follows for fuzzy numbers:

$$D_{ij}^* = 1 - \left(\underset{x}{Sup} \left[\mu_{v_{ij}}(x) \wedge \mu_{v_j^*}(x) \right] \right) = 1 - L_{ij} \qquad \forall i, j \tag{16}$$

$$D_{ij}^- = 1 - \left(\underset{x}{Sup} \left[\mu_{v_{ij}}(x) \wedge \mu_{v_j^-}(x) \right] \right) \qquad \forall i, j \tag{17}$$

As D_{ij}^* and D_{ij}^- are crisp values, S_{ij}^* and S_{ij}^- will be crisp.

Step 5:The relative closeness to the positive ideal solutions is defined as:

$$c_i = \frac{S_i^-}{S_i^- + S_i^*} \tag{18}$$

Step 6: Alternatives are ranked in descending order of their relative closeness values.

3 Proposed Method for Choosing the Most Appropriate Fuzzy Ranking Algorithm

In this section we compare three fuzzy ranking algorithms: SAW, Negi, and Chen and Hwang, in terms of computation and time consumption to choose the best fuzzy ranking algorithm for solving large size FMADM problems. These three algorithms are designed to solve MADM problems with both crisp and fuzzy numbers and are widely applied to solve MADM problems.

Let m be the number of alternatives and n the number of attributes for a MADM problem. Considering the computations of these three fuzzy ranking algorithms, the number of every mathematic operator which is used in these algorithm's computations is calculated. The results are summarized in Table1.

Table 1 Mathematic operators applied among 3 fuzzy ranking algorithms computations

computations	SAW	Negi's approach	Chen & Hwang approach (Using TOPSIS)
+,*,-,/	13*(m*n)	20*(m*n) + m	12*(m*n)+2*m
Calculation of supremum	---	2*(m*n)	2*(m*n)
Fuzzy numbers comparison	m!	4*(m*n)	---
Crisp numbers comparison	---	m!	2*(m*n) + m!

In the SAW algorithm, to calculate $\mu_i(u_i)$ Bonissone's approach (1982) is applied. Bonissone assumed that crisp or fuzzy numbers in fuzzy problems can be estimated by trapezoidal numbers and fuzzy functions can be applied for calculating U_i. Negi applied Lee and Li's (1988) method to compare fuzzy numbers (Chen and Hwang 1992). In this paper, we also applied Lee and Li's (1988) method to compare fuzzy numbers in the SAW algorithm, to use the same fuzzy number comparision method in these two algorithms and make our comparision more accurate. In these two algorithms, fuzzy numbers comparisons are therefore converted to crisp numbers comparisons and the results in Table 1 are changed as follows (Table 2).

Table 2 Mathematic operators applied among 3 fuzzy ranking algorithms computations

computations	SAW	Negi's approach	Chen & Hwang approach (Using TOPSIS)
+,*,-,/	31*(m*n)	33*(m*n) + m	12*(m*n)+2*m
Calculation of supremum	---	2*(m*n)	2*(m*n)
Fuzzy numbers comparison	---	---	---
Crisp numbers comparison	m!	4*(m*n) +m!	2*(m*n) + m!

Chen and Hwang (1992) claimed that most fuzzy ranking algorithms have the problem of cumbersome computations. As a result, none of them is efficient for solving large size MADM problems ($m>10$ and $n>10$), except their approach. But as can be seen from the calculations comparison in Table 2, SAW's calculations are less than those in the Chen and Hwang algorithm. The Chen and Hwang algorithm has the second rank, and both methods apply fewer mathematic operators than Negi's algorithm. Therefore, despite Chen and Hwang's claim, SAW could be efficient for solving large size MADM problems, but the difference between cost and benefit attributes to assess alternatives is not mentioned in the SAW algorithm. To solve such problems, then, we should apply the Negi or Chen and Hwang algorithm, and according to the comparison results in Table 2, Negi's algorithm is not suitable for solving large size problems. Therefore, when we have a large size FMADM problem with cost and benefit attributes, the Chen and Hwang algorithm would be the best choice.

In some decision situations such as selecting an alternative from a short list, the decision outcome produced by some MADM methods may not differ (Belton 1986; Karni et al. 1990). In addition, considering the results in Table 2, the calculation time would be reasonable in all three algorithms when m and n are small, therefore the most appropriate algorithm for solving small scale MADM problems would be the simplest one, and in such problems, SAW would be the best choice.

In conclusion, the following conditions could be applied to choose the most appropriate fuzzy ranking algorithm for solving every FMADM problem size.

Let $J = \{j | n_j$ is a benefit attribute$\}$ and $J' = \{j | n_j$ is a cost attribute$\}$

If (m>1 and n>1) then

  If ($J=\phi$ or $J'=\phi$) then

    SAW algorithm is appropriate

  Else

    Chen and Hwang algorithm is appropriate

4 Software Development and Empirical Study

We developed a software to simulate these three algorithms and compare their time consumption for solving FMADM problems (Figure 2). In this software, we
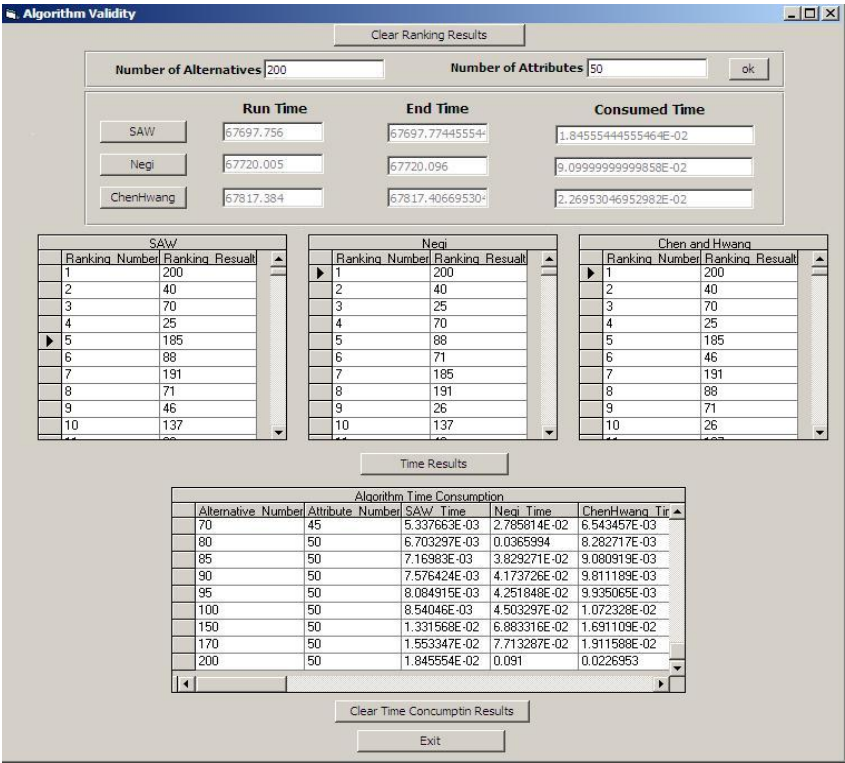
Fig. 2 Software interface

Table 3 Time consumption by SAW, Negi and Chen & Hwang algorithms

Alternatives (m)	Attribute (n)	Time Consumption (Millisecond)		
		SAW	Negi	Chen & Hwang
4	4	0.001	0.002	0.001
8	8	0.001	0.010	0.002
10	10	0.002	0.016	0.002
15	20	0.005	0.027	0.007
15	40	0.010	0.056	0.013
25	40	0.016	0.091	0.021
40	40	0.027	0.144	0.034
45	40	0.030	0.162	0.037
50	40	0.035	0.180	0.042
60	45	0.045	0.238	0.056
70	45	0.053	0.279	0.065
80	50	0.067	0.366	0.083
85	50	0.072	0.383	0.091
90	50	0.076	0.417	0.098
95	50	0.081	0.425	0.099
100	50	0.085	0.450	0.107
150	50	0.133	0.688	0.169
170	50	0.155	0.771	0.191
200	50	0.185	0.910	0.227

compare different FMADM problem sizes by considering the number of attributes and alternatives. In addition types of all FMADM problems attributes assumed to be benefits.

The software comparison results are illustrated in Table 3. As can be seen from this simulation results, SAW has the least time consumption to solve MADM problems with different number of alternatives and attributes. Chen and Hwang's algorithm has the second rank, and both of them consume much less time than Negi's algorithm. So, despite Chen and Hwang's claim, the SAW algorithm could be efficient for solving large size MADM problems with cost or benefit attributes, and the Chen and Hwang algorithm would be the best choice for solving large size FMADM problems when they involve both cost and benefit attributes.

5 Conclusion and Future Works

In regard to the importance of applying the most appropriate fuzzy ranking algorithm for solving given fuzzy multiple attribute decision making problems which may involve both fuzzy and crisp numbers and both cost and benefit attributes, this paper has proposed a new method for choosing the most appropriate fuzzy ranking algorithm for solving FMADM problems, based on the type and number of attributes and the number of alternatives. In addition, we developed a software

to simulate SAW, Negi and, Chen and Hwang fuzzy ranking algorithms and compare their efficiency.

Concerning the importance of determining the weight of all attributes and the score of alternatives under all attributes, which are obtained through the decision maker's judgement, a Sensitivity Analysis System should be developed to analyze the results obtained after changing the attributes' weights and alternatives' scores. Via this Sensitivity Analysis System, the reliability of the decision making process could be assessed.

References

1. Baas, S.M., Kwakernak, H.: Rating and ranking of multiple aspect alternatives using fuzzy sets. Automatica 13, 47–58 (1977)
2. Belton, V.: A comparison of the analytic hierarchy process and a simple multi-attribute value function. European Journal of Operational Research 26, 7–21 (1986)
3. Bonissone, P.P.: A fuzzy sets based linguistic approach: theory and applications. In: Gupta, M.M., Sanchez, E. (eds.) Approximate reasoning in decision analysis. North Holland, Amsterdam (1982)
4. Chen, S.H.: Ranking fuzzy numbers with maximizing set and minimizing set. Fuzzy set and systems 17(2), 113–129 (1985)
5. Chen, S.J., Hwang, C.L.: Fuzzy multiple attribute decision making. Springer, Berlin (1992)
6. Chen, S.M.: A new approach to handling fuzzy decision-making problems. In: Proceedings of the 18th International Symposium of Multiple Valued logic, Spain, pp. 72–76 (1988)
7. Colson, G., Bruyn, C.: Models and methods in multiple criteria decision making. Pergamon, Oxford (1989)
8. Dyer, J.S., Fishburn, P.C., Steuer, R.E., Wallenius, J., Zionts, S.: Multiple criteria decision making, multi attribute utility theory: the next ten years. Management Science 38,
645–653 (1992)
9. Efstathiou, J., Rajkovic, V.: Multi attribute decision making using a fuzzy heuristic approach. IEEE Transactions on System Man and Cybernetics 9, 326–333 (1979)
10. Efstathiou, J., Tong, R.: Ranking fuzzy sets: a decision theoretic approach. IEEE Transactions on System Man and Cybernetics 12, 655–659 (1982)
11. Fishburn, P.C.: Additive utilities with incomplete product set: applications to priorities and assignments. ORSA Publication, Baltimore (1967)
12. Hwang, C.L., Yoon, K.: Multiple attribute decision making. Springer, Heidelberg (1981)
13. Jain, R.: Decision making in the presence of fuzzy variables. IEEE Transactions on Systems Man and Cybernetics 6, 698–703 (1976)
14. Jain, R.: A procedure for multi aspect decision making using fuzzy sets. International Journal of System Science 8, 1–7 (1977)
15. Karni, R., Sanchez, P., Tummala, V.M.R.: A comparative study of multi-attribute decision making methodologies. Theory and Decision 29, 203–222 (1990)

16. Kerre, E.E.: The use of fuzzy set theory in electro cardio logical diagnostics. In: Gupta, M.M., Sanchez, E. (eds.) Approximate Reasoning in Decision Analysis. North-Holland, Amsterdam (1982)
17. Lee, E.S., Li, R.L.: Comparison of fuzzy numbers based on the probability measure of fuzzy events. Computers and Mathematics with Applications 15, 887–896 (1988)
18. MacCrimmon, K.R.: Decision making among multiple attribute alternatives: a survey and consolidated approach. Rand memorandum RM-4823-ARPA, Washington, DC (1968)
19. Olson, D.L., Moshkovich, H.M., Schellenberger, R., Mechitov, A.I.: Consistency and accuracy in decision aids: Experiments with four multiattribute systems. Decision Sciences 26, 723–748 (1995)
20. Stewart, T.J.: A critical study on the status of multiple criteria decision making: theory and practice. Omega 20, 569–586 (1992)
21. Wenstop, F.: Fuzzy set simulation models in a systems dynamic perspective. Kybernetes 6, 209–218 (1976)
22. Yeh, C.H.: A problem-based selection of multi-attribute decision-making methods. International Transactions in Operational Research 9, 169–181 (2002)
23. Zanakis, S.H., Solomon, A., Wishart, N., Dublish, S.: Multi-attribute decision making: a simulation comparison of select methods. European Journal of Operational Research 107, 507–529 (1998)
24. Zeleny, M.: Multiple criteria decision making. McGraw-Hill, New York (1982)

From Process Control Systems towards Virtual Automation Networks – Contributions of Computer Science

Peter Neumann

With the availability of microprocessors from the early nineteen-seventies onwards a new era began, which saw an enormous variety of computer-based technical solutions for different application fields. As a consequence, a great need for communication solutions with regard to those application functions arose. These had to be executed as software, which was logically connected but technically allocated to distributed equipment. In this paper the influence of this development on automation systems will be discussed, particularly the special subject of Industrial communication with its high demand on the Quality of Service.

1 Introduction

In the nineteen-sixties and early seventies, computer power was concentrated in process control computers. However, in the seventies the basic idea arose to distribute the computer power of an automation system to several stations (process stations/base stations for data acquisition, processing and output; operator terminals for the man/ machine interaction) connected by means of digital communication systems. This principle is still valid today but the degree of distribution has increased drastically. This was made possible by the availability of enormous computing power and a broad spectrum of industrial communication systems. Nowadays, hierarchical multi-level computer architectures connected with industrial and office communication systems can be used. They enable the data throughput between the various levels of hierarchy.

Recent research is directed towards Virtual Automation Networks using heterogeneous Wide Area Networks as well as (industrial) standardized local communication systems, which very appropriate due to name-based addressing and high levels of Quality of Service.

To summarize, in the nineteen-seventies distributed computer-based data processing was introduced, and it remains an effective way of utilizing computer science in all areas of automation even today. The following contributions made

Peter Neumann
e-mail: `peter.neumann@ifak.eu`

by computer science facilitated the introduction of Industrial Communications into the area of automation (referring to research activities in Magdeburg):

- Demands to Fieldbus Systems based on experience with Process Control Systems (PCS) communication approaches [4], [6], [7], [9], and [15].
- Theoretical basis of formal description and computer-aided engineering of data exchange processes between the installed devices of PCS [1-4], [5], [8], [10-12].
- Methods of Protocol Engineering used to Fieldbus systems [16-20], [29].
- Performance of Industrial Communication systems [11-12], [21].
- Formal specification of Fieldbus systems (particularly PROFIBUS and its predecessors) [14] [22], [32].
- Important contribution to national and international standardization of Fieldbus systems including development of prototypes [23], [24].
- Introduction of Ethernet into automation including performance investigations [33], [35-42].
- Test methods for conformance test interoperability test and safety test [26-28], [34].
- Methodology for formal specification and test of Fieldbus profiles [19], [25].

This paper explains the applied principles of distributed real-time data acquisition and processing, important steps and results of the research work conducted in the field of Industrial Communications as well as the influence of computer science activities (especially in Magdeburg) from the nineteen-eighties to this day.

2 Demands on Digital Communications from the Point of View of the PSC Development

2.1 From Process Computer to Process Control System

Process computers were characterized by real-time operating systems connected with the technical process via peripheral devices (analogue and binary data input, data output). This man/machine interaction was realized by means of alphanumerical monitors and keyboards (or light pens). Application programs were implemented using assembler languages. Initiated by the process industry, Process Control Systems, which were characterized by dedicated computer equipment (Process Station, Operator Station, and Engineering Station), were developed. Data exchange between these stations was realized by serial digital Transmission systems using different media access mechanisms (Polling, Token rotation, Random access). The application functions were functionally separated, but the equipment was locally concentrated in a dedicated control room. This means, the cables in the field area could still be used further. Figure 1 depicts the principal architecture of a Process Control System.
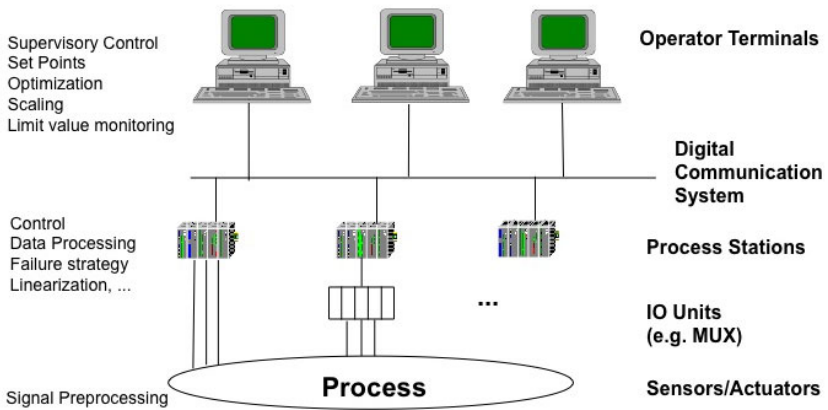
Fig. 1 Principal architecture of a Process Control System

An early example was the PCS "TDC 2000" (Honeywell, 1975), later followed by "Teleperm M" (Siemens) and different, application-specific PCSs (Hartmann &Braun, late seventies) as well as "audatec" (GRW, early eighties). The dedicated research in Magdeburg was directed towards the scientific correct design and parameterization of the digital transmission system, and the performance aspects regarding the distribution of application functions towards the stations [1-11].

2.2 From Process Control System to Locally Distributed Control System (DCS)

The next step was aimed at the local separation and distribution of information processing programs to Field Devices. This was requested mostly by the Manufacturing industry (particularly in Germany). The kernel of these requests was the development of Fieldbus, a digital data transmission system connecting Field Devices with central Processing Devices (Programmable Logic Controller PLC, Operator Station, Engineering Station). This led to further requirements:

- Real-time behaviour (performance, synchronization mechanisms, guaranteed response time, etc.)
- Failure protection (data redundancy, hardware/software redundancy, automatic start/restart, etc.)
- Security (Functional safety; Data security)
- Environmental conditions (operation in harsh locations, explosion protection, degrees of protection).

Furthermore, there was a wide variety of sensors and actuators. Thus, the installations contained equipment from different vendors (multi-vendor systems). These particular circumstances required:

- Conformance of Fieldbus interfaces following a unique standard to be validated by conformance tests
- Unification of application functions (for automation) and of functions directed at increasing the efficiency of Engineering work (commissioning, maintenance, version handling, order handling etc.) by standardization of device Profiles
- Interoperability of Field Devices from different vendors but with the same Application Profile, to be validated by a vendor-independent interoperability test.

Figure 2 depicts a typical DCS with Fieldbus and Field Devices.

The phase described above lasted from 1990 to 2005.
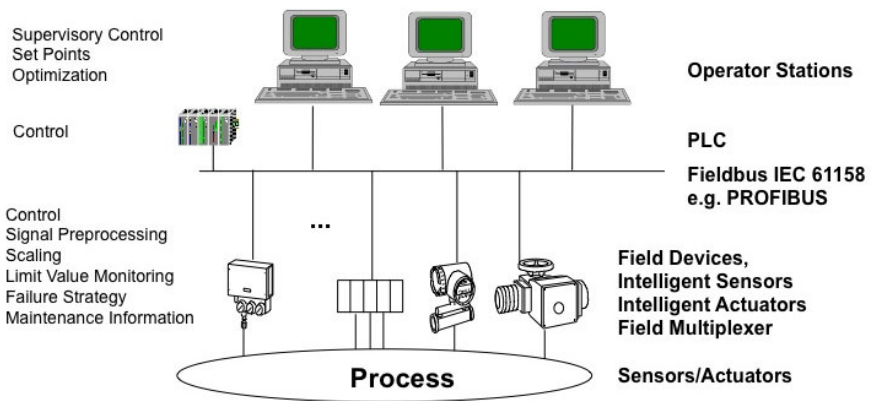
Fig. 2 Principal architecture of a DCS with Fieldbus and decentral Field Devices

Over the last decade, research activities have been directed towards the usage of ETHERNET technology for Industrial Communications. However, some further effort was needed to make it fit for usage in Industrial Communications. A unified data throughput to all levels of the hierarchy provides a substantial benefit. The activities mentioned above required the following additional effort (as compared with the Fieldbus technology):

- Protocol Engineering (Specification of Ethernet adaptations as well as the higher OSI layers) for PROFINET, based on experience with PROFIBUS
- Adaptation of Profile specifications for Field Devices
- Extension of test methodology, test scenarios, and test systems.

2.3 From Locally Distributed DCS to Geographically Distributed DCS

Due to globalization, the requirements on information exchange between geographically distributed enterprise units have increased rapidly. The Internet has

enabled communication between remote offices. Nevertheless, the production units are still regarding data connection predominantly in isolation. One reason for this is the island character of the Fieldbus systems with a range of only a few
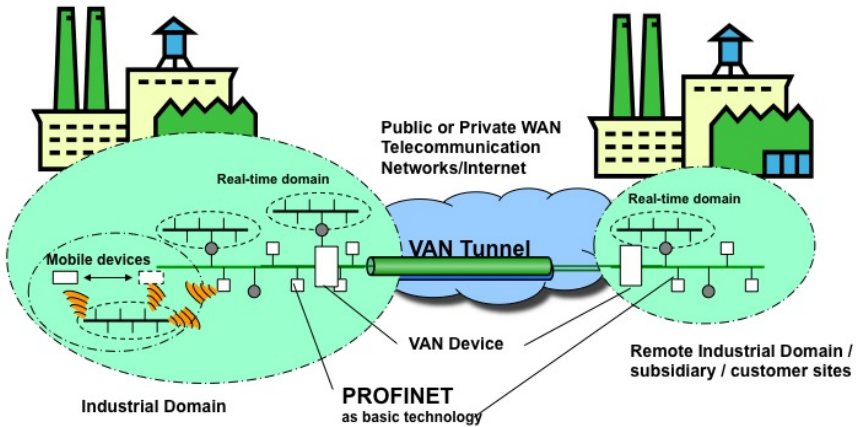
Fig. 3 Geographically distributed DCS

- kilometers and no dedicated security mechanisms to protect them. The aim is a secure end-to-end communication (in the sense of the requested Quality of Service QoS) between two or more geographically distributed automation functions, running in distributed installed equipment. That connection has to work without the ownership of the communication systems used (Figure 3). The Ethernet technology offers the technical basis for global data access. The general QoS requirements for DCSs (real-time behavior, failure protection, safety, security) are also valid for geographically distributed DCSs.

Yet, there are further prerequisites:

- Data security: There are many security approaches for local networks as well as the Internet. However, they are not entirely adequate regarding the requirements of automation especially organizational measures (unauthorized access, non-certified, vendor-specific tools used for commissioning and maintenance, enterprise-internal attacks caused by frustration, etc.)
- Local automation networks are managed by the automation department. Office networks are managed by the computer department of the enterprise. The computer departments have the sovereignty over the ports. Caused by the joint usage, there are competency problems putting the network security at risk.
- Network heterogeneity: The automation expert does not have management access to public or private Wide Area Networks (WAN). Thus, he cannot influence the guaranteed performance (by service level agreement) of the data channel. However, there are essential methods to solve this problem, e.g. monitoring

the WAN and mechanisms to save the guaranteed Quality of Service (e.g. switching to alternative channels).

3   Industrial Communication Systems and Related Computer Science Areas

3.1   Protocol Engineering

In the standardization of Communication a high quality of protocol specifications is necessary to minimize the risk of failures as early as possible within the process. For Fieldbus development the specification languages ESTELLE, SDL and UML have been used.

A few research activities conducted in Magdeburg with regard to Fieldbus development:

- (Partially) formal specification of Fieldbus architecture, particularly for the market leader PROFIBUS and its predecessors during the international standardization (nineties until 2005) [19], [22], [25], [29-31].
- (Partially) Protocol Engineering of PROFINET (Ethernet adaptations as well as higher OSI layers) [30-31], [37].

3.2   Specification and Standardization of Device Profiles

To support (or enable) the interoperability and changeability of devices from different vendors, the specification of vendor-independent (automation) application functions in Field Devices as well as in PLCs has been done in "Profiles". Therefore, a specification method, different formal device description languages and many profiles have been developed. They enable us to improve the efficiency of the commissioning and maintenance of Field Devices and to offer a common data basis for management processes. Relevant research contributions were:

- Development of specification method [14], [16], [22], [32]
- International standard of device description language
- Responsibility over specification of many device and application profiles [13], [19], [25]
- Method of programming of application programs for automation [23], [24]
- Adaptation of profile specifications for Field Devices on Ethernet-based Industrial Communications.

3.3   Test of Field Devices

DCSs consist of many devices offered by different vendors (multi-vendor system). To guarantee their conformance to standards, interoperability, and QoS requirements,

all vendor-based solutions have to be certified. Therefore, comprehensive vendor-independent tests for the following test areas have to be implemented:

- Conformance test: Test of implementation against protocol specification using a conformance test system
- Interoperability test: Test of correct co-operation of application functions of Field Devices with functions of devices from different other vendors. The proof includes: Conformance, behavior of application functions, multi-vendor ability.
- Safety test: Test of implemented safety properties against the (as much as possible formal described) Safety specification
- Performance test: Proof of real-time behavior.

Related research work in Magdeburg has been:

- Development of conformance test methods and systems for PROFIBUS [26-28], [34]
- Development of interoperability test methods and tools [13], [19]
- Extension of test methods, test scenarios and test systems, adaption on Ethernet-based communications (PROFINET)
- Establishment of a test center, which operates globally.

4   Virtual Automation Network (VAN)

4.1   Motivation and Requirements

Recent developments are aiming at the usage of heterogeneous networks, which cover local networks or WAN, wired or wireless networks, public or private networks, provider-based or provider-less networks. The requirements on heterogeneous networks in industrial automation arise from future scenarios of distributed automation using geographically distributed automation functions [37], [41], [42]:

- Centralized supervision and control of (many) decentralized (small) technological plants
- Remote control, commissioning, parameterization and maintenance of distributed automation systems
- Inclusion of remote expert or external machine-readable knowledge in the plant operation and maintenance.

- The usage of heterogeneous networks in automation requires a guarantee for *scalable QoS* of end-to-end communication connecting the geographically distributed application functions. Furthermore, a formal framework (standardized Application Service Elements ASE) for the handling of distributed automation applications by the automation expert, e.g. IEC 61158 [30], IEC 61784 [31] for Fieldbus systems, needs to be in place. This enables the same design practice as it already exists for homogeneous communication systems of automation,

e.g. PROFINET or Ethernet IP. Thus, an end-to-end connection between two au-
tomation applications has to be established during a connection establishment
phase (runtime tunnel establishment). The tunnel can be used for the exchange of
automation data between the distributed applications during the runtime phase.
This means that, once the tunnel has been established, the exchange of data be-
tween the distributed runtime objects follows the rules of the Fieldbus system used
(e.g. PROFINET with connected PROFIBUS or other connected Fieldbus sys-
tems). These rules are well known to the automation engineer. The mechanisms
used in the runtime phase are the same as the ones in homogeneous communica-
tion systems. The required infrastructure has been called "Virtual Automation
Network".

    From the point of view of an automation expert, the specifics of heterogeneous
networks (e.g. using Web services, transmission technology) remain hidden.

4.2   The VAN Concept

4.2.1   VAN Domains

VAN domains are playing a central role within the VAN concept [35], [36]. In
Virtual Automation Networks the distributed automation function (local and re-
mote parts) forms a domain (figure 4). VAN domain covers all devices, which will
be grouped together on a logical or virtual basis to represent a complex applica-
tion, independent of where the devices are located. Depending on the complexity
of the automation function, a hierarchical domain approach might be appropriate.
This means that there might be sub-domains, which can again consist of sub-
domains etc. Consequently, within a domain (or sub-domain) there will be one or
more automation devices, Fieldbus systems and infrastructure components.
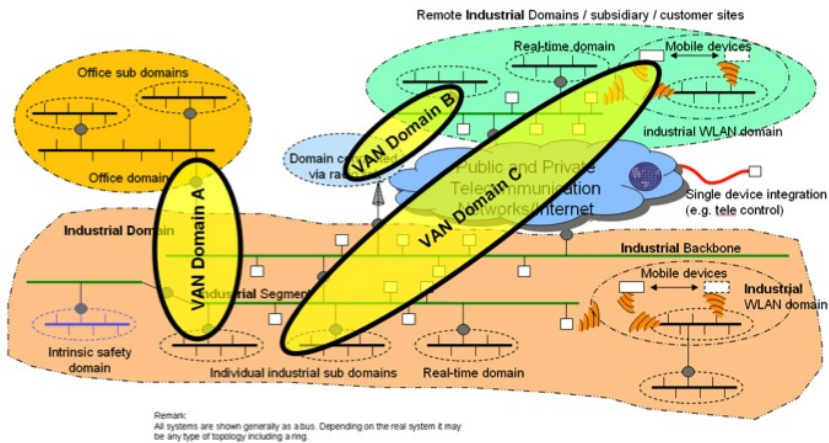
Fig. 4 VAN domains related to different automation applications

A VAN domain can include any given industrial domains (equipped with automation devices and connected via industrial communications, e.g. Fieldbus) either entirely or partially. Thus, an industrial domain can consist of segments related to a VAN domain (VAN segment) or of segments, which are not related to a VAN segment (figure 4). A VAN enabled automation device is connected directly with other VAN enabled devices within the VAN domain, since they are inside the necessary VAN network capabilities. All other devices of a sub-network within a VAN segment of an industrial domain have to be connected via a VAN Access Point or a VAN Proxy [38].

4.2.2 Addressing Concept

A VAN domain is a name space of a geographically distributed automation application. For the establishment of a runtime-tunnel as well as for management tasks, logical addresses (names) are used, independent of the utilized transmission technologies and their addressing mechanisms (e.g. MAC or IP addresses) within a heterogeneous network. The name space can be public (mostly global) or private (mostly local). The public name space can be visible on the Internet. The private name space includes installations, which should not be visible within the public area. To summarize, during a connection establishment between remote applications (i.e. during tunnel establishment) a name based (logical) addressing is used. Once the tunnel has been established, the MAC and IP addresses of distributed devices, which have been negotiated during the process of establishing the tunnel, are used.

4.2.3 Establishment of Connections between Communication Endpoints

During the establishment and maintenance of runtime tunnels, Web services can be utilized. However, the mechanisms will not be described in this paper. To enable a requested QoS (especially failure protection in case of reduced performance of the network channel), monitoring of the channel has been introduced. This monitoring offers the actual performance data at a particular moment. The monitored network channel is described as a black box. In case of significant restrictions, alternative transmission systems could be used, i.e. switched e.g. from DSL to GSM (VAN switching). These alternative transmission technologies have to be installed during the design phase. The runtime tunnel remains open in this case. In case of more extensive restrictions, a new route through the network has to be found (VAN routing), i.e. interrupting and automatic establishment of a new tunnel.

4.2.4 Exchange of Productive Data

The exchange of productive data denotes the exchange of data between the distributed automation functions during the runtime phase. It is comparable to the data exchange in local homogeneous networks (e.g. on the basis of Fieldbus systems) in that the utilized heterogeneous network remains hidden. Due to the fact that mostly connection-oriented services are used, two phases have to be distinguished:

- Connection establishment between application objects using the established runtime tunnel. The connection establishment follows the rules of that protocol, which enables the exchange of distributed automation objects (e.g. application layer of an industrial communication system as described in IEC 61158/61784).
- Exchange of cyclic and acyclic data between automation objects regarding the protocols of the involved Fieldbus systems.

4.2.5 Quality of Services

In the use of industrial communication systems, the following QoS categories are relevant (see 2.2):

- Real-time behavior (performance, synchronization mechanisms, guaranteed response time, predictable jitter)
- Failure protection (data redundancy, hardware/software redundancy, automatic start/restart, deployment of new functions)
- Security (Functional safety; Data security)
- Environmental conditions (operation in harsh locations, explosion protection, degrees of protection), not discussed here.

The *real-time behavior* is determined by the transmission technologies used. Typical applications are the ones without strong real-time requirements (non real-time, soft real-time), e.g. remote supervisory, tele-control, remote maintenance. For hard real-time applications (possibly synchronous) developed mechanisms within the local area can be used. In a VAN enabled device, real-time-critical applications (e.g. Motion Control) as well as real-time non-critical applications (e.g. tele-control) can be implemented. The former can be realized using a so-called "Native Interface" that has been enabled with the requested real-time behavior. The latter uses the VAN mechanisms as described above.

Failure protection can be improved by network channel monitoring as well as by VAN switching and VAN routing mechanisms (see 4.2.3).

Functional safety can be enhanced using an additional Safety layer on top of the OSI Application layer as used for local industrial communication systems.

Data security can be improved by security mechanisms that are used in WAN. Of particular importance are also the organizational measures within the enterprises.

4.3 Research Contributions

The idea and the concept was introduced into a European Integrated Project by the author, and was developed and tested between 2005 and 2009 by an international consortium, which consisted of a number of well-known European automation enterprises. The development of products is still outstanding.

Research in Magdeburg made the following contributions:

- Concept, system architecture (protocol engineering) [35], [36], [38], [40], [42].
- Safety (functional safety, security)
- Wireless communications
- Application example "Tele-control"
- Engineering methods and tools

Additional trend-setting work accompanied this research.

5 Conclusion

This paper describes its influence of Computer Science on automation systems, where the focus lies on one specific area, namely "Industrial Communications", a field which is characterized by high demands on the Quality of Services.

Today, Fieldbus systems are well established in local DCSs and the Ethernet-based real-time communication systems are likely to take on a similarly significant role. Using comparable technical solutions, industrial automation and office automation are slowly merging. Most of the applications with standardized background are situated within the local area of an enterprise. Geographically distributes solutions are in their early stages and mostly use proprietary solutions.

There is a need for developing mechanisms for Virtual Automation Networks and Standards in this area. However, the basic concept already exists.

References

[1] Sawatzky, J.: Zur kommunikationsseitigen Beschreibung und Dimensionierung von Mikrorechner- Automatisierungsanlagen. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik, Dissertation (1984)

[2] Orlow, J.K.: Beitrag zur Dimensionierung von störungstoleranten Mikrorechner-Automatisierungsanlagen. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik, Dissertation (1984)

[3] Häßner, H.: Zur Dimensionierung von Mikrorechner-Automatisierungsanlagen bezüglich Verfügbarkeit und Sicherheit. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik, Dissertation (1986)

[4] Schumann, H.-J.: Systematische Einsatzerprobung von Mikrorechnerreglern und re-gelungstechnischen Verfahren. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik, Dissertation (1986)

[5] Fiedler, K.: Experimentelle Ermittlung von Zeitverhaltensparametern am audatec-Bussystem. Forschungsbericht TU Magdeburg - Diplomarbeit (1986)

[6] Barge, W.: Zur Gestaltung von Mikrorechner-Automatisierungsanlagen aus der Sicht der Kommunikation Mensch-Anlage. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik, Dissertation (1987)

[7] Dähmlow, B.: Zur strukturellen und funktionellen Gestaltung des prozessnahen Bereiches perspektivischer Mikrorechner- Automatisierungsanlagen. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik. Dissertation (1987)

[8] Eckstädt, A.: Verhaltensanalyse eines mehrkanaligen, ausfalltoleranten Feldbussystems. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik. Dissertation (1988)

[9] Eichelbaum, P.: Dimensionierung der Speicherkopplung von Mikrorechnern in Mikrorechner-Funktionseinheiten. Technische Universität Otto-von-Guericke Magdeburg, Sektion Automatisierungs- und Elektrotechnik. Dissertation (1988)

[10] Neumann, P.: Verhaltensanalyse und Dimensionierung von Mikrorechner-Automatisierungsanlagen. In: Taschenbuch der Elektrotechnik. Bd. 4. Systeme der Informationstechnik. Verlag Technik, Berlin (1990)

[11] Fiedler, K.: Zeitverhaltensanalyse serieller Bussysteme mittels Lastmethode. Technische Universität Otto-von-Guericke Magdeburg, Fakultät Elektrotechnik, Dissertation (1990)

[12] Hoyer, R.: Leistungsanalyse von Feldbussystemen im Hochlastbereich. Otto-von-Guericke Universität Magdeburg, Fakultät Elektrotechnik, Dissertation (1993)

[13] Diedrich, C.: Interface Layer und Profile für Feldbussysteme sowie deren formale Spezifikation. Otto-von-Guericke Universität Magdeburg, Fakultät Elektrotechnik, Dissertation (1994)

[14] Czekalla, J.: Spezifikationsstile und Entwurfswerkzeuge für Kommunikationsprotokolle im Echtzeitbetrieb. Otto-von-Guericke- Universität Magdeburg, Fakultät Elektrotechnik, Dissertation (1995)

[15] Grüterich, T.: Verarbeitungsanforderungen und Anwendungsstrukturen in real konfigurierten Prozessleitsystemen. Otto-von-Guericke-Universität Magdeburg, Fakultät Elektrotechnik, Dissertation (1996)

[16] Dietrich, D., Neumann, P., Schweinzer, H.: Fieldbus Technology. Systems Integration, Networking, and Engineering. Springer Wien, New York (1999); ISBN 3-211-83394-3

[17] Neumann, P.: Der Feldbus als Motor der Automatisierungstechnik. Atp 41, Nr.7, pp. 27–34. Oldenbourg Industrieverlag (1999); ISSN 0178-2320

[18] Neumann, P.: Locally distributed automation – but with which fieldbus system? Assembly Automation, vol. 19(4), pp. 308–312. MCB University Press (1999); ISSN 0144-0154

[19] Neumann, P., Diedrich, C.: Profile für Feldgeräte. Erforderliche Ergänzungen der Protokollspezifikationen am Beispiel von Profibus-PA. In: Tagungsband FET 1995, Wien 1995, pp. 208–218 (1995)

[20] Neumann, P., Diedrich, C., Simon, R.: Der Weg zu offenen Prozessleitsystemen. In: GMA Jahrbuch 1997, pp. 128–148. VDI-Verlag, Düsseldorf (1996)

[21] Grüterich, T., Neumann, P.: Mathematische Beschreibung von Leiteingriffs- und Meldungsprozessen in Prozessleitsystemen. at- Automatisierungstechnik, vol. 45, pp. 175–180. R.Oldenbourg Verlag, München (1997)

[22] Pöschmann, A., Hähniche, J., Deicke, P., Neumann, P.: Experience with Formal Methods Implementing the PROFIBUS FMS and DP Protocol. In: WFCS 1997, Proceedings of 2nd IEEE International Workshop on Factory Communication Systems, Barcelona, pp. 277–286 (1997)

[23] Neumann, P., Diedrich, C.: Distributed Function Block technology - quo vadis? In: 1st International PLCopen Conference on Industrial Control Programming ICP 1996, Paris (1996) (Preprints)

[24] Neumann, P.: Function Block technology standard. We are on the way towards open, distributed computer control systems. Control & Instrumentation EUROPE London 2, 60–62 (1996)

[25] Neumann, P.: Diedrich, Chr.: Profile für Feldgeräte. Erforderliche Ergänzungen der Protokollspezifikationen am Beispiel von Profibus-PA. In: Tagungsband FET 1995, Wien 1995, pp. 208–218 (1995)

[26] Neumann, P., Henninger, O., Ulrich, A., Nickoll, K.: Automatisierte Testfolgenableitung aus formalen Spezifikationen, Abschlußbericht, Ministerium für Wissenschaft und Forschung des Landes Sachsen Anhalt. In: Förderkennzeichen 949A0541, Otto-von-Guericke Universität Magdeburg, Institut für Automatisierungstechnik (1994)

[27] Henninger, O., Neumann, P.: Test Case Generation Based on Formal Specifications in ESTELLE. In: IEEE-Workshop Factory Communication Systems, Leysin 1995, pp. 131–140 (1995)

[28] Henninger, O., Neumann, P.: Generierung von Testfolgen für den Konformitätstest aus der formalen Spezifikation. In: Tagungsband FET 1995, Wien 1995, pp. 377–383 (1995)

[29] Neumann, P., Diedrich, C., Simon, R.: Necessary extensions of fieldbus systems for distributed processing. In: IEEE- Workshop Factory Communication Systems, Leysin 1995, pp. 242–249 (1995) (preprints)

[30] IEC 61158 (all parts). Industrial communication networks. Fieldbus specifications

[31] IEC 61784-2. Industrial communication networks Profiles. Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3

[32] Simon, R.: Methoden zur Feldinstrumentierung von verteilten Automatisierungssystemen. Otto-von-Guericke Universität Magdeburg, Fakultät Elektrotechnik und Informationstechnik, Dissertation (2001)

[33] Jasperneite, J.: Leistungsbewertung eines lokalen Netzwerkes mit Class-of-Services Unterstützung. Otto-von-Guericke Universität Magdeburg, Fakultät Elektrotechnik und Informationstechnik, Dissertation (2002)

[34] Henniger, O.: Testgenerierung aus Spezifikationen in Estelle und SDL. Brandenburgische Technische Universität Cottbus, Fakultät für Mathematik, Naturwissenschaften und Informatik, Dissertation (2002)

[35] Specification of the Open Platform for Automation Infrastructure. Deliverable D02.2-1. Topology Architecture for the VAN Virtual Automation Domain. European Integrated Project VAN FP6/2004/IST/NMP/2 - 016696 VAN Virtual Automation Networks (2006)

[36] Neumann, P., Pöschmann, A., Messerschmidt, R.: Architectural Concept of Virtual Automation Networks. In: IFAC World Congress, Invited Session Virtual Automation Networks (2008)

[37] Neumann, P.: Communication in Industrial Automation. What is going on? Control Engineering Practice 15, 1332–1347 (2007)

[38] Neumann, P., Pöschmann, A., Flaschka, E.: Virtual Automation Networks. Heterogeneous Networks for Industrial Automation. Atp. International 2, 36–46 (2007)

[39] Neumann, P.: Wireless Sensor Networks in der Prozessautomation. Übersicht und Standardisierungsaktivitäten. Atp. 3, 61–67 (2007)

[40] Neumann, P.: Kommunikation ohne Grenzen. Das VAN-Konzept, Teil 1. Computer Automation 3 (2008)

[41] Neumann, P., Pereira, E.: Industrial communication protocols. In: Handbook of Automation, pp. 981–999. Springer, Heidelberg (2009); ISBN: 978-3-540-78830-0

[42] Neumann, P., Messerschmidt, R.: Virtual Automation Networks. In: The Industrial Electronics Handbook. Industrial Communications Systems, pp. 15-1–15-15. CRC Press, Taylor & Francis Group (2010); ISBN 978-1-4398-0281-6

Control of Distributed Autonomous Systems – How to Get a Swarm to Succeed over an Overwhelming Challenge

Josef von Stackelberg

Abstract. Since Frank Schätzing has published his novel "The Swarm" in 2006, management coaches use it as an image how to guide a heterogeneous group of individuals to be successful, as employees in a company (branch), as team members in a project etc. Thereby they like to neglect that strong rules and control are necessary to keep many individuals on track to pursue one certain and well defined aim.

In principle, there is not much of a difference between the control of distributed autonomous technical systems and the guidance of distributed autonomous individuals which both are expected to achieve an aim, e.g. to produce some complex machine within a certain time as a multi-controlled robot system or to develop a complex new electronic module with a multi-skilled project team.

While engineers know how to define their technical models to describe the distributed functions and their interactions of the complex robot, in management trainings the models which shall describe the "functions" of a team or staff member seem to miss the amount of diversity of individual interests and capabilities, which are the most important factors to prevent a group of people from being successful.

The intention to write this paper is, to show that the electrical rules how to engineer a well-operating control of distributed autonomous systems can be transported into a different kind of engineer's work: The management of teams and staffs.

Summary. When Jesus Christ gathered his Apostles, he promised Simon and Andreas, who were fisher men, in the application interview, to make man-fishers out of them [1]. Within the development of a career, an engineer also has to become a man or a woman who treats with people instead of machines and technical modules, who has to calculate with human skills instead of specified attributes of a component. Although it looks different to be a well-skilled engineer or a manager, there are quite a lot of similarities during daily life, and the technical models which have been learned during the engineer's studies are also valid when leading a team to a project success.

To consider it from this point of view, a project team is a number of distributed autonomous systems, a swarm of individuals, which have to be interconnected to be successful as a system rsp. as a team.

1 Basic Information about Project Management

In industrial, economic or in social life there can be distinguished two different kind of activities: The line production, the standard trading sequences and the daily routines on the one hand and the projects on the other hand; further on in the scientific environment most of the research activities are projects [2]. Hence a project is characterized by the following attributes:

- It is unique in its aim and
- its treatment procedure,
- it has a fixed budget and
- a defined time for finishing and
- it is processed by more than one individual.

Out from this characteristics, the main duties of the head of the project, the project manager, can be transferred easily:

- He has to take care for the budget,
- the time in which the project is fulfilled and
- for reaching the technical, economical or social aims which are defined at the beginning of the project rsp. for which the project has been created.

(When there are used just the male forms for pronouns and subjects, this does not mean sexistic harassment for females, it shall simply keep the text easily readable.)

Therefore he sets up and controls different management parts:

- The task management,
- the time management,
- the resource management and
- the stakeholder management.

Before the management parts are described, there should be added a general rule for the project management rsp. the project manager: As the project manager is the ONE who knows everything considering the PROJECT (not necessarily the technical or economical or social details) it is regarded to be bad project management when the project manager does the workload of the tasks by himself. The project manager takes care that the information which is necessary for the project team members is spread to the ones which need the information, completely and in time, and therefore he gathers the information from other team members. The project manager is the Great Communicator.

In real life, a top project manager has beside his management capabilities enough skills to jump into the task level of the project whenever it is necessary to support a team, because there is missing a skill or a hand. As long as this behavior is not used to show off for ones own reputation and to affront the team members, it is highly appreciated by the team, especially in critical phases when it is

required to keep the motivation spirit of exhausted members in a high level. But this way to manage a project is nearly not to describe with a formal structure and shall not be regarded further on.

2   The Task Management

Like going from one position to a different one by taking a more or less huge number of steps, to fulfill a project means to make working steps. Therefore the total workload is divided into partial projects, the partial projects are divided into working packages, the working packages are divided into tasks. Some of the tasks can be processed in parallel, some depend from each other, e.g. the one can only be started when the other has been fulfilled completely or when the other has already been started. A very helpful tool for the task management is the task list.

The task list contains at least all tasks, each named by a characteristic expression, the time span which is necessary to fulfill the task, and the condition which is necessary to start – and sometimes to fulfill – the task, i.e. which task has to be already started or fulfilled.

The task list can be extended by the information what team member(s) shall be responsible rsp. shall process the task, what materials, tools, supportive actions, teams, companies etc. are necessary to fulfill it and what costs are be grown by the task. At last there can be added some information if a task may concern any carrier of an interest. This information is quite necessary and leads into the stakeholder management which is described later.

The task list is the information base for the whole project planning and controlling. With the task list there can be easily set up the different project plans and monitoring lists; in fact, the task list is a monitoring list by itself.

Task management during the running time of the project consists of starting a task by setting an initial information to the processors, by communicating the details of the tasks and finding answers whenever there occur questions and by watching that the tasks are fulfilled in time. In a complex project there may be running several tasks in parallel; in this case the task management can become an exhausting task by itself.

Further on task management is also used to watch over the task list and taking care, that no task is forgotten to be processed.

3   The Time Management

After the task list is completed, a time plan can be generated easily. Therefore the time relevant information in the task list are added with the beginning time for each task. The time relevant information can be set into diagrams, which the abscissa is the time beam of. There exist some standard diagram forms, like the Gantt or bar graph. An intelligent time plan includes hidden time puffers added to each task. Those time puffers are dimensioned in relation to the risk that the task won't be finished within planned time.

Further on the information about the dependencies between the tasks has to be added to the diagrams.

Within the time plan of a project there are sometimes defined situations, e.g. when a certain and/or important part of the whole development is finished. Those situations are marked as so called mile stones.

The next step is to identify the so called critical path: In a project there are most time several tasks or task series in parallel to other tasks or task series. One of those task series is that one in which there is no time puffer besides the integrated puffers. This series will lead to a delay of the whole project when it is not finished in time. The critical path may change within the project phase, when in a different task series there occurs a delay which exceeds the series time so much that the whole project time is endangered.

The time management activities consist mainly of the monitoring whether the planned time schedule is realized or not. When there occur differences between the plan and the project reality, the project manager has to start activities to come back to the scheduled times, e.g. by increasing the manpower in some tasks to accelerate their processing.

4   The Resource Management

Resources for a project are persons, machines, production units, materials etc.; finally the resources also contain the project budget. The resources have to be planned well and during the project progress they have to be managed carefully.

The resource plan is deviated from the task list. In a complex project the resource plan should be integrated into the time plan to ensure that no resource will be planned to be used twice at a time. In the personnel plan the holiday schedules of the team members have to be considered. Further on it is wise not to plan a full forty hours week for a team member, because there are always different duties and activities beside the project which have to be processed by the people.

A well done resource management also takes into account the real skills of the individuals and is not limited to counting of heads.

During the project progress the resource management keeps an open eye on everything which deviates from the plan, keeps the people informed well in time to avoid delays and is available for unseen problems which the team members need support for.

5   The Stakeholder Management

Stakeholders in a project are all people which have an interest in the success or nonsuccess of the project or are touched by the project in any kind. Stakeholder management starts with writing a stakeholder list which contain all persons who are touched by the project, the interests of the stakeholders concerning the project and – if necessary – possible measures to improve a negative situation. An important rule is to keep the stakeholders informed and in a good mood. Stakeholder

management is the most sensitive part of project management and is too often underestimated, concerning the influence on a positive project result.

6 The Electrical Model for Corresponding Distributed Autonomous Systems

Like the electronic engineer has to read precisely each detail of a component's specification which he intends to implement into the circuit, the project manager has to evaluate as precise as possible the personalities, capabilities and skills of his team members. While for a power stage of an amplifier only a transistor will be used which voltage and current limit values correspond with the power stage requirements, only a development engineer makes sense to be consigned with some development tasks who has the knowledge, e.g. how to design a proper power stage, when a power stage is needed.

Further on, as each electronic component has its characteristic time behavior, each individual has its personal working velocity, this may be acceptable or not. In case it's not acceptable, there has to be found a different team member. If there cannot be found a specialist who complies to the requirements, the time and the resource planning has to be modified.

In principle, each member of a project team can be considered to be an autonomous controller. As the controllers are not fixed together, they can be considered as being distributed systems. When the team members are linked by the project rsp. its project manager, they can be considered to be corresponding distributed autonomous systems. Further on, as most time one of the project requirements is to fulfill the dates for the mile stones and the finishing of the project, e.g. when the keys of the building are delivered to the – hopefully satisfied – customer, the project is bound to the requirements of so-called real time systems.

Planning a well-operating real time system, care has to be taken for the time requirements of the controlled environment on the one hand and for the time behavior of the controller system on the other hand and link them together in that way that the environment (process) has never to wait for the controller.

Each individual engineer takes his individual time to understand the requirements he has to fulfill within his development task (or test task), he needs his individual time to breed the solution, to make his calculations, draw the diagrams etc. There is no sense to try to push a creative act. Well-done project management considers those character attributes of the team members and plans with those attributes.

Same is valid for the planning of production tasks, where e.g. placement machines are used to provide test units, where after the soldering process some time is necessary to check the placement and soldering, where a certain amount of misplacement has to be assumed etc.

Further on the procurement of parts and components can become time relevant. It's not seldom even for "simple" electronic components that there occur lead times which are nearly as long as the whole development time for a complex

electronic circuit. In this case, when it is not possible to substitute those kind of components, they have simply to be considered within the time plan.

It's also necessary to consider the time behavior of the communication between the controller systems and between controller systems and the master, e.g. a well-educated engineer sometimes bears more complex and condensed technical information than a low-educated supporting craft. Hence during a project meeting it is inevitable that point-to-point information is sent with the signal velocity and density which can be received by the information sink and the broadcast velocity and density is adapted to the weakest sink. Therefore the time span for a project meeting has to be planned according the communication requirements of the communication partners.

Summarized, a project is a highly deterministic process with high demands for real time control functions with which the distributed autonomous systems have to be coordinated. The master function of the control system is taken over by the project manager. Experience says, that a project – a system of distributed autonomous controllers – is not working properly when not mastered – controlled – properly by a project manager, even if there exists a detailed, complete and correct planning system from the beginning and if this planning system is broadcast, discussed and agreed with all team members. It does not work because the singular team member is also an individual with individual understandings, priorities and aims, and there cannot be planned every maybe-cause in the future, such as changing lead times or the moment in which the genius engineer finds the breakthrough for the required solution.

Those indeterminablenesses are similar to the imponderables of electronic distributed autonomous systems: Each controller has its tasks, which it processes with its priorities and aims and its systematic faults e.g. in the software, and the changes within the process which the controller is used to control, besides the stochastic disturbances because of e.g. electromagnetic incompatibilities which cause communication failures.

7   Control of a Swarm

In general a swarm is a more or less huge amount of more or less similar systems; basically the expression has been used for swarms in the nature [3, 4]. In younger times it's been settled also for different systems like groups of human beings or for technical control systems, mostly in combination with "intelligence". Hence in extended meaning, a project team can also be considered to be a swarm, although a project team is different from a conventional swarm insofar, as the members are heterogeneous with reference to their education, their knowledge and the tasks they have to process. The similarity to the swarm model refers mostly to the fact, that each team member shall know everything about its tasks and shall know who are its immediate neighbours, e.g. the layouter of a certain printed circuit board and the development engineer of this PCB.

Within the general model also the project manager is part of the swarm, with his special knowledge, tasks and duties as described before; his responsibility for the project success is just considered to be a task.

As long as everything within the project runs like modeled in the project planning, the project runs fine and smooth. But, as experience teaches, there is no project which runs fine. There are numerous reasons to spoil the time plan, e.g. technical requirements are found to be unsolvable within the team, lead times extend from weeks to months as somewhere in the world occurs a natural catastrophe, team members become pregnant or die etc. Within a team of human beings there are also the problems with bad chemistry between the one and the other member.

To take the technical requirement which is unsolvable within the team as an example: In this case there are three alternatives to solve the problem:

- Talk to the customer that the requirement cannot be solved and either give the development order back to the customer or try to modify the requirement,
- find rsp. hire a specialist which is known for his knowledge concerning this special part of technical issues and hope he will find a solution within acceptable time or
- give the problem to the swarm by founding a round table and gathering the team members in the hope one of them will have an idea.

The first and the second item to solve the problem are in a way similar, in the first one it's up to the customer to find the specialist, in the second the specialist shall be found by oneself. For the third alternative experience again teaches that the probability to get a solution is quite low, for several reasons:

- At a round table people either don't understand the facts because of their different knowledge and simply disturb the really creative thinking by more or less idle chat or
- they don't dare to talk about their thoughts as they don't know whether their thoughts are relevant or not or
- solutions are just decided to be because of one on the table being the type of human being who likes to decide anything or
- nobody dares to decide a solution to be as the positive effect of the possible solution cannot be seen immediately.

As the controller of the swarm and a member of the swarm by himself the project manager needs at least the superiority in knowledge or instinct to detect a problem and lead it into the right direction, to a competent team member, to an external specialist or back to the customer. This superiority in knowledge or instinct may be considered as being a standard skill of a project manager. If it were, round tables won't be founded to solve problems.

8 Conclusion

The management of a project or even of a department of a company is close to the engineering process of a controller system of distributed autonomous systems, with the specialty, the human beings which are implemented in this controller system are more complex than an ordinary computer. That does not mean, that they are not calculable and predictable in their behavior.

Therefore to become a man-fisher means for a person, to get a deep understanding of the team members to be able to lead them, as the engineer must know the specification details of his electrical components very well to design a fine working circuit or machine.

To be a man-fisher and to have an understanding of the team members means also to understand their profession to make the right decision or to trust fully to the professional competence and decision of the team member. The latter does only work in a very few cases, as experience teaches.

There is another way to lead a group of human individuals, the way of order and obedience. In this case no understanding of anything is required, but here – as experience teaches also – the efficiency of the team is quite low, as members are either over- or undercharged. While the former leads finally to output zero as the overcharged team member not only cannot deliver a part of the requested results, at one point he even loses the contact to the process, the latter will lead to a minimum output as the member is more in charge with finding ways to trick the leader than to work on the duties.

Literature

[1] The New Testament, Gospel according to Markus, ch. 1, Verse 17
[2] Bohinc, T.: Grundlagen des Projektmanagements. Gabal Verlag, Offenbach (2010)
[3] Enzyklopädie, B.: 19. Auflage, 19. Band. F.A. Brockhaus, Mannheim (1992)
[4] http://de.wikipedia.org/wiki/Kollektive_Intelligenz

Understanding and Control of Power Grids

Zhong Li and Yan Li

Abstract. What happened in 2011 should be remembered by people all over the world and recorded in the human history? Definitely are two events among many others. One is the Japan's most powerful earthquake on March 11 2011 since records began, which has struck the north-east coast, triggering a massive tsunami, and further arising the Fukushima Daiichi nuclear disaster; and a consequent event is that on May 30, 2011 German chancellor Angela Merkel announced that Germany would halt construction of new nuclear power reactors and would completely phase out all nuclear power in Germany by 2022. Germany has thus become the first one among the developed industrial countries to ban nuclear power. To supply this gap of energy, we will have to seek new green energy, like wind power and solar energy; and on the other hand, to build more efficient and safer power grids, which is currently the main concern of governments and scientists and also of this paper by giving a review on this research area, including pointing out the problems in power grids, research topics, and possible resolutions.

This paper is dedicated to the 60th birthday of Prof. Wolfgang Halang, an advocator of new green energy.

1 Introduction

Energy sources have become an everlasting topic for scientists, governments, and common people. As demand for electricity soars, the pollution produced from fossil

Zhong Li
Faculty of Mathematics and Computer Science, FernUniversität in Hagen,
58084 Hagen, Germany
e-mail: `zhong.li@fernuni-hagen.de`

Yan Li
College of Engineering, South China Agricultural University,
510642 Guangzhou, China
e-mail: `homlesly@yahoo.cn`

fuel-burning plants is heading towards dangerous levels. Coal, gas and oil burning power plants are already responsible for half of America's air pollution, for example. The world's reserves of fossil fuels are running out, and according to estimates, fossil fuels will be burned up within fifty years [1].

Nuclear power was considered as a solution to the energy problem, because it generates a large amount of energy, is very cost effective, and does not contribute to air pollution. Nowadays, nuclear power provides about 6% of the world's energy and 13-14% of the world's electricity [2]. However, what happened in Japan on March 11 2011, namely, a very powerful earthquake and tsunami, as well consequently, the Fukushima nuclear disaster, has prompted a rethink of nuclear energy policy worldwide, followed by Germany's decision to close all its reactors by 2022, and Italy's vote to ban nuclear power for decades. Thus, Germany becomes the first one among the developed industrial countries to ban nuclear power.

To fill up such a big gap of electricity and energy, we have to seek clean, safe, renewable energy sources, such as wind, solar, small hydro, modern biomass, geothermal, and biofuels. In 2008, about 19% of global final energy consumption came from renewables, with 13% coming from traditional biomass, which is mainly used for heating, and 3.2% from hydroelectricity [3]. Wind power is growing at the rate of 30% annually, with a worldwide installed capacity of 158 gigawatts (GW) in 2009, and is widely used in Europe, Asia, and the United States [29]. As one of the world's top photovoltaics (PV) installers, Germany installed 7,400 MW from nearly one-quarter million individual systems in 2010, and solar PV provided 12 TWh (billion kilowatt-hours) of electricity in 2010, about 2% of total electricity [17]. Some market analysts expect that this could reach 25% by 2050 [4]. However, wind power has also some problems, which are shared by solar energy. Because of the intermittency and variability of wind, conventional power plants must be kept running at full capacity to meet the actual demand for electricity. Most cannot simply be turned on and off as the wind dies and rises, and the quick ramping up and down of those would actually increase their output of pollution and carbon dioxide [30]. Moreover, wind power grids as local networks should be integrated to the main power grids, leading to some problems with collection and transmission network, generator characteristics and stability (low voltage ride through (LVRT)), capacity factor, penetration, intermittency and penetration limits, etc., to be addressed. Fortunately, new theoretical results and findings of recently quickly developed complex networks theory may help understanding the structure, dynamics, and emergent behavior of power grids as a whole [32, 28, 31].

In this paper, we first discuss the problems and possible solutions of wind power being integrated to power grids, which apply also to other new energy sources, like solar. Then, power grids as a whole are to be discussed from the perspective of complex networked systems, including:

- understanding the structure of power grids,
- understanding dynamics of power grids,
- predictive modeling and simulation for power grids, and
- design and control of power grids.

2   Wind Power Grids Integration

Wind power is undergoing the fastest growth in the world. It promises a clean and renewable source of electricity, and reduces the dependence on fossil fuels and the output of greenhouse gases and other pollution. However, electricity generation from wind fluctuates greatly, requiring additional reserves of conventional capacity to compensate; high-demand periods of cold and heat correspond to periods of low wind; forecasting for wind power is limited; wind power needs an expansion of the (extra-)high-voltage grid infrastructure; and expansion of wind power makes the grid more unstable [30].

On May 5, 2011 China's State Electricity Regulatory Commission(SERC) reported two incidents of wind turbines, which were disconnected from the power grid because of voltage dips. The incidents occurred on April 17, 2011 in Gansu and Hebei provinces in China, according to the announcement. On April 17, 702 wind turbines were disconnected from the power grid in Jiuquan city, Gansu province, China. The disconnection was caused by breakdowns in two box transformers at a windfarm in Jiuquan. The accident caused a 54 percent decrease in wind power output. On the same day, 644 wind turbines were disconnected from the power grid in Zhangjiakou, Hebei Province, China. This disconnection was also caused by a box transformer malfunction, and resulted in a 48.5% decrease in wind power output. Besides, in late March, the SERC announced that 598 wind turbines were disconnected from Jiuquan's power grid on Feb. 24 2011 for similar reasons. The SERC said that the three incidents have exposed major problems in China's booming wind power industry, particularly the absence of low voltage ride through (LVRT) capability in China's wind turbines [5].

According to Wikipedia, in electricity supply and generation, LVRT is what an electric device, especially wind generator, may be required to be capable of functioning as following requirements when the voltage in the grid is temporarily reduced due to a fault or load change in the grid. Depending on the application the device may, during and after the dip, be required to

- disconnect temporarily from the grid, but reconnect and continue operation after the dip,
- stay operational and not disconnect from the grid,
- stay connected and support the grid with reactive power, which means the reactive current of the positive sequence of the fundamental.

For generating units such as wind turbines and solar power stations, the required LVRT behavior is defined in grid codes issued by the grid operator. Examples of the such grid codes are the German E.On grid code and UK National Grid code [20, 21, 23, 19, 18].

To overcome the LVRT problem, modern wind power generation technology focuses on using variable speed operation, which is achieved by electrical converters [25]. Examples of variable speed wind turbine generators are doubly fed induction generators (DFIGs) and permanent magnet synchronous generators (PMSGs) with primary converters.

As a result of large-scale wind power generation, interconnectiong large wind farms to power grids and the relevant influences on the host grids become a major concern. Fig. 1 shows the typical configuration for distributed resources (DR), like wind and solar, connected with electric power systems [26, 20]. Wind farms are required to comply with stringent connection requirements: reactive power support, transient recovery, system stability and voltage/frequency regulation.
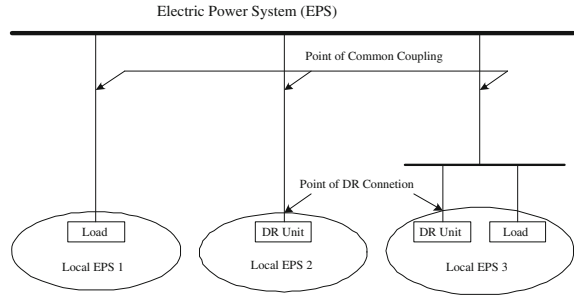
Fig. 1 Distributed resources connected to a power system

Two of the strongest challenges to wind power's future prospects are the problems of intermittency and grid reliability. The intermittent profile can affect grid reliability and is a new dimension that transmission and distribution network operators have not traditionally had to manage on any significant scale. Technical challenges and opportunities include the role and future architecture of networks, control, data exchange and provision of security services across the transmission/distribution network boundary, quality of supply, reliability and network resilience, together with issues such as dynamic stability and intermittency seen at a system level.

Research on technology, tools and practices for the integration of wind power into electricity systems has lagged its explosive growth. The issues related to grid integration have gained more attention in the last few years as large amounts of wind power have been developed and ambitious targets or obligations for renewable energy for electricity supply have been adopted at regional, national and state levels.

In general, research topics concerned with wind power and distributed generation integration into electricity supply systems can be categorized into four main areas [22]:

1. **Improving forecasting for electricity production** The intermittent nature of wind power and its non-dispatchability present relatively new challenges to grid system operators who must ensure that supply and demand are always in balance. Better forecasts for wind power production potential mean fewer network system imbalances.
2. **Modeling and grid simulation studies to develop management tools and practices to ensure grid system optimization**
3. **Investigations and planning of designs to reinforce and extend the grid** In some countries wind turbines have been developed in rural areas necessitating transmission to areas with higher demand. As well, the oldest and thus smallest

turbines that are located at the favorable wind sites are being replaced by new ones with ten times the capacity, which can put strain on existing infrastructure. Off-shore wind developments will be large and require high voltage transmission capacity.

4. **Analysis and development of grid access rules, technical code requirements and international standards** In the early years of wind power development the technical aspects of grid connection were largely a matter of local concern with utilities and grid operators. With growth in wind power, grid operators become more concerned with wind turbine effects on grid safety and reliability. Currently most grid operators have their own technical requirements for interconnection. This imposes additional costs on turbine manufacturers as they have to adapt their products to local requirements. The need for certified testing procedures and additional international standards need to be addressed.

3 Power Grids as Complex Networked Systems

A power grid is an interconnected network for delivering electricity from suppliers to consumers, which may support all or some of the following four distinct operations:

- Electricity generation
- Electric power transmission
- Electricity distribution
- Electricity control

A power grid comprises transmission lines and several substations that include generators, i.e. the electricity power source; transmission substations, which connect high-voltage transmission lines; and distribution substations, which deliver the electricity to consumers [7]. Generators are the sources for power, transmission substations transfer the power among high voltage transmission lines, and distribution substations are at the outer edge of the transmission grid, and the centers of local distribution grids.

Some cascading failures and blackouts have been reported. One of the most serious was the Northeast Blackout that affected 50 million people in the USA and Canada on 14 August 2003, and resulted in a huge loss of money (around US $30 billion) [9].

The power grid system is the core of all critical infrastructures, including energy, communications, transportation, water, and food supply. Vulnerabilities of the power grids system are well documented. Power grids should have the capabilities for resilience, restoration, and recovery.

How can we describe such complex power grids and understand their emergent behavior? Because real power grid can not be used for testing and validation, computational modeling and simulation provides a safe and cost-effective alternative that can help enormously in developing needed understanding. In this context,

a model is a simplified representation of a power grid at a particular location or point in time for understanding the real system. Modeling interdependencies of infrastructure accompanying the power grid, including sensors, control, communication network and computational components using software simulation is challenge. A simulation is the manipulation of a model to study the grid's changing behavior over time or space, thus enabling an analyst to perceive behavior that would otherwise be apparent. Modeling and simulation, then, represent a discipline for understanding the interaction of the parts of an electric grid and of the system as a whole. They are an integral part of the management, planning, and stewardship of the grid system.

3.1 Modeling and Simulation of Power Grids

The power grid can be modeled as a linear network of RLC (Resistor, Inductor, Capacitor) elements excited by constant voltage sources and time varying current sources. It can be expressed by an ordinary differential equation following the Modified Nodal Analysis (MNA) formulation [14, 13]:

$$Gx + C\dot{x} = u(t), \tag{1}$$

where x is a vector of node voltages, and source and inductor currents; $G \in \mathbf{R}^{m \times m}$ is the conductance matrix, which depends on the topology of the grid, with m denoting the number of nodes in the power grid; C includes the capacitance and inductance terms, and $u(t)$ denotes the time varying sources modeling the sources and drains.

These models can often be described by systems of partial differential equations, and sophisticated methodologies have been developed to translate the physics into mathematical and computational models that can be analyzed to provide predictive understanding of their behavior.

3.1.1 Understanding Power Grids Structure

To understand the overall behavior of power grids, we need to know the underlying topology of the power grids. Power grids are normally of sufficient scale and complexity so that it is impossible to measure their structures. In stead, statistical inference and machine learning methods can be helpful to indirectly infer the power grids topologies from a limited set of noisy observations or data, which can be only indirectly connected to the basic network structures and interconnections [10].

For a power network of 314,123 nodes, Chassin and Posse considered any vertex regardless of voltage (note that the size of the power networks used in the other studies did not take into account nodes of small voltages), and showed that the network has a radial form. The generators are at the center, with transmission substations at the middle and the distribution substations at the border, in the so-called "bow-tie" configuration [11].

3.1.2 Understanding Power Grids Dynamics

The structure of power grids serves as the substrate for dynamic behavior under verying conditions, e.g., at steady state and as the network evolves, but particularly large-scale emergent behavior changes due to changes or events in the system or its environment [31, 27].

Graph models have been extensively used to model network structure, which can also apply to power grids. The mathematical structures should be able to be extended to incorporate dynamics and scale as the networks grow. Emergent behavior is one of the hallmarks of dynamics of complex networked systems including power grids. Using the mathematical models, we can capture the nonlinear interactions, which underline the emergent behavior. It is also required to develop stability conditions and understand the critical conditions, under which phase transitions happen [24]. Furthermore, it is necessary to provide theoretical bounds on reconstructability of power grids and estimates of confidence in model structure and parameters for understanding the foundation of power grids.

3.1.3 Mathematical Modeling and Simulation

Given a mathematical or statistical model of a power grid, simulations can be developed to realize the model structure and produce dynamic network behavior. Multi-scale information system simulations will be conducted, by simulating the real power grids at highest resolution levels and analyzing with sophisticated mathematical models at lower levels. The ultimate goal of modeling and simulation is to develop methods for understanding and controlling events and anomalies in power grid systems.

Recent studies have revealed that a large fraction of complex networks are highly heterogeneous with respect to the node degree. In these networks the majority of the nodes have low degrees, but there is a continuous hierarchy of high-degree nodes (hubs) that play an important role in the system. The degree distribution of these networks follows a power law $P(k) \sim k$ with the exponent mostly between 2 and 3. It was demonstrated both numerically and analytically that these so-called scale-free networks are resilient to the random loss of nodes, but are vulnerable to attacks targeting the high-degree hubs [8, 15, 16, 12]. Therefore it is important both from a theoretical and practical viewpoint to determine whether the connectivity of the power grid is reliant on a small set of hubs and whether their loss will cause a large-scale breakdown of the power grid's transmission capability.

A fundamental requirement of the power grid is robustness, the ability to withstand and tolerate errors (random failure) and targeted attacks [7]. To ensure the reliability of power distribution, the transmission grid was conceived in such a way that there is more than one electrical path between any two points in the system. When one generator fails, the others provide the additional power to supply the whole network. Due to the redundant paths, the power grids are robust to random failures of generators, since the others can handle the additional power, but can have large blackouts if highly connected transmission substations fail as the other substations cannot take the additional overload. A possible measure of network redundancy is

the so-called edge range, defined as the distance between the two endpoints of an edge if the edge connecting them were removed. It is found that parallel edges and short alternative paths are fairly frequent. However, around 15% of the edges in the power grid have an infinite range.

The connectedness of the power grid ensures the transmission of power over large distances, thus, it implies that local disturbances propagate over the whole power grid. The failure of a power line due to lightning strike or short-circuit leads to the overloading of parallel and nearby lines. Power lines are guarded by automatic devices that take them out of service when the voltage on them is too high. Generating substations are designed to switch off if their power cannot be transmitted; however, this protective measure has the unwanted effect of diminishing power for all consumers. Another possible consequence of power line failure is the incapacitation of transmission substations, possibly causing that the power from generators cannot reach distribution substations and ultimately consumers [7].

3.2   Towards Smart Grids

Current electric grid infrastructure is more than several decades old. Smart grid is a type of electrical grid which attempts to predict and intelligently respond to the behavior and actions of all electric power users connected to it - suppliers, consumers and those that do both - in order to efficiently deliver reliable, economic, and sustainable electricity services [6].

In Europe, the smart grid is conceived of as employing innovative products and services together with intelligent monitoring, control, communication, and self-healing technologies in order to [6]:

- better facilitate the connection and operation of generators of all sizes and technologies;
- allow consumers to play a role in optimizing the operation of the system;
- provide consumers with better information and options for choice of supply;
- significantly reduce the environmental impact of the whole electricity supply system;
- maintain or even improve the existing high levels of system reliability, quality and security of supply;
- maintain and improve the existing services efficiently.

The function of an Electrical grid is not a single entity but an aggregate of multiple networks and multiple power generation companies with multiple operators employing varying levels of communication and coordination, most of which is manually controlled. Smart grids increase the connectivity, automation and coordination between these suppliers, consumers and networks that perform either long distance transmission or local distribution tasks.

4   Conclusions

This paper reviews partially the problems, challenges, possible solutions, and research topics in power grids, with the aim to attract the attention to use the new results, findings and methodologies from the theory of complex networked systems.

All discussed above are aiming to answer the following questions:

- Can the structure of a power grid be reconstructed from necessarily limited observations? There are critical interactions among network elements arising unpredictably emergent behavior.
- How do dynamic processes on power grids evolve? Interactions between network elements on multiple space and time scales can lead to emergent behavior.
- Can simulation methods be scaled to performance levels allowing the use in design and control optimization for large-scale power grids?

References

1. http://en.wikipedia.org/wiki/Fossil_fuel
2. http://en.wikipedia.org/wiki/Nuclear_power
3. http://en.wikipedia.org/wiki/Energy_development
4. http://www.worldwatch.org/node/5449
5. http://www.21cbh.com/HTML/2011-5-6/4NMDAwMDIzNjk4Ng.html
6. http://www.smartgrids.eu/?q=node/163
7. Albert, R., Albert, I., Nakarado, G.: Structural vulnerability of the North American power grid. Phys. Rev. E 69(2), 25103 (2004)
8. Albert, R., Jeong, H., Barabási, A.-L.: Error and attack tolerance of complex networks. Nature 406, 378 (2000)
9. Bai, W., Zhou, T., Fu, Z., Chen, Y., Wu, X., Wang, B.: Electric power grids and blackouts in perspective of complex networks. In: Proceedings of the International Conference on Communications, Circuits and Systems, vol. 4 (2006)
10. Brase, J., Brown, D.: Modeling, Simulation and Analysis of Complex Networked Systems. A Program Plan (May 2009)
11. Chassin, D., Posse, C.: Evaluating North American electric grid reliability using the Barabasi-Albert network model. Physica A 355(2-4), 667–677 (2005)
12. Callaway, D.S., Newman, M.E.J., Strogatz, S.H., Watts, D.J.: Network robustness and fragility: percolation on random graphs. Phys. Rev. Lett. 85, 5468 (2000)
13. Chen, H.H., Neely, J.S.: Interconnect and circuit modeling techniques for full-chip power supply noise analysis. IEEE Transactions on Components, Packaging, and Manufacturing Technology, Part B: Advanced Packaging 21(3), 209–215 (1998)
14. Chua, L.O., Lin, P.: Computer-Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques. Prentice Hall Professional Technical Reference, Eaglewood Cliffs, NJ, USA (1975)
15. Cohen, R., Erez, K., ben-Avraham, D., Havlin, S.: Resilience of the internet to random breakdowns. Phys. Rev. Lett. 85, 4626 (2000)
16. Cohen, R., Erez, K., ben-Avraham, D., Havlin, S.: Breakdown of the Internet under intentional attack. Phys. Rev. Lett. 86, 3682 (2001)
17. Gipe, C.: New Record for German Renewable Energy in 2010. Renewable Energy World (March 25, 2011)

18. Gipe, P.: Wind Power. Chelsea Green Publishing Company. Post Mills, Vermount, USA (1995)
19. Ghosh, A., Ledwich, G.: Power Quality Enhancement using Custom Power Devices. Kulwer Academic (2002)
20. IEEE Recommended Practices on Monitoring Electric Power Quality. IEEE Std. 1159 (1995)
21. IEEE 1547 Standard for Interconnecting Distributed Resources with Electric Power Systems (2007)
22. International Energy Technology Collaboration and Climate Change Mitigation. COM/ENV/EPOC/IEA/SLT. 5 (2005)
23. Jenkins, N., Allan, R., Crossley, P., Kirschan, D., Strbac, G.: Embedded Generation. IEEE Power and Energy Series (31) (2000)
24. Liu, Y., Slotine, J., Barabasi, A.: Controllability of complex networks. Nature 473, 167–173 (2011)
25. Luna, A., Rodriguez, P., Teodorescu, R., Blaabjerg, F.: Low voltage ride through strategies for SCIG wind turbines in distributed power generation system. In: Power Electronics Specialists Conference, June 15-19, pp. 2333–2339 (2008)
26. Mittal, R., Sandhu, K.S., Jain, D.K.: Low voltage ride-through (LVRT) of grid interfaced wind driven PMSG. ARPN Journal of Engineering and Applied Science 4, 73–83 (2009)
27. Motter, A., Matias, M., Kurths, J., Ott, E.: Dynamics on complex networks and applications. Physica D 224, vii (2006)
28. Newman, M.: The structure and function of complex networks. SIAM Review 45, 167–256 (2003)
29. REN21: Renewables Global Status Report 9 (2009)
30. Rosenbloom, E.: `http://docs.wind-watch.org/ProblemWithWind.pdf`
31. Strogatz, S.: Exploring complex networks. Nature 410, 268 (2001)
32. Watts, D., Strogatz, S.: Collective dynamics of 'small-world' networks. Nature 393, 440–442 (1998)

Proving the Safety of Autonomous Systems with Formal Methods - What Can You Expect?

Theodor Tempelmeier

Abstract. This contribution briefly recapitulates the notions of autonomous systems and formal methods and clarifies their meaning as used in the following. Two examples of possible fallacies with formal syntax and semantics are given, but irrespectively of that, a perfect formal method is assumed for the rest of the paper. In the main part three examples are given, where even with a perfect formal proof of certain safety aspects, safety may nevertheless be compromised. The reasons for this are environmental influence, unaccounted world knowledge, and misbehaviour of neighbour systems. As conclusion, however, the use of formal methods is not discouraged at all, but awareness of the limitations of formal methods is requested from everybody.

1 Introduction

Autonomous systems may operate in an environment shared with human beings. In this case, safety of the autonomous system is an important concern. The autonomous system must not impose a danger to the health or even to the lives of human beings. Formal methods are a way to reason about the system and prove certain aspects, which are necessary to ensure safety of the system.

2 Autonomous Systems

First the notion of "autonomous systems" shall be clarified. Most definitions of the term "system" include people to be part of the system [1, 2]. Thus, a conventional car with a human driver would be an autonomous system in view of these definitions. However, such systems usually are not meant with the term autonomous system. So, probably a more precise definition is necessary, restricting the

Theodor Tempelmeier
University of Applied Sciences, Hochschulstr.
1, 83024 Rosenheim, Germany
e-mail: `tempelmeier@fh-rosenheim.de`

term to (sub-)systems which are controlled solely by computers or equivalent mechanisms, e.g. programmable logic devices. Thus, only fully automated autonomous systems are considered for now.

3   Formal Methods

Another clarification seems necessary for the term formal methods as understood in this contribution.

3.1   *Formal Syntax and Semantics*

Clearly, a notation with a formal syntax and with formal semantics is a prerequisite for a formal method. But even this seemingly trivial statement needs a few comments concerning possible fallacies.

For instance, a formal syntax is not beneficial per se. One can see this when looking at machine code, the syntax of which may be perfectly formalized–it is just a sequence of 32-bit words. But this kind of formal notation is unacceptable to human reasoning. Formal notations which are hard to understand for humans are only acceptable as intermediate forms to be processed automatically. So readability of the formal notation is a concern when the notation is to be used by humans.

As for semantics, taking the Unified Modeling Language (UML) as an example, the UML has been notoriously criticized for its lack of semantics. But with UML 2 it is said that several profiles have been adopted …

"… with the ability to specify accurately the qualitative and quantitative characteristics of software and related systems. The most recent of these is called the 'UML Profile for Modeling and Analysis of Real-Time and Embedded Systems (MARTE)'" [3].

However, looking at MARTE [4], in order to clarify, say for instance the semantics of an event (called "NotificationRessource" in MARTE, see figure 1) one can still find disappointing "definitions" of semantics such as the following.
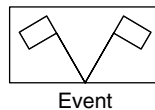
Event

Fig. 1 Icon for an Event (a "NotificationRessource") in MARTE

"Semantics: NotificationResource supports control flow by notifying the occurrences of conditions to awaiting concurrent resources, such as POSIX Signal, OSEK\VDX Event, ARINC-653 Event… Occurrences can be memorized (i.e., memorized in a buffer), bounded (i.e., each occurrence increments a counter) or memoryless (i.e., not memorized in a buffer, hence multiple occurrences are lost)." [4]

In the opinion of the author, such definitions are too simple and too vague to reason about safety-critical systems. Further, semantics is deferred to Posix, OSEK, ARINC, or whatsoever. Thus the criticism of weak semantics in the UML is still valid.

3.2 Formal Proof

In addition to a formal syntax and semantics a formal proof as a third pillar of a formal method has to be done to show that (some) aspects of system safety can be assured.

For this contribution it is simply assumed that there is a perfect formal method with adequate formal syntax and semantics and with a correct proof.

4 Limitations of Formal Methods

Unfortunately, even with a perfect formal proof of (some) safety aspects, there is still no full guarantee that safety is not compromised. The following three examples will show the problems.

4.1 Environmental Influence

In the first example, a formal proof concerning the safety of an Automated Guided Vehicle (AGV) will be investigated [5].[1]

In this contribution it is proven that the vehicle in the AGV system will come to a halt within its prescribed limits, depending on speed, trajectory, etc. This is achieved by an emergency stop and a "defined deceleration".

This proof itself is not called into question here.

However, it is obvious that the mentioned proof does not cover environmental influences such as an oil spill–not uncommon in a factory–, changing the defined deceleration or making deceleration impossible altogether.

To avoid safety problems like this all possible environmental influence would have to be included in the formal proof.

4.2 Unaccounted World Knowledge

In another (fictitious) example, an airplane is assumed to have a software component which prevents reverse thrust while the airplane is not on the ground. Further, a formal proof concerning this aspect is assumed to have been done.

---

[1] By coincidence, the subject is related to the author's personal past experience in developing an AGV system [6], in getting some insight into safety-critical systems [7] and in working on formal methods [8].

The input assumption to the proof shall have been that the condition "airplane is not on the ground" was seen as equivalent to the condition "wheels (of the landing gear) do not rotate". However, as is known, hydroplaning may lead to a situation where the wheels do not rotate but skid over the water. So the airplane may be on the ground, but its wheels do not rotate, and reverse thrust should be enabled but is disabled, with possibly disastrous consequences.[2] The knowledge about hydroplaning obviously was not accounted for in this fictitious proof.

To avoid safety problems like this all world knowledge about all aspects related somehow to the system under investigation would have to be included in the formal proof.

4.3 Unexpected Behaviour of Neighbour Systems

In the last example a system of (sub-)systems is considered (figure 2a) consisting of two airplanes A and B and an air traffic controller C. The airplanes may be seen as a combination of the human pilot flying and the Traffic Alert and Collision Avoidance System (TCAS). TCAS is only giving recommendations on how to avoid a collision, so-called resolution advisories (RA), to the pilot. In the case of a flight path possibly leading to a collision (fig. 2b) resolution advisories are given to avoid the collision (fig. 2c).

As only the externally observable behaviour of the systems A and B matters (figures 2a, b, c), it is irrelevant, whether the systems are built internally from two systems, one of which is the pilot. One could, equivalently to the system formed by Pilot $\cup$ TCAS, figure out a fully automatic system with equivalent behaviour. Such a system would then actively engage the controls instead of relying on the pilot to do this. For the following it is assumed that system B fictitiously has a fully automatic TCAS[3], formally proven to operate correctly.

Now assume that in the system of systems, it is system A which shows an erroneous behaviour (fig. 3d). Again, it is irrelevant whether the system Pilot $\cup$ TCAS or the system A $\cup$ C or a fictitiously fully automatic TCAS is at fault.

An in-air collision would result from the erroneous behaviour of system A.[4]

To avoid accidents like this the TCAS could be enhanced to react appropriately by reversing its resolution advisory for system B (figure 2e)[5]. One might then again prove that the system is safe.

---

[2] The reader might recognize some resemblance with the so-called Warsaw accident. But it is well known, that there were other and different influences with this accident [9].

[3] TCAS does not engage the controls but only recommends action to the pilot. Note however that in the automotive industry, there already exist driver assist systems which actively and autonomously engage the controls.

[4] This happened in the Überlingen accident [10] resulting in a loss of 71 lives.

[5] Such a reversal resolution advisory probably would have prevented the Überlingen accident, had it already been implemented in the TCAS version in use. [10]
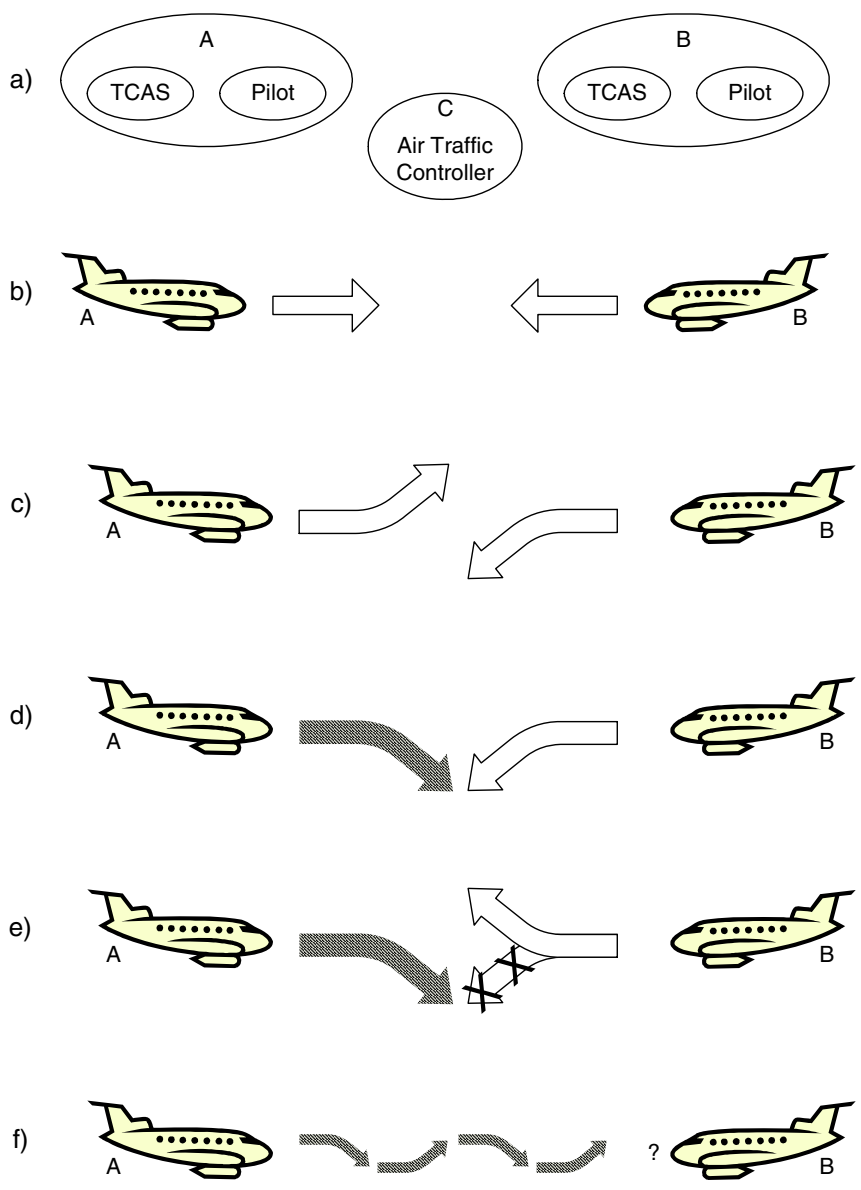
Fig. 2 A System of (Sub-)Systems.
a) Two airplanes and an air traffic controller seen as a system of (sub-)systems.. **b)** Two airplanes on a collision course. **c)** Correct escape manoeuvres. **d)** Erroneous behaviour of system A. **e)** Compensating the erroneous behaviour of a system by reverse action. **f)** Fictitiously compromising safety again by an unforeseen oscillating erroneous behaviour of system A

But again one could possibly construe a situation, where the system would be unsafe again, for instance, maybe, when the misbehaving plane would follow an oscillating path pattern (figure 2f). Such behaviour need not necessarily be the result of evil-minded human action, but might be caused by some other defects, e.g. by so-called pilot induced oscillations.

To avoid safety problems like this all conceivable misbehaviour of all neighbour systems would have to be foreseen and included in the proof.

5 Conclusion

Various limitations of formal methods have been shown. The reason for the described shortcomings is in summary that it is practically impossible to include all potential external effects on the system in the formal proof.

However, this contribution must not be seen as a case against formal methods. To the contrary, the author advises to use formal methods to the utmost extent in safety critical systems. However, everybody involved must be aware of the limitations of formal methods. As Albert Einstein put it

„Insofern sich die Sätze der Mathematik auf die Wirklichkeit beziehen, sind sie nicht sicher, und insofern sie sicher sind, beziehen sie sich nicht auf die Wirklichkeit." (As far as the theorems of mathematics relate to reality, they are not certain, and as far as they are certain, they do not relate to reality.) [11]

References

[1] ISO/IEC, IEEE, International Standard ISO/IEC 26702, IEEE Std 1220-2005, First edition 2007-07-15, Systems engineering – Application and management of the systems engineering process (2007)
[2] INCOSE, A Consensus of the INCOSE Fellows – Definition of a system (2011), http://www.incose.org/practice/fellowsconsensus.aspx (accessed June 30, 2011)
[3] Selic, B.: Counterpoint. UML 2: Designed for Architects. IEEE Software 27(6), 55–57 (2010)
[4] OMG, UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded Systems. Version 1.0. OMG Document Number: formal/2009-11-02 (2009), Standard document url, http://www.omg.org/spec/MARTE/1.0 Section F.8.15
[5] Frese, U., Hausmann, D., Lüth, C., Täubig, H., Walter, D.: Zertifizierung einer Sicherungskomponente mittels durchgängig formaler Modellierung. In: Maalej, W., Bruegge, B. (eds.) SoftwareEngineering 2008 – Workshopband. Proceedings Series of the Gesellschaft für Informatik (GI). LNI, vol. P-122, pp. 335–338 (2008)
[6] Tempelmeier, T.: Microprocessors in Factory Automation – A Case Study of an Automated Guided Vehicle System and its Integration into a Hierarchical Control Structure. In: Proceedings EUROMICRO 1986, Microprocessing and Microprogramming, Venice, September 15-18, vol. 18, pp. 647–656 (1986)

[7] Roßkopf, A., Tempelmeier, T.: Aspects of Flight Control Software – A Software Engineering Point of View. Control Engineering Practice 8, 675–680 (2000)

[8] Tempelmeier, T.: Formal Methods – An Informal Assessment. Technischer Report. Dasa MT36 SR-1775-a. Daimler-Benz Aerospace, Ottobrunn (May 1998)

[9] Main Commission Aircraft Accident Investigation, Report on the Accident to Airbus A320-211 Aircraft in Warsaw (September 14, 1993) Warsaw (March 1994), `http://www.rvs.unibielefeld.de/publications/Incidents/DOCS/ComAndRep/Warsaw/warsaw-report.html` (accessed June 30, 2011)

[10] Bundesstelle für Flugunfalluntersuchung, Untersuchungsbericht AX001-1-2/02 (May 2004), `http://www.bfu-web.de/cln_030/nn_223648/DE/Publikationen/Untersuchungsberichte/untersuchungsberichte_node.html?_nnn=true` (accessed June 30, 2011)

[11] Einstein, A.: Geometrie und Erfahrung. Festvortrag in der Preußischen Akademie der Wissenschaften am 27 (January 1921) In: Mein Weltbild. Ullstein Buch Nr. 65, Berlin (1955)

The Active Element Machine

Michael Stephen Fiske

Abstract. A new computing machine, called an active element machine (AEM), and the AEM programming language are presented. This computing model is motivated by the positive aspects of dendritic integration, inspired by biology, and traditional programming languages based on the register machine. Distinct from the traditional register machine, the fundamental computing elements – active elements – compute simultaneously. Distinct from traditional programming languages, all active element commands have an explicit reference to time. These attributes make the AEM an inherently parallel machine, enable the AEM to change its architecture (program) as it is executing its program. Using a random bit source from the environment and the Meta command, we show how to generate an AEM that represents an arbitrary real number in $[0, 1]$. Exploiting the randomness from the environment, this example is extended to an AEM that can recognize an arbitrary binary language $L \subseteq \{0, 1\}^*$. Finally, we demonstrate an AEM that finds the Ramsey number $r(3, 3)$, illustrating how parallel AEM algorithms and time in the commands help compute an NP-hard problem.

1 Introduction

We present a new computing machine called an active element machine (AEM) and the active element machine programming language. This computing model is motivated by the positive aspects of dendritic integration, inspired by biology, and traditional programming languages based on the register machine. Distinct from the traditional register machine, the fundamental computing elements – active elements – compute simultaneously. Distinct from traditional programming languages, all active element commands have an explicit reference to time. These attributes make the AEM an inherently parallel machine and enable the AEM to change its architecture (program) as it is executing its program.

Michael Stephen Fiske
Aemea Institute, San Francisco, CA, 94129
e-mail: `mf@aemea.org`

1.1 Wilfrid Rall's Models of Dendritic Integration

Wilfrid Rall's research [35] in neurophysiology influenced the development of the active element machine – in particular, his work on dendritic integration and how this contributes to computation. Rall's mathematical models are thorough and complicated; Rall modelled the non-linearities of the neuron and much of his work focussed on the dendrites.

Our goal was to capture the critical computational properties of dendritic integration that use its computational power while keeping the mathematics as simple as possible. Another goal was to assure that the mathematics and computing mechanism were simple enough to implement in silicon and other kinds of hardware ([18], [19]).

Our third goal was to make the machine and language simple enough to design autonomous systems (implicitly program) with evolutionary methods or for a person to explicitly program or both. Early and current neural network models [25], [31] are complicated to program or do not have a simple programming language for designing the network. For the above reasons, implicit and explicit programmability were important criteria that influenced the design of the active element machine.

1.2 Register Machine Computation

Another part of this development comes from the formal model of the Turing machine [45] and the subsequent von Neumann architecture. (This section contains some rhetorical content as a means to motivate new notions.) Today's computers do not conceptually work much differently than these early models. Perhaps, the biggest difference is that today's computers are much faster. In the current notion of an *algorithm*, the relevant concepts of the Turing computing model (see [45] and definitions 13, 14, 15, 16) are:

- There are finite number of alphabet symbols $A = \{a_1, \ldots, a_n\}$ read and written to a tape.
- There are a finite number of machine states $Q = \{q_1, \ldots, q_m\}$.
- The Turing program, η, is a finite set of rules that stays fixed i.e. the rules do not change as the program executes.
- The execution of one rule represents a computational step. During this computational step, one of the rules is selected, based on the current alphabet symbol pointed to by the tape head and the current machine state. The output of the rule specifies that a new alphabet symbol or the same symbol is written to the tape, the machine moves to a new state or stays in the current state and that the tape head moves one square to the left or right.
- *Computational steps are executed sequentially with no explicit reference to time.*

In light of the above, it seems natural for the Turing machine to lead to the register machine (see [1], [33], [44]). In the register machine, a program is a finite number of instructions that are executed in a linear sequence. Further, the contents of a register is changed in one computational step, which is analogous to writing a new symbol on the tape during one computational step of the Turing machine. In the register machine, there is also no explicit reference to an absolute or relative time. Furthermore, usually one register machine instruction is executed at a time, which creates a computational bottleneck.

1.3   *Explicit Representation of Time*

The register machine is a programmable machine but the program is fixed during program execution. There is also no notion of explicit time in the register machine model, only the order in which instructions are executed. Rall's research does not address programmability and has no notion of commands. His models used time, dendritic integration and adaptability of the synapses. The active element machine explicitly represents time in the machine commands which enables the following useful properties.

- Parallel algorithms can be implemented in a natural way, since each active element performs computation and all of them simultaneously compute.
- Explicit time in the active element commands enhances control over the active element machine computation because the synchronization of computation among different groups of active elements can be coordinated. This coordination helps avoid race conditions that can occur in the standard programming languages that implement concurrent processes.
- The machine can change its own architecture (program) with the Meta command while it is executing.
- The Meta command enables the active element machine's complexity to increase over time.

In [30], Edward Lee proposes using explicit time in a computing model and computing applications.

> This paper argues that to realize its full potential, the core abstractions of computing need to be rethought to incorporate essential properties of the physical systems, most particularly the passage of time. It makes a case that the solution cannot be simply overlaid on existing abstractions, ... The emphasis needs to be on repeatable behavior rather than on performance optimization.

1.4   *Summary*

Overall, we introduce the active element machine and a programming language that can be used to explicitly or implicitly program the machine. We show that any register machine can be computed by an active element machine. Using randomness

in the environment, time and the Meta command, we show how to construct an active element machine that corresponds to an arbitrary real number in $[0,1]$. Building upon this example, this AEM is extended so that it can recognize an arbitrary language $L \subseteq \{0,1\}^*$. Finally, we demonstrate an example of an active element machine that finds the Ramsey number $r(3,3)$ ([22], [36]), illustrating how parallel AEM algorithms and time in the commands compute an NP-hard problem ([10], [12], [21]).

2 Machine Architecture

An active element machine is composed of computational primitives called active elements. There are three kinds of active elements: Input, Computational and Output active elements. Input active elements process information received from the environment or another active element machine. Computational active elements receive messages from the input active elements and other computational active elements firing activity and transmit new messages to computational and output active elements. The output active elements receive messages from the input and computational active elements firing activity. The firing activity of the output active elements represents the output of the active element machine. Every active element is an active element in the sense that each one can receive and transmit messages simultaneously.

Each active element receives messages, formally called pulses, from other active elements and itself and transmits messages to other active elements and itself. If the messages received by active element, E_i, at the same time sum to a value greater than the threshold, then active element E_i fires. When an active element E_i fires, it sends messages to other active elements.

Let $\mathbb{Z}$ denote the integers. We define the extended integers as $\overline{\mathbb{Z}} = \{m + kdT : m, k \in \mathbb{Z}$ and dT is a fixed infinitesimal$\}$. For more on infinitesimals, see [38].

Definition 1. *Machine Architecture*
Γ, Ω, and Δ are index sets that index the input, computational, and output active elements, respectively. Depending on the machine architecture, the intersections $\Gamma \cap \Omega$ and $\Omega \cap \Delta$ can be empty or non-empty. A machine architecture, denoted as $\mathcal{M}(\mathcal{I}, \mathcal{E}, \mathcal{O})$, consists of a collection of input active elements, denoted as $\mathcal{I} = \{E_i : i \in \Gamma\}$; a collection of computational active elements $\mathcal{E} = \{E_i : i \in \Omega\}$; and a collection of output active elements $\mathcal{O} = \{E_i : i \in \Delta\}$. Each computational and output active element, E_i, has the following components and properties:

- A threshold θ_i

- A refractory period r_i where $r_i > 0$.

- A collection of pulse amplitudes $\{A_{ki} : k \in \Gamma \cup \Omega\}$.

- A collection of transmission times $\{\tau_{ki} : k \in \Gamma \cup \Omega\}$, where $\tau_{ki} > 0$ for all $k \in \Gamma \cup \Omega$.

- A function of time, $\Psi_i(t)$, representing the time active element E_i last fired. $\Psi_i(t) = \sup\{s : s < t \text{ and } g_i(s) = 1\}$, where $g_i(s)$ is the output function of active element E_i and is defined below. The sup is the least upper bound.

- A binary output function, $g_i(t)$, representing whether active element E_i fires at time t. The value of $g_i(t) = 1$ if $\sum A_{ki}(t) > \theta_i$ where the sum ranges over all $k \in \Gamma \cup \Omega$ and $t \geq \Psi_i(t) + r_i$. In all other cases, $g_i(t) = 0$. For example, $g_i(t) = 0$, if $t < \Psi_i(t) + r_i$.

- A set of firing times of active element E_k within active element E_i's integrating window, $W_{ki}(t) = \{s : \text{active element } E_k \text{ fired at time } s \text{ and } 0 \leq t - s - \tau_{ki} < \omega_{ki}\}$. Let $|W_{ki}(t)|$ denote the number of elements in the set $W_{ki}(t)$. If $W_{ki}(t) = \emptyset$, then $|W_{ki}(t)| = 0$.

- A collection of input functions, $\{\phi_{ki} : k \in \Gamma \cup \Omega\}$, each a function of time, and each representing pulses coming from computational active elements, and input active elements. The value of the input function is computed as $\phi_{ki}(t) = |W_{ki}(t)|A_{ki}(t)$.

- The refractory periods, transmission times and pulse widths are positive integers; and pulse amplitudes and thresholds are integers. The time t – that these parameters are a function of i.e. $\theta_i(t), r_i(t), A_{ki}(t), \omega_{ki}(t), \tau_{ki}(t)$ – is an element of the extended integers $\overline{\mathbb{Z}}$.

Input active elements that are not computational active elements have the same characteristics as computational active elements, except they have no inputs ϕ_{ki} coming from active elements in this machine. In other words, they don't receive pulses from active elements in this machine. Input active elements are assumed to be externally firable. An external source such as the environment or an output active element from another distinct machine $\mathcal{M}(\mathcal{I}', \mathcal{E}', \mathcal{O}')$ can cause an input active element to fire. The input active element can fire at any time as long as the current time minus the time the input active element last fired is greater than or equal to the input active element's refractory period.

An active element, E_i, can be an input active element and a computational active element. Similarly, an active element can be an output active element and a computational active element. Alternatively, when an output active element, E_i, is not a computational active element, where $i \in \Delta - \Omega$, then E_i does not send pulses to active elements in this machine.

Example 1. Overlapping Pulses with Different Firing Times
    Consider the four element machine where X, Y, and Z are input active elements and B is a computational active element. The parameters of the elements and their connections are shown in tables 1 and 2.
Input elements X, Y, and Z are externally fired at times 4, 1 and 2, respectively. At time $t = 3$, pulses created by Y and Z are travelling to B but have not yet arrived. B does not fire. At time $t = 4$, pulses created by Y and Z arrive at B. The input to B is

Table 1 Element Parameter Values

Element	Threshold	Refractory	Firing Times
X		1	4
Y		1	1
Z		1	2
B	10	2	5

No thresholds are shown for X, Y, and Z since they are input elements.

Table 2 Connection Parameter Values

Connection	From	To	Amplitude	Width	Transmission Time
XB	X	B	3	4	1
YB	Y	B	4	2	3
ZB	Z	B	4	2	2

$A_{YB}(4) + A_{ZB}(4) = 8$ which does not exceed B's threshold. B does not fire. At time $t = 5$, pulses created by Y and Z are still at B because their pulse widths are 2. Also the pulse from X arrives. The input to B is $A_{XB}(5) + A_{YB}(5) + A_{ZB}(5) = 11$ which exceeds B's threshold. B fires at time $t = 5$. At time $t = 7$, the refractory period of B has expired. The pulses created by Y and Z have passed through B. The pulse from X is still at B. The input to B is $A_{XB}(7) = 3$ which does not exceed B's threshold. B does not fire at time $t = 7$.

We intuitively summarize the machine architecture. If $g_i(s) = 1$, this means active element E_i *fired* at time s. The *refractory period*, r_i, is the amount of time that must elapse after active element E_i just fired before E_i can fire again. The *transmission time*, τ_{ki}, is the amount of time it takes for active element E_i to find out that active element E_k has fired. The *pulse amplitude*, A_{ki}, represents the strength of the pulse that active element E_k transmits to active element E_i after active element E_k has fired. After this pulse reaches E_i, the *pulse width* ω_{ki} represents how long the pulse lasts as input to active element E_i. At time s, the *connection* from E_k to E_i represents the triplet $(A_{ki}(s), \omega_{ki}(s), \tau_{ki}(s))$. If $A_{ki} = 0$, then there is no connection from active element E_k to active element E_i.

3 Active Element Machine Programming Language

In this section, we show how to explicitly program an active element machine and how to change the machine architecture as program execution proceeds. It is helpful to define a programming language, influenced by S-expressions. There are five types of commands: `Element`, `Connection`, `Fire`, `Program` and `Meta`.

Definition 2. *AEM Program*

In Backus-Naur form, an AEM program is defined as follows.

```
<AEM_program> ::= <cmd_sequence>

<cmd_sequence> ::= "" | <AEM_cmd><cmd_sequence>
                   | <program_def><cmd_sequence>

<AEM_cmd> ::= <element_cmd> | <fire_cmd> | <meta_cmd>
             | <cnct_cmd> | <program_cmd>
```

Definition 3. *AEM Symbols and Extended Integer Expressions*

In Backus-Naur form, the AEM symbols are defined as follows.

```
<ename> ::= "" | <int>| <symbol>

<symbol> ::= <symbol_string> | (<ename> . . . <ename>)

<symbol_string> ::= "" | <char_symbol><str_tail>

<str_tail> ::= "" | <char_symbol><str_tail> | 0<str_tail>
             | <pos_int><str_tail>

<char_symbol> ::= <letter> | <special_char>

<letter> ::= <lower_case> | <upper_case>

<lower_case> ::= a|b|c|d|e|f|g|h|i|j|k|l|m
                |n|o|p|q|r|s|t|u|v|w|x|y|z

<upper_case> ::= A|B|C|D|E|F|G|H|I|J|K|L|M
                |N|O|P|Q|R|S|T|U|V|W|X|Y|Z

<special_char> ::= "" | _
```

These rules represent the extended integers, addition and subtraction.

```
<int> ::= <pos_int> | <neg_int> | 0

<neg_int> ::= − <pos_int>

<pos_int> ::=  <non_zero><digits>

<digits> ::= <numeral> | <numeral><digits>

<non_zero> ::= 1|2|3|4|5|6|7|8|9

<numeral> ::= "" | <non_zero> | 0

<aint> ::= <aint> <math_op> <d> | <d> <math_op> <aint> | <d>

<math_op> ::= + | -

<d> ::= <int> | <symbol_string> | <infinitesimal>

<infinitesimal> ::= dT
```

Definition 4. `Element`

An `Element` command specifies the time when an active element's values are updated or created. This command has the following Backus-Naur syntax.

```
<element_cmd> ::= (Element (Time <aint>) (Name <ename>)
               (Threshold <int>)(Refractory <pos_int>)(Last <int>))
```

The keyword `Time` tags the time integer value s at which the element is created or updated. If the name symbol value is E, the keyword `Name` tags the name E of the active element. The keyword `Threshold` tags the threshold $\theta_E(s)$ assigned to E. `Refractory` tags the refractory value $r_E(s)$. The keyword `Last` tags the last time fired value $\Psi_E(s)$.

The following is an example of an element command.

```
(Element (Time 2) (Name H) (Threshold -3) (Refractory 2) (Last 0))
```

At time 2, if active element H does not exist, then it is created. Active element H has its threshold set to -3, its refractory period set to 2, and its last time fired set to 0. After time 2, active element H exists indefinitely with threshold $= -3$, refractory $=$ 2 until a new Element command whose name value H is executed at a later time; in this case, the Refractory, Threshold and Last values specified in the new command are updated.

Definition 5. `Connection`

A `Connection` command creates or updates a connection from one active element to another active element. This command has the following Backus-Naur syntax.

```
<cnct_cmd> ::= (Connection (Time <aint>)(From <ename>)(To <ename>)
                [(Amp <int>)(Width <pos_int>)(Delay <pos_int>)] )
```

The keyword `Time` tags the time value s at which the connection is created or updated. The keyword `From` tags the name F of the active element that sends a pulse with these updated values. The keyword `To` tags the name T of the active element that receives a pulse with these updated values. The keyword `Amp` tags the pulse amplitude value $A_{FT}(s)$ that is assigned to this connection. The keyword `Width` tags the pulse width value $\omega_{FT}(s)$. The keyword `Delay` tags the transmission time $\tau_{FT}(s)$.

When the AEM clock reaches time s, F and T are name values that must be the name of an element that already has been created or updated before or at time s. Not all of the connection parameters need to be specified in a connection command. If the connection does not exist beforehand and the Width and Delay values are not specified appropriately, then the amplitude is set to zero and this zero connection has no effect on the AEM computation. Observe that the connection exists indefinitely with the same parameter values until a new connection is executed at a later time between `From` element F and `To` element T.

The following is an example of a connection command.

```
(Connection (Time 2) (From C) (To L) (Amp -7) (Width 1) (Delay 3))
```

At time 2, the connection from active element C to active element L has its amplitude set to -7, its pulse width set to 1, and its transmission time set to 3.

Definition 6. Fire
The Fire command has the following Backus-Naur syntax.

```
<fire_cmd> ::= (Fire (Time <aint>) (Name <ename>) )
```

The Fire command fires the active element indicated by the Name tag at the time indicated by the Time tag. This command is primarily used to fire input active elements in order to communicate program input to the active element machine.

An example is (Fire (Time 3) (Name C)), which fires active element C at $t = 3$.

Definition 7. Program
The Program command is convenient when a sequence of commands are used repeatedly. This command combines a sequence of commands into a single command. It has the following definition syntax.

```
<program_def> ::= (Program <pname> [(Cmds <cmds>)][(Args <args>)]
                   <cmd_sequence> )

<pname> ::= <ename>

<cmds> ::= <cmd_name> | <cmd_name><cmds>

<cmd_name> ::=  Element | Connection | Fire | Meta | <pname>

<args> ::= <symbol> | <symbol><args>
```

The Program command has the following execution syntax.

```
<program_cmd> ::= (<pname> [(Cmds <cmds>)] [(Args <args_cmd>)] )

<args_cmd> ::= <ename> | <ename><args_cmd>
```

The FireN program is an example of definition syntax.

```
(Program  FireN (Args t  E)
 (Element (Time 0) (Name E)(Refractory 1)(Threshold 1)(Last 0))
 (Connection (Time 0) (From E) (To E)(Amp 2)(Width 1)(Delay 1))
 (Fire (Time 1) (Name E))
 (Connection (Time  t+1) (From E) (To E) (Amp 0))
 )
```

The execution of the command (FireN (Args 8 E1)) causes element E1 to fire 8 times at times 1, 2, 3, 4, 5, 6, 7, and 8 and then E1 stops firing at time = 9.

Definition 8. Keywords clock and dT
The keyword clock evaluates to an integer, which is the value of the current active element machine time. clock is an instance of <ename>. If the current AEM time is 5, then the command

```
(Element (Time clock) (Name clock) (Threshold 1) (Refractory 1)
                                                 (Last -1))
```

is executed as

```
(Element (Time 5) (Name 5) (Threshold 1) (Refractory 1) (Last -1))
```

Once command `(Element (Time clock) (Name clock) (Threshold 1)` `(Refractory 1) (Last -1))` is created, then at each time step this command is executed with the current time of the AEM. If this command is in the original AEM program before the clock starts at 0, then the following sequence of elements named 0, 1, 2, ... will be created.

```
(Element (Time 0) (Name 0) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 1) (Name 1) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 2) (Name 2) (Threshold 1) (Refractory 1) (Last -1))
 . . .
```

The keyword `dT` represents a positive infinitesimal amount of time. `dT > 0` and `dT` is less than every positive rational number. Similarly, `-dT < 0` and `-dT` is greater than every negative rational number. The purpose of `dT` is to prevent an inconsistency in definition 1. For example, the use of `dT` helps remove the inconsistency of a `To` element about to receive a pulse from a `From` element at the same time that the connection is removed.

Definition 9. `Meta`
     The `Meta` command causes a command to execute when an element fires within a window of time. This command has the following execution syntax.

```
<meta_cmd> ::= (Meta (Name <ename>) [<win_time>]   <AEM_cmd>)
<win_time> ::= (Window <aint> <aint>)
```

To understand the behavior of the Meta command, consider the execution of

```
(Meta (Name E) (Window l  w) (C (Args t a)))
```

where `E` is the name of the active element. The keyword `Window` tags an interval i.e. a window of time. `l` is an integer, which locates one of the boundary points of the window of time. Usually, `w` is a positive integer, so the window of time is `[l, l+w]`. If `w` is a negative integer, then the window of time is `[l+w, l]`.

The command `C` executes each time that `E` fires during the window of time, which is either `[l, l+w]` or `[l+w, l]`, depending on the sign of `w`. If the window of time is omitted, then command `C` executes at any time that element `E` fires. In other words, effectively $l = -\infty$ and $w = \infty$.

Consider the example where the `FireN` command was defined in 7.

```
(FireN (Args 8 E1))
(Meta  (Name E1) (Window 1 5) (C (Args clock  a  b)) )
```

Command `C` is executed 6 times with arguments `clock, a, b`. We say the firing of `E1` *triggers* the execution of command `C`.

In regard to the Meta command, we explain one assumption that is analogous to the Turing machine tape being unbounded as Turing program execution proceeds. (See Definitions 13 and 14.) During execution of a finite active element program, an active element can fire and due to one or more Meta commands, new elements and connections can be added to the machine. As a consequence, at any time the active element machine only has a finite number of computing elements and connections but the number of elements and connections can be unbounded as a function of time as the active element program executes.

4   Resolving Concurrent Generation of AEM Commands

We first explain how to resolve concurrency issues pertaining to two or more commands about to set parameter values of the same connection or same element at the same time. Then we address the Fire, Meta and Program commands.

For example, consider two or more connection commands, connecting the same active elements, that are generated and scheduled to execute at the same time.

```
(Connection (Time t) (From A) (To B) (Amp 2)  (Width 1) (Delay 1))
(Connection (Time t) (From A) (To B) (Amp -4) (Width 3) (Delay 7))
```

Then the simultaneous execution of these two commands can be handled by defining the outcome to be equivalent to the execution of only one connection command where the respective amplitudes, widths and transmission times are averaged.

```
(Connection (Time t) (From A) (To B) (Amp -1) (Width 2) (Delay 4))
```

In the general case, for n connection commands

```
(Connection (Time t) (From A) (To B) (Amp a1)(Width w1)(Delay s1))
(Connection (Time t) (From A) (To B) (Amp a2)(Width w2)(Delay s2))
. . .
(Connection (Time t) (From A) (To B) (Amp an)(Width wn)(Delay sn))
```

we resolve these to the execution of one connection command

```
(Connection (Time t) (From A) (To B) (Amp a) (Width w) (Delay s))
```

where a, w and s are defined based on the application. For theoretical studies of the AEM, averaging the respective amplitudes, widths and transmission times can be useful in mathematical proofs.

```
a = (a1 + a2 +  . . .  + an) / n
w = (w1 + w2 +  . . .  + wn) / n
s = (s1 + s2 +  . . .  + sn) / n
```

For some applications, when there is noisy environmental data fed to the input effectors and amplitudes, widths and transmission times are evolved and mutated ([9], [14], [17], [19], [26], [27], [29]), extremely large (in absolute value)

amplitudes, widths and transmission times can arise that skew an average function. In this context, computing the median of the amplitudes, widths and delays provides a simple method to address skewed amplitude, width and transmission time values.

```
a = median(a1, a2,  . . . , an)
w = median(w1, w2,  . . . , wn)
s = median(s1, s2,  . . . , sn)
```

Another alternative is to add the parameter values.

```
a = a1 + a2 +  . . . + an
w = w1 + w2 +  . . . + wn
s = s1 + s2 +  . . . + sn
```

Similarly, consider when two or more element commands – that all specify the same active element E – are generated and scheduled to execute at the same time.

```
(Element (Time t) (Name E)(Threshold h1)(Refractory r1)(Last s1))
(Element (Time t) (Name E)(Threshold h2)(Refractory r2)(Last s2))
 . . .
(Element (Time t) (Name E)(Threshold hn)(Refractory rn)(Last sn))
```

we resolve these to the execution of one element command,

```
(Element (Time t) (Name E) (Threshold h) (Refractory r) (Last s))
```

where h, r and s are defined based on the application. Similar to the connection command, for theoretical studies of the AEM, the threshold, refractory and last time fired values can be averaged.

```
h = (h1 + h2 + . . . + hn) / n
r = (r1 + r2 + . . . + rn) / n
s = (s1 + s2 + . . . + sn) / n
```

In autonomous applications, where evolution of parameter values occurs, the median can also help address skewed values in the element commands.

```
h = median(h1, h2, . . ., hn)
r = median(r1, r2, . . ., rn)
s = median(s1, s2, . . ., sn)
```

Another alternative is to add the parameter values.

```
h = h1 + h2 + . . . + hn
r = r1 + r2 + . . . + rn
s = s1 + s2 + . . . + sn
```

Rules 1, 2, and 3 resolve concurrency issues pertaining to the Fire, Meta and Program commands.

1. If two or more Fire commands attempt to fire element E at time t, then element E is fired just once at time t.

2. Only one Meta command can be triggered by the firing of an active element. If a new Meta command is created and it happens to be triggered by the same element E as a prior Meta command, then the old Meta command is removed and the new Meta command is triggered by element E.

3. If a Program command is called by a Meta command, then the Program's internal Element, Connection, Fire and Meta commands follow the previous concurrency rules defined. If a Program command exists within a Program command, then these rules are followed recursively on the nested Program command.

5   Copy and Nand Program Examples

We demonstrate an active element copy program and a nand program.

Example 2. Copy Program
This active element program copies an element's firing state to another element.

```
(Program copy (Args s t b a)
(Element (Time s-1)(Name b)(Threshold 1)(Refractory 1)(Last s-1))
(Connection (Time s-1)(From a) (To b)(Amp 0) (Width 0) (Delay 1))
(Connection (Time s) (From a) (To b) (Amp 2) (Width 1) (Delay 1))
(Connection (Time s) (From b) (To b) (Amp 2) (Width 1) (Delay 1))
(Connection (Time t) (From a) (To b) (Amp 0) (Width 0) (Delay 1))
)
```

When the `copy` program is called, active element b will start firing if a fired during the window of time [s, t]. Further, a connection is set up from b to b so that b will keep firing indefinitely. This enables b to *store* active element a's firing state.

Example 3. Nand Program
This active element program computes a nand circuit.

```
(Program nand (Args s x y l h)
(Element (Time s) (Name x) (Threshold 0) (Refractory 1) (Last s))
(Element (Time s) (Name y) (Threshold 0) (Refractory 1) (Last s))
(Element (Time s) (Name h) (Threshold -3)(Refractory 2) (Last s))
(Element (Time s) (Name l) (Threshold 3) (Refractory 2) (Last s))
(Connection (Time s) (From x) (To l) (Amp 2) (Width 1) (Delay 1))
(Connection (Time s) (From x) (To h) (Amp -2)(Width 1) (Delay 1))
(Connection (Time s) (From y) (To l) (Amp 2) (Width 1) (Delay 1))
(Connection (Time s) (From y) (To h) (Amp -2)(Width 1) (Delay 1))
)
```

B and C are input elements. L (represents a 0 output) and H (represents a 1 output) are output elements. The call `(nand  (Args -1 B C L H))` creates the connections between B, C, L and H. We verify all four cases where | is the Sheffer stroke.

1. At $t = 0$, active elements B and C do not fire i.e. B | C = 0|0 = 1. Since the threshold of H is -3, H fires at time $t = 1$.

2. At $t = 0$, active element B fires and C does not fire i.e. B | C = 1|0 = 1. Since one pulse with amplitude -2 reaches H at $t = 1$ just as the refractory period expires and $-2 > -3$, then H fires at time t = 1.

3. At $t = 0$, active element B does not fire and C fires i.e. B | C = 0|1 = 1. Since one pulse with amplitude -2 reaches H at $t = 1$ just as the refractory period expires and $-2 > -3$, then H fires at time t = 1

4. At $t = 0$, active elements B and C both fire i.e. B | C = 1|1 = 0. At time $t = 1$, two pulses with amplitude -2 reach H so H does not fire. At time $t = 1$, two pulses reach L, and each pulse has amplitude 2. L has threshold = 3, so L fires at $t = 1$.

6 Machine Computation

In the nand program example, we used the firing of output elements L, H to represent the low, high outputs respectively. The firing of elements was used to represent the computation of a boolean function. We formalize firing representations, machine computation and interpretation in the next set of definitions.

Definition 10. *Firing Representation*

Consider active element E_i's firing times in the interval of time $W = [t_1, t_2]$. Let s_1 be the earliest firing time of E_i lying in W, and s_n the latest firing time lying in W. Then E_i's firing sequence $F(E_i, W) = [s_1, \ldots, s_n] = \{s \in W : g_i(s) = 1\}$ is called a firing sequence of the active element E_i over the window of time W. From active elements $\{E_1, E_2, \ldots, E_n\}$, create the tuple $(F(E_1, W), F(E_2, W), \ldots, F(E_n, W))$, which is called a firing representation of the active elements $\{E_1, \ldots, E_n\}$ within the window of time W.

At the machine level of interpretation, firing representations express the input to, the computation of, and the output of an active element machine. At a more abstract level, firing representations can represent an input symbol, an output symbol, a sequence of symbols, a spatio-temporal pattern, a number, or even a sequence of program instructions.

Definition 11. *Sequence of Firing Representations*

Let $W_1, W_2, \ldots, W_n$ be a sequence of time intervals. Let $\mathscr{F}(\mathscr{E}, W_1) = (F(E_1, W_1), F(E_2, W_1), \ldots, F(E_n, W_1))$ be a firing representation of active elements $\mathscr{E} = \{E_1, E_2, \ldots, E_n\}$ over the interval W_1. In general, let $\mathscr{F}(\mathscr{E}, W_k) = (F(E_1, W_k), F(E_2, W_k), \ldots F(E_n, W_k))$ be a firing representation over the interval of time W_k. From these, a sequence of firing representations, $[\mathscr{F}(\mathscr{E}, W_1), \mathscr{F}(\mathscr{E}, W_2), \ldots, \mathscr{F}(\mathscr{E}, W_n)]$ is created.

Definition 12. *Machine Computation*

Let $[\mathscr{F}(\mathscr{E}, W_1), \mathscr{F}(\mathscr{E}, W_2), \ldots, \mathscr{F}(\mathscr{E}, W_n)]$ be a sequence of firing representations. $[\mathscr{F}(\mathscr{E}, S_1), \mathscr{F}(\mathscr{E}, S_2), \ldots, \mathscr{F}(\mathscr{E}, S_m)]$ is some other sequence of firing representations. Suppose machine architecture $\mathscr{M}(\mathscr{I}, \mathscr{E}, \mathscr{O})$ has input active elements

$\mathscr{I}$ fire with the pattern $[\mathscr{F}(\mathscr{E},S_1),\mathscr{F}(\mathscr{E},S_2),\ldots,\mathscr{F}(\mathscr{E},S_m)]$ and consequently $\mathscr{M}$'s output active elements $\mathscr{O}$ fire according to $[\mathscr{F}(\mathscr{E},W_1),\mathscr{F}(\mathscr{E},W_2),\ldots,\mathscr{F}(\mathscr{E},W_n)]$. In this case, we say the machine $\mathscr{M}$ *computes* $[\mathscr{F}(\mathscr{E},W_1),\mathscr{F}(\mathscr{E},W_2),\ldots,\mathscr{F}(\mathscr{E},W_n)]$ from $[\mathscr{F}(\mathscr{E},S_1),\mathscr{F}(\mathscr{E},S_2),\ldots,\mathscr{F}(\mathscr{E},S_m)]$.

An active element machine is an *interpretation* between two sequences of firing representations if the machine can compute the output sequence of firing representations from the input sequence of firing representations. Using the definition of machine computation, examples 2 and 3 help derive the following theorem.

Theorem 1. *A register machine with an unbounded number of registers can be constructed with an active element program.*

Proof. We sketch a proof. A finite boolean function can be constructed by composing a finite number of nand circuits. The repeated use of the copy program enables an active element machine to store an unbounded amount of state in terms of bits. A register machine can be constructed from the boolean functions and the ability to store state. An unbounded number of registers can be supported by the active element machine because the Element and Connection commands enable the program to add new Elements and Connections at any time. Thus, new registers can be added as needed during the computation of the register machine program.

Corollary 1. *Any Turing computable function can be computed by some active element machine, specified by a finite active element program.*

Proof. In [44], they show that the partial recursive functions are the same as the functions computable by their register machine model and consequently the same as the Turing computable functions (see [45], and Definitions 13, 14, 15 and 16). The corollary follows from this fact and Theorem 1.

Example 4. Randomness generates an AEM, representing a real number in $[0,1]$

Using a random process in the environment to fire or not fire one input effector $\mathtt{I}$ at each unit of time, we describe an active element program with a firing representation of an arbitrary real number in the unit interval $[0,1]$.

This example uses a random process from the environment to either fire input effector $\mathtt{I}$ or not fire $\mathtt{I}$ at time $t = n$ where n is a natural number $\{0,1,2,3,\ldots\}$. This random sequence of 0 and 1's can be generated by quantum optics ([28], [43]) or another type of quantum or physical phenomena [2].

Using the Meta command, the random sequence of bits creates active elements $0,1,2,\ldots$ that store the binary representation $b_0 b_1 b_2 \ldots$ of real number $x \in [0,1]$. If input effector $\mathtt{I}$ fires at time $t = n$, then $b_n = 1$; thus, we create active element $\mathtt{n}$ so that after $t = n$, element $\mathtt{n}$ fires every unit of time indefinitely. If input effector $\mathtt{I}$ does not fire at time $t = n$, then $b_n = 0$ and active element $\mathtt{n}$ is created so that it never fires. The following finite active element machine program exhibits this behavior.

```
(Program C   (Args t)
(Connection (Time t) (From I) (To t) (Amp 2) (Width 1) (Delay 1))
(Connection (Time t+1+dT) (From I) (To t) (Amp 0))
(Connection (Time t) (From t) (To t) (Amp 2) (Width 1) (Delay 1))
)
(Element (Time clock) (Name clock) (Threshold 1)   (Refractory 1)
                                                   (Last -1))
(Meta (Name I) (C (Args clock)))
```

We explain how this program exhibits this behavior, assuming the sequence of random bits from the environment begins with 1, 0, 1, ... Thus, input element I fires at times 0, 2, ... At time 0, the following commands are executed.

```
(Element (Time 0) (Name 0) (Threshold 1) (Refractory 1)(Last -1))
(C (Args 0))
```

The execution of `(C (Args 0))` causes the three connection commands to execute.

```
(Connection (Time 0) (From I) (To 0) (Amp 2) (Width 1) (Delay 1))
(Connection (Time 1+dT) (From I) (To 0) (Amp 0))
(Connection (Time 0) (From 0) (To 0) (Amp 2) (Width 1) (Delay 1))
```

Because of the first connection command

```
(Connection (Time 0) (From I) (To 0) (Amp 2) (Width 1) (Delay 1))
```

the firing of input element I at time 0 sends a pulse with amplitude 2 to element 0. Thus, element 0 fires at time 1. Then at time 1+dT, a moment after time 1, the connection from input element I to element 0 is removed. At time 0, a connection from element 0 to itself with amplitude 2 is created. As a result, element 0 continues to fire indefinitely, representing that $b_0 = 1$.

At time 1, command

```
(Element (Time 1) (Name 1) (Threshold 1) (Refractory 1)(Last -1))
```

is created. Since element 1 has no connections into it and threshold 1, element 1 never fires. Thus $b_1 = 0$.

At time 2, input element I fires, so the following commands are executed.

```
(Element (Time 2) (Name 2) (Threshold 1) (Refractory 1)(Last -1))
(C (Args 2))
```

The execution of `(C (Args 2))` causes the three connection commands to execute.

```
(Connection (Time 2) (From I) (To 2) (Amp 2) (Width 1) (Delay 1))
(Connection (Time 3+dT) (From I) (To 2) (Amp 0))
(Connection (Time 2) (From 2) (To 2) (Amp 2) (Width 1) (Delay 1))
```

Because of the first connection command

```
(Connection (Time 2) (From I) (To 2) (Amp 2) (Width 1) (Delay 1))
```

the firing of input element I at time 2 sends a pulse with amplitude 2 to element 2. Thus, element 2 fires at time 3. Then at time 3+dT, a moment after time 3, the connection from input element I to element 2 is removed. At time 2, a connection from element 2 to itself with amplitude 2 is created. As a result, element 2 continues to fire indefinitely, representing that $b_2 = 1$.

7 AEM Binary Language Recognizer

In this section, based on example 4, we show how to build a binary language recognizer, but do not focus on optimizing the speed of the binary language recognition. In the next section, we will show active elements performing simultaneous computations on an NP-hard problem.

First, we review some definitions and notation for binary strings and languages. A binary string is a finite sequence of 0s and 1s. The set of binary strings of length n is denoted as $\{0,1\}^n$. If string $w \in \{0,1\}^k$, then w's length is $|w| = k$. The set of all finite binary strings is denoted as $\{0,1\}^* = \bigcup_{n=1}^{\infty} \{0,1\}^n$. Consider function $f : \{0,1\}^* \to \{0,1\}$, then $L = f^{-1}(1)$ is a binary language. Every binary language can be represented as the inverse image $g^{-1}(1)$ for some unique function $g : \{0,1\}^* \to \{0,1\}$. For any function $f : \{0,1\}^* \to \{0,1\}$, define $f_n : \{0,1\}^n \to \{0,1\}$ such that f restricted on $\{0,1\}^n = f_n$ i.e. $f_n(u) = f(u)$ for all u in $\{0,1\}^n$.

We can create an ordering of the strings $[\{0,1\}^*, \prec]$ as follows. For any two distinct strings $u, w \in \{0,1\}^*$, then $u \prec w$ if $|u| < |w|$; $w \prec u$ if $|w| < |u|$. Otherwise, $|u| = |w|$ in which case $u \prec w$ if u is smaller than w, treating u, w as binary numbers; or $w \prec u$ if w is smaller than u as binary numbers.

There are 2^n binary strings of length n. Let $\phi(n) = 2^n$ and $\beta(n) = \phi(\phi(n))$. From the following diagram, it is an easy observation

$$
\begin{aligned}
000\ldots0 &\longrightarrow 0 \text{ or } 1 \\
100\ldots0 &\longrightarrow 0 \text{ or } 1 \\
010\ldots0 &\longrightarrow 0 \text{ or } 1 \\
110\ldots0 &\longrightarrow 0 \text{ or } 1 \\
&\ldots \\
111\ldots1 &\longrightarrow 0 \text{ or } 1
\end{aligned}
$$

that there are $\beta(n)$ distinct Boolean functions $f_n : \{0,1\}^n \to \{0,1\}$. Each integer c_n satisfying $0 \le c_n < \beta(n)$ corresponds to a distinct $f_n : \{0,1\}^n \to \{0,1\}$.

This correspondence can be implemented with an active element machine that uses the elements created in example 4 to encode for the numbers $c_1, c_2, \ldots$. The encoding of boolean function f_1 is stored (represented) in the first $\beta(1)$ active elements 0, 1, 2, 3, which represent natural number c_1. For each natural number n, the boolean function f_n is represented with the next $\beta(n)$ elements generated by the Meta command from the random input to element I, as described in example 4.

These groups of active elements representing $c_1, c_2, \ldots$ are called registers. In general, register R_{-n} stores c_n as shown in table 3.

Table 3 Register encoding of Boolean functions f_n and binary string $a_1 \ldots a_m$

Register	...	$-n$	...	-1	0	1	...	m
Contents	...	c_n	...	c_1	m	a_1	...	a_m

The binary string of length m that is accepted or not accepted by this machine is stored in registers $R_1, R_2, \ldots, R_m$ and represented as $a_1 a_2 \ldots a_m$ where each a_k is a 0 or 1. Each register R_k can be represented with a single active element J_k where it fires indefinitely if $a_k = 1$ and never fires if $a_k = 0$.

In order to refer to the register of a register, we use the notation R_{R_n}. If register R_n contains -5 and the contents of $R_{-5} = 29$, then the contents of $R_{R_n} = 29$.

Solution 1. *Binary Language Decision Steps 1 to 6*

Steps 1 to 6 decide whether a binary string $S = a_1 a_2 \ldots a_m$ is in the language determined by the active element machine. Before step 1 is started, for each natural number $n \leq m$, the value c_n is stored in register R_{-n}, where $0 \leq c_n < \beta(n)$. Since each c_n uses 2^n bits to store c_n, input element I needs $1 + 2 + \ldots 2^m$ time steps i.e. $1 + 2 + \ldots 2^m$ random bits from the environment before all these register values are assigned.

The values in registers R_1, R_2, $\ldots$, R_m are 0 or 1 where R_k, such that $1 \leq k \leq m$, stores the value of a_k. The value m is stored in register 0.

1. Read the string length m from register R_0. Compute 2^m. Store in register R_{m+1}.
2. Store 0 in register R_{m+2}. Store c_m in register R_{m+3}.
3. Initialize registers $R_{m+5}, \ldots, R_{m+4+2^m}$ with 0.
4. In a loop that is executed c_m times,
   Do
   {
     Increment register R_{m+2}.

     Increment the binary number in registers $R_{m+5}, \ldots R_{m+4+2^m}$.

   } Until register R_{m+2} equals register R_{m+3}.

5. Compute $(m + 5 + \text{binary } a_1 a_2 \ldots a_m)$ stored in R_1, R_2, $\ldots$, R_m.
   Store 1 in register R_{m+1}.

   Store $m + 5$ in register R_{m+2}.

   Store 0 in register R_{m+3}.

   Store 1 in register R_{m+4}.

   In a loop that is executed m times,
   Do
   {
     If the contents of $R_{R_{m+1}}$ equals 1, add the value in R_{m+4} to R_{m+2}.

     Increment R_{m+3}.

     Increment R_{m+1}.

     Double value in R_{m+4} and store back in register R_{m+4}.

   } Until the value in R_{m+3} equals m.

6. If register $R_{R_{m+2}}$ contains a 1 then binary string $a_1 a_2 \dots a_m$ is in the language recognized by this machine. Otherwise, $R_{R_{m+2}}$ contains a 0 and binary string $a_1 a_2 \dots a_m$ is not in the language recognized by this machine.

Example 5. Steps 1 through 6 when $c_3 = 203$
This example shows the encoding of f_3 when $c_3 = 203$ and describes steps 1, 2, ..., 6 as they are executed.

Suppose R_{-3} has $c_3 = 203$ stored in it. This example demonstrates how $c_3 = 203$ encodes for the corresponding f_3 and how $f_3(S)$ is computed for any S in $\{0,1\}^3$. Decimal number 203 in reversed binary is $11010011 = 2^0 + 2^1 + 2^3 + 2^6 + 2^7$. The ordering of $\{0,1\}^3 = [000, 100, 010, 110, 001, 101, 011, 111]$, which is iterated in this order in the loop of step 5. The bits of 1101 0011 represent the range values of f_3 where $f_3(000) =$ the 1st bit in 1101 0011 ; $f_3(100) =$ the 2nd bit in 1101 0011 ; ... ; $f_3(111) =$ the eighth (2^3) bit in 1101 0011. This is summarized in table 4.

Table 4 Boolean functions f_3

$f_3(000) = 1$	$f_3(100) = 1$	$f_3(010) = 0$	$f_3(110) = 1$
$f_3(001) = 0$	$f_3(101) = 0$	$f_3(011) = 1$	$f_3(111) = 1$

Before program execution starts the contents of $R_0 = 3$, $R_1 = 0$, $R_2 = 1$ and $R_3 = 1$. Thus, the program will decide whether $S = 011$ lies in the language determined by this machine. Steps 1 to 6 execute as follows.

- Step 1 reads 3 from R_0 and computes $2^3 = 8$. Then 8 is stored in R_4.

- Step 2 stores the contents of $R_{-R_0} = 203$ in register R_6. Also, 0 is stored in R_5.

- Step 3 stores 0 in R_8, R_9 , R_{10}, ..., R_{15}.

- Step 4 executes the loop 203 times. The binary counter increments registers R_8, R_9, R_{10}, ..., R_{15} in reversed binary representation. After executing this loop 203 times, $R_8 = 1$, $R_9 = 1$, $R_{10} = 0$, $R_{11} = 1$, $R_{12} = 0$, $R_{13} = 0$, $R_{14} = 1$, $R_{15} = 1$.

- Step 5 initializes R_{m+2} to 8. The loop adds binary number R_1, R_2, R_3 to 8 and stores it in R_{m+2}. The purpose of register R_{m+2} is that it indexes the value of $f_3(011)$ in step 6 which decides whether 011 is in the language computed by this machine. After the first pass through the loop, since $R_1 = 0$, then $R_{m+2} = 8$. After the second pass, $R_{m+4} = 2$, so $R_{m+2} = 10$. After the third and final pass through the loop, $R_{m+4} = 4$, so $R_{m+2} = 14$.

- In Step 6, the value of $R_{m+2} = 14$. Thus, $R_{R_{m+2}} = R_{14} = 1$. Thus, $f_3(011) = 1$, so 011 is in the language computed by this machine.

Storing and initializing can be implemented with a register machine; increment-ing can be implemented with a register machine; looping, doubling, counting and

adding can be implemented with a register machine; and computing 2^m involves executing a doubling routine inside a loop that is executed m times. From these observations and Theorem 1, steps 1 through 6 can be implemented with an active element machine.

Next, we present three lemmas and a theorem. These results prove that steps 1 through 6 decide whether an arbitrary binary string $S = a_1 a_2 \ldots a_m$ lies in the language determined by this active element machine.

Lemma 1. *For any $L \subseteq \{0,1\}^*$, then $L = f^{-1}\{1\}$ for some $f : \{0,1\}^* \rightarrow \{0,1\}$.*

Proof. Set $L_n = L \cap \{0,1\}^n$. Then $L_n \cap L_m = \emptyset$ whenever $n \neq m$. Define the boolean function $f_n : \{0,1\}^n \rightarrow \{0,1\}$ as follows. For each S in $\{0,1\}^n$, if S lies in L, then define $f_n(S) = 1$. If S does not lie in L, then define $f_n(S) = 0$. Set $f = \cup_{n=1}^{\infty} f_n$. Then $f^{-1}\{1\} = L$.

Lemma 2. *For each c_n, satisfying $0 \leq c_n < \beta(n)$ and such that R_{-n} contains c_n before program execution starts, then for any binary string $S = a_1 a_2 \ldots a_n$ in $\{0,1\}^n$, step 6 decides whether S lies in the language of the machine. As S ranges over each element of $\{0,1\}^n$, then this determines the set $L_n \subseteq \{0,1\}^n$ of all binary strings of length n, that are recognized by this machine.*

Proof. Step 4 is iterated c_n times where $0 \leq c_n < \beta(n)$ and each time through the loop, the contents of registers $R_{n+5} \ldots, R_{n+4+2^n}$ are incremented as a binary number where R_{n+5} stores the 2^0 bit, R_{n+6} stores the 2^1 bit, . . ., and R_{n+4+2^n} stores the 2^n bit. As a result, step 4 creates a sequence of 0's and 1's, representing the binary encoding of f_n, according to the ordering of $\{0,1\}^n$ created in step 5. Step 6 uses this encoding and ordering to determine whether a binary string S of length n is recognized by the machine.

Lemma 3. *Steps 1 through 6 with c_n initially stored in R_{-n}, where $0 \leq c_n < \beta(n)$ before program execution starts, decide a unique language $L \subseteq \{0,1\}^n$. If $b_n \neq c_n$ and $0 \leq b_n, c_n < \beta(n)$, then the languages determined by these two different values are distinct.*

Proof. Let $K_n \subseteq \{0,1\}^n$ be the language decided by machine K with register R_{-n} containing b_n. Let $L_n \subseteq \{0,1\}^n$ be the language decided by machine L with register R_{-n} containing c_n. When a binary string of length n is stored in registers $R_1, R_2, \ldots, R_n$ before program execution, then after step 4 is completed, the values of whether to accept or not accept a binary string of length n are stored as $R_{n+5}, \ldots, R_{n+4+2^n}$. If b_n stored in register R_{-n} of machine K is not equal to c_n stored in R_{-n} of machine L, then after step 4 is completed R_j in machine K is not equal to R_j in machine L for some j satisfying $n+5 \leq j \leq n+4+2^n$. This implies that language $K_n \neq L_n$.

Theorem 2. *There is a one to one correspondence between the binary languages $L \subseteq \{0,1\}^*$ and the sequence of natural numbers $c_1, c_2, \ldots, c_n, \ldots$, such that for each n, the natural number c_n satisfies $0 \leq c_n < \beta(n)$. Furthermore, if each register R_{-n} is*

initialized to c_n where $1 \leq n \leq m$ before program execution of step 1 begins, then the program execution recognizes this corresponding binary language, where the binary string of length m to be recognized is initially stored in registers $R_1, R_2, \ldots, R_m$.

Proof. This theorem follows immediately from the three previous lemmas.

8   An AEM Program Computes a Ramsey Number

In this section, we turn our attention to computing a Ramsey number with an AEM program. Ramsey theory can be intuitively described as the study of structure which is preserved under finite decomposition (see [10], [22], [36]). Applications of Ramsey theory include results in number theory [40], algebra, geometry [15], topology [20], set theory, logic [36], ergodic theory [20], computer science ([3], [4], [5], [6], [7], [8], [46]), including lower bounds for parallel sorting [16], game theory [24] and information theory ([37], [41]). Progress on determining the basic Ramsey numbers $r(k,l)$ has been slow. For positive integers k and l, $r(k,l)$ denotes the least integer n such that if the edges of the complete graph K_n are 2-colored with colors red and blue, then there always exists a complete subgraph K_k containing all red edges or there exists a subgraph K_l containing all blue edges.

To put our slow progress into perspective, arguably the best combinatorist of the 20th century, Paul Erdös asks us to imagine an alien force, vastly more powerful than us, landing on Earth and demanding the value of $r(5,5)$ or they will destroy our planet. In this case, Erdös claims that we should marshal all our computers and all our mathematicians and attempt to find the value. But suppose instead that they ask for $r(6,6)$. For $r(6,6)$, Erdös believes that we should attempt to destroy the aliens [42].

Theorem 3. *The standard finite Ramsey theorem.*

For any positive integers m,k,n, there is a least integer $N(m,k,n)$ with the following property: no matter how we color each of the n-element subsets of $S = \{1,2,...,N\}$ with one of k colors, there exists a subset Y of S with at least m elements, such that all n-element subsets of Y have the same color *(See [23], [36], [39]).*

When G and H are simple graphs, there is a special case of theorem 3. Define the Ramsey number $r(G,H)$ to be the smallest N such that if the complete graph K_N is colored red and blue, either the red subgraph contains G or the blue subgraph contains H. (A simple graph is an unweighted, undirected graph containing no graph loops or multiple edges. In a simple graph, the edges of the graph form a set and each edge is a pair of distinct vertices.) In [10], S.A. Burr proves that determining $r(G,H)$ is an NP-hard problem.

We show how to build an AEM program that solves a special case of Theorem 3. Color each edge of the complete graph K_6 red or blue. Then there is always at least one triangle, which contains only blue edges or only red edges. In terms of the standard Ramsey theorem, this is the special case $N(3,2,2)$ where $n = 2$ since we

color edges (i.e. 2-element subsets); $k = 2$ since we use two colors; and $m = 3$ since we seek a red or blue triangle.

To demonstrate how an AEM program can be designed to compute $N(3,2,2) = 6$, we first show how to build an AEM program that verifies $N(3,2,2) > 5$, based on figure 1.

Fig. 1 A 2-coloring of K_5 containing no monochromatic K_3 (triangle)
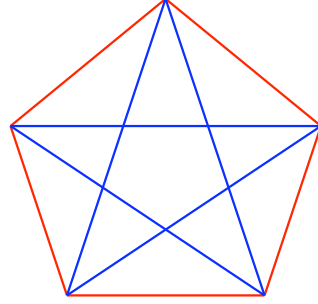
The symbols B and R represent blue and red, respectively. We put indices on B and R to denote active elements that correspond to the K_5 graph geometry. First, we explain where the indices come from. Let $E = \{\{1,2\}, \{1,3\}, \{1,4\}, \{1,5\}, \{2,3\}, \{2,4\}, \{2,5\}, \{3,4\}, \{3,5\}, \{4,5\}\}$ denote the edge set of K_5. The triangle set $T = \{\{1,2,3\}, \{1,2,4\}, \{1,2,5\}, \{1,3,4\}, \{1,3,5\}, \{1,4,5\}, \{2,3,4\}, \{2,3,5\}, \{2,4,5\}, \{3,4,5\}\}$. As shown in figure 1, each edge is colored red or blue. Thus the red edges are $\{\{1,2\}, \{1,5\}, \{2,3\}, \{3,4\}, \{4,5\}\}$ and the blue edges are $\{\{1,3\}, \{1,4\}, \{2,4\}, \{2,5\}, \{3,5\}\}$.

We number each group of AEM commands for K_5, based on the group's purpose. This is useful because we will refer to these groups, when describing the computation for K_6.

1. The elements representing red and blue edges are established as follows.

```
(Element (Time 0) (Name R_12) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name R_15) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name R_23) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name R_34) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name R_45) (Threshold 1) (Refractory 1) (Last -1))

(Element (Time 0) (Name B_13) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name B_14) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name B_24) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name B_25) (Threshold 1) (Refractory 1) (Last -1))
(Element (Time 0) (Name B_35) (Threshold 1) (Refractory 1) (Last -1))
```

2. Fire element R_jk if edge $\{j,k\}$ is red.

```
(Fire (Time 0) (Name R_12))
(Fire (Time 0) (Name R_15))
(Fire (Time 0) (Name R_23))
(Fire (Time 0) (Name R_34))
(Fire (Time 0) (Name R_45))
```

Fire element B_jk if edge $\{j,k\}$ is blue where $j < k$.

```
(Fire (Time 0) (Name B_13))
(Fire (Time 0) (Name B_14))
(Fire (Time 0) (Name B_24))
(Fire (Time 0) (Name B_25))
(Fire (Time 0) (Name B_35))
```

3. The following Meta commands cause these elements to keep firing after they have fired once.

```
(Meta (Name  R_jk) (Window 0 1)
 (Connection (Time 0) (From R_jk) (To R_jk) (Amp 2) (Width 1) (Delay 1)))

(Meta (Name  B_jk) (Window 0 1)
 (Connection (Time 0) (From B_jk) (To B_jk) (Amp 2) (Width 1) (Delay 1)))
```

4. To determine if a blue triangle exists on vertices $\{i,j,k\}$, where $\{i,j,k\}$ ranges over T, three connections are created for each potential blue triangle.

```
(Connection (Time 0) (From B_ij) (To B_ijk) (Amp 2) (Width 1) (Delay 1))
(Connection (Time 0) (From B_jk) (To B_ijk) (Amp 2) (Width 1) (Delay 1))
(Connection (Time 0) (From B_ik) (To B_ijk) (Amp 2) (Width 1) (Delay 1))
```

5. To determine if a red triangle exists on vertex set $\{i,j,k\}$, where $\{i,j,k\}$ ranges over T, three connections are created for each potential red triangle.

```
(Connection (Time 0) (From R_ij) (To R_ijk) (Amp 2) (Width 1) (Delay 1))
(Connection (Time 0) (From R_jk) (To R_ijk) (Amp 2) (Width 1) (Delay 1))
(Connection (Time 0) (From R_ik) (To R_ijk) (Amp 2) (Width 1) (Delay 1))
```

6. For each vertex set $\{i,j,k\}$ in T, the following elements are created.

```
(Element (Time 0) (Name R_ijk) (Threshold 5) (Refractory 1) (Last -1))
(Element (Time 0) (Name B_ijk) (Threshold 5) (Refractory 1) (Last -1))
```

Because the threshold is 5, we see that element R_ijk only fires when all three elements R_ij, R_jk, R_ik fired one unit of time ago. Likewise, the element B_ijk only fires when all three elements B_ij, B_jk, B_ik fired one unit of time ago. From this, we observe that as of clock $= 3$ i.e. 4 time steps, this AEM program determines that $N(3,2,2) > 5$. This AEM computation uses $|E| + 2|T| = \frac{5!}{2!3!} + 2\frac{5!}{3!2!} = 30$ active elements. Further, this AEM program creates and uses $3|T| + 3|T| + |E| = 70$ connections.

For K_6, the edge set $E = \{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,6\},\{2,3\},\{2,4\},\{2,5\},$ $\{2,6\},\{3,4\},\{3,5\},\{3,6\},\{4,5\},\{4,6\},\{5,6\}\}$. The triangle set $T = \{ \{1,2,3\},$ $\{1,2,4\},\{1,2,5\},\{1,2,6\},\{1,3,4\},\{1,3,5\},\{1,3,6\},\{1,4,5\},\{1,4,6\},\{1,5,6\},$ $\{2,3,4\},\{2,3,5\},\{2,3,6\},\{2,4,5\},\{2,4,6\},\{2,5,6\},\{3,4,5\},\{3,4,6\},\{3,5,6\},$ $\{4,5,6\}\}$. For each 2-coloring of E, each edge is colored red or blue. There are $2^{|E|}$ 2-colorings of E. For this graph, $|E| = \frac{6!}{2!4!}$.

To build a similar AEM program, the commands in groups 1 and 2 range over every possible 2-coloring of E. The remaining groups 3, 4, 5 and 6 are the same based on the AEM commands created in groups 1 and 2 for each particular 2-coloring.

This AEM program verifies that every 2-coloring of E contains at least one red triangle or one blue triangle i.e. $N(3,2,2) = 6$. We make no optimizations using

graph isomorphisms [32]. If we build an AEM language construct for generating all active elements for each 2-coloring of E at time zero, then the resulting AEM program can determine the answer in 5 time steps. (We need one more time step, 2^{15} additional connections and one additional element to verify that every one of the 2^{15} AEM programs is indicating that it found a red or blue triangle.) This AEM program – that determines the answer in 5 time steps – uses $2^{|E|}(|E| + 2|T|) + 1$ active elements and $2^{|E|}(3|T| + 3|T| + |E| + 1)$ connections, where $|E| = 15$ and $|T| = 20$.

9   Discussion and Further Work

First, we summarize the results in section 7. It is well-known ([13], [33], [45]) that there are an uncountable number of binary languages in $\{0,1\}^*$, for which no standard Turing machine with a finite number of non-blank symbols on the tape is able to recognize. This is also true for a finite register machine program ([13], [33], [44]). Yet by building upon example 4, using randomness from the environment and the Meta command, we were able to generate an AEM with a finite number of AEM commands, that is able to recognize an arbitrary language $L \subseteq \{0,1\}^*$. This suggests that evolutionary methods ([9], [14], [17], [26], [27], [29]) along with quantum randomness ([11]) may be able to implicitly design AEMs that can exhibit useful computing behavior, which explicit register machine programs are unable to attain.

In regard to section 8, an extension of the Ramsey theorem (large Ramsey numbers) occurs when the set Y is large. A set Y is large if its cardinality is larger than its smallest element (e.g. $Y = \{1,2,3\}$). Large Ramsey numbers $L(m,k,n)$ grow too fast to be provably total in Peano arithmetic [34]. As a consequence of the extremely high growth rate of large Ramsey numbers, we expect that studying the computation of these numbers with AEM programs will be of interest in terms of understanding how to best use parallelism, geometry and time. Another area of interest is the trade-off of using a separate AEM to find graph isomorphisms that eliminate isomorphic 2-colorings, when testing for a red or blue triangle [32]. An additional area to explore would use AEM computations that rely on the simple property that every subgraph, of a monochromatic graph G, is monochromatic.

References

1. Abelson, H., Sussman, G.J.: Structure and Interpretation of Computer Programs. MIT Press, Cambridge (1996)
2. Agnew, G.B.: Random sources for cryptographic systems. In: Price, W.L., Chaum, D. (eds.) EUROCRYPT 1987. LNCS, vol. 304, pp. 77–81. Springer, Heidelberg (1988)

3. Ajtai, M., Komlós, J., Szemerédi, E.: A note on Ramsey numbers. Journal Combinatorial Theory 29(3), 354–360 (1980)

4. Ajtai, M., Komlós, J., Szemerédi, E.: A dense infinite Sidon sequence. European J. Combin. 2(1), 1–11 (1981)

5. Ajtai, M., Komlós, J., Szemerédi, E.: Sorting in $c \log n$ parallel steps. Combinatorica 3(1), 1–19 (1983)

6. Alon, N.: Eigenvalues, geometric expanders, sorting in rounds, and Ramsey theory. Combinatorica 6(3), 207–219 (1986)

7. Alon, N.: Explicit Ramsey graphs and orthonormal labelings. Electron. J. Combin. Research Paper 12 1, 8 (1994) (electronic)

8. Alon, N.: The Shannon capacity of a union. Combinatorica 18(3), 301–310 (1998)

9. Axelrod, R., Hamilton, W.D.: The Evolution of Cooperation. Science New Series 211(4489), 1390–1396 (1981)

10. Burr, S.A.: Determining Generalized Ramsey Numbers is NP-hard. Ars Combinatoria 17, 21–25 (1984)

11. Calude, C.S., Dinneen, M.J., Dumitrescu, M., Svozil, K.: Experimental Evidence of Quantum Randomness Incomputability. Phys. Rev. A 82, 022102, 1–8 (2010)

12. Cook, S.A.: The complexity of theorem proving procedures. In: Proceedings, Third Annual ACM Symposium on the Theory of Computing, pp. 151–158. ACM, New York (1971)

13. Davis, M.: Computability and Unsolvability. Dover Publications, New York (1982)

14. Eberhart, R.C., Kennedy, J.: A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, pp. 39–43 (1995)

15. Erdös, P., Szekeres, G.: A combinatorial problem in geometry. Compositio Math. 2, 464–470 (1935)

16. Erdös, P., Rado, R.: Combinatorial theorems on classifications of subsets of a given set. Proc. London Math. 3(2), 417–439 (1952)

17. Fiske, M.S.: Machine Learning. US 7,249,116 B2 (2003),
http://tinyurl.com/3pcgfvr, or http://tinyurl.com/3pcgfvr

18. Fiske, M.S.: Effector Machine Computation. US 7,398,260 B2 (2004), Provisional 60/456,715 (2003),
http://tinyurl.com/6l5wuhz or http://tinyurl.com/6l5wuhz

19. Fiske, M.S.: Active Element Machine Computation. US Application 20070288668 (2007),
http://tinyurl.com/62gv8ke or http://tinyurl.com/62gv8ke

20. Furstenberg, H.: Recurrence in Ergodic Theory and Combinatorial Number Theory. Princeton University Press, Princeton (1981)

21. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman (1979)

22. Graham, R.L., Rodl, V.: Numbers in Ramsey Theory. Surveys in Combinatorics. LMS Lecture Note Series 123. Cambridge University Press (1987)

23. Graham, R.L., Rothschild, B.L.: A Survey of Finite Ramsey Theorems. In: Proc. 2nd Louisiana State Univ. Conference on Combinatorics, Graph Theory and Computation, pp. 21–40 (1971)

24. Hales, A.W., Jewett, R.I.: Regularity and positional games. Trans. Amer. Math. Soc. 106, 222–229 (1963)

25. Hertz, J., Krogh, A., Palmer, R.G.: Introduction To The Theory of Neural Computation. Addison-Wesley Publishing Company, California (1991)
26. Holland, J.H.: Outline for a logical theory of adaptive systems. JACM 3, 297–314 (1962)
27. Holland, J.H.: Adaptation in Natural and Artificial Systems. MIT Press, Cambridge (1992)
28. Quantique, I.D., Quantis, S.: Random Number Generation Using Quantum Optics (electronic), Geneva, Switzerland (2001-2011),
    http://www.idquantique.com/images/stories/PDF/quantis-random-generator/quantis-whitepaper.pdf
29. Koza, J.R.: Genetic Programming: On the Programming of Computer by Means of Natural Selection. MIT Press, Cambridge (1992)
30. Lee, E.A.: Computing Needs Time. Technical Report No. UCB/EECS-2009-30 (electronic). Electrical Engineering and Computer Sciences, University of California at Berkeley (2009),
    http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-30.html
31. McCulloch, W.S.: Walter Pitts. A logical calculus immanent in nervous activity. Bulletin of Mathematical Biophysics 5, 115–133 (1943)
32. McKay, B.D., Radziszowski, S.P.: Subgraph counting identities and Ramsey numbers. Journal Combinatorial Theory 69(2), 193–209 (1997)
33. Minsky, M.: Computation: Finite and Infinite Machines, 1st edn. Prentice-Hall, Inc., Englewood Cliffs (1967)
34. Paris, J., Harrington, L.: A mathematical incompleteness in Peano arithmetic. In: Barwise, J. (ed.) Handbook for Mathematical Logic. North-Holland (1977)
35. Rall, W.: The Theoretical Foundation of Dendritic Function. In: Segev, I., Rinzel, J., Shepherd, G. (eds.) Selected Papers of Wilfrid Rall with Commentaries. MIT Press, Cambridge (1995)
36. Ramsey, F.P.: On a problem of formal logic. Proc. London Math. Soc. Series 2(30), 264–286 (1930)
37. Roberts, F.S.: Applications of Ramsey theory. Discrete Appl. Math. 9(3), 251–261 (1984)
38. Robinson, A.: Non-standard Analysis. Princeton University Press, Princeton (1996), Revised Edition
39. Rothschild, B.L.: A generalization of Ramsey's theorem and a conjecture of Erdös. Doctoral Dissertation. Yale University, New Haven, Connecticut (1967)
40. Schur, I.: Uber die Kongruenz $x^m + y^m \equiv z^m (\mod\ p)$. Deutsche Math. 25, 114–117 (1916)
41. Shannon, C.E.: The zero error capacity of a noisy channel. Transactions on Information Theory Institute of Radio Engineers IT-2, 8–19 (1956)
42. Spencer, J.H.: Ten Lectures on the Probabilistic Method, p. 4. SIAM (1994)
43. Stefanov, A., Gisin, N., Guinnard, O., Guinnard, L., Zbinden, H.: Optical quantum random number generator. Journal of Modern Optics 10, 1362–3044 (2000)
44. Sturgis, H.E., Shepherdson, J.C.: Computability of Recursive Functions. J. Assoc. Comput. Mach. 10, 217–255 (1963)
45. Turing, A.M.: On computable numbers, with an application to the Entscheidungsproblem. Proc. London Math. Soc. Series 2(42) (Parts 3 and 4), 230–265 (1936); A correction, ibid. 43, 544–546 (1937)
46. Yao, A.C.C.: Should tables be sorted? J. Assoc. Comput. Mach. 28(3), 615–628 (1981)

Appendix

We present a Turing Machine definition, where the program definition η is explicitly represented as a function instead of quintuples ([13], [45]).

Definition 13. *Turing Machine*
A Turing machine is a triple (Q, A, η) where

- Q is a finite set of states that does not contain the halt state. The states are sometimes represented as natural numbers $Q = \{2, ..., K\}$. There is a unique halt state, represented as h or as 1.

- When machine execution begins, the machine is in an initial state s and $s \in Q$.

- A is a finite set of alphabet symbols that are read from and written to the tape.

- -1 and $+1$ represent advancing the tape head to the left or right square, respectively.

- η is a function where $\eta : Q \times A \to (Q \cup \{h\}) \times A \times \{-1, +1\}$. η acts as the program for the Turing machine. For each q in Q and α in A, $\eta(q, \alpha) = (r, \beta, x)$ describes how machine (Q, A, η) executes one computational step. When in state q and scanning alphabet symbol α on the tape:

    - Machine (Q, A, η) changes to state r.
    - Machine (Q, A, η) rewrites alphabet symbol α as symbol β on the tape.
    - If $x = -1$, then machine (Q, A, η) moves its tape head one square to the left on the tape and is subsequently scanning the symbol in this square.
    - If $x = +1$, then machine (Q, A, η) moves its tape head one square to the right on the tape and is subsequently scanning the symbol in this square.
    - If $r = h$, machine (Q, A, η) enters the halting state h, and the machine halts.

Definition 14. *Turing Machine Tape*
The Turing machine tape T is represented as a function $T : \mathbb{Z} \to A$ where $\mathbb{Z}$ denotes the integers. The tape T is M-bounded if there exists a bound $M > 0$ such that $T(k) = T(j)$ whenever $|k|, |j| \geq M$. The Turing machine definitions in [13] and [45] assume the initial tape, before program execution begins, is M-bounded and the tape contains only blank symbols, denoted here as #, outside the bound. The symbol on the kth square of the tape is $T(k)$.

Definition 15. *Turing Machine Configuration with Tape Head Location*
Let (Q, A, η) be a Turing machine with tape T. A configuration is an element of the set $C = (Q \cup \{h\}) \times \mathbb{Z} \times \{T : T$ is tape with range $A\}$. If (q, k, T) is a configuration, then k is called the tape head location.

Consider the configuration $(p, 2, \dots \#\#\alpha\underline{\beta}\#\#\dots)$. The 1st coordinate indicates that the Turing machine is in state p. The 2nd coordinate indicates that its tape head is currently scanning tape square 2, denoted as $T(2)$. The 3rd coordinate indicates that tape square 1 contains symbol α, tape square 2 contains symbol β, and all other tape squares contain the # symbol.

Definition 16. *Turing Machine Computational Step*

Given Turing machine (Q, A, η) in current configuration (q, k, T) such that $T(k) = \alpha$. After the execution of one computational step, the new configuration is determined by one and only one of the four cases

1. $(r, k-1, S)$ if $\eta(q, \alpha) = (r, \beta, -1)$ for non-halting state r
2. $(r, k+1, S)$ if $\eta(q, \alpha) = (r, \beta, +1)$ for non-halting state r
3. $(h, k+1, S)$ if $\eta(q, \alpha) = (h, \beta, +1)$ for halting state h
4. $(h, k-1, S)$ if $\eta(q, \alpha) = (h, \beta, -1)$ for halting state h

such that for all four cases the new tape $S(j) = T(j)$ whenever $j \neq k$ and $S(k) = \beta$. In cases (3) and (4), the machine execution halts.

If the machine is currently in configuration (q_0, k_0, T_0) and over the next n steps the sequence of machine configurations (points) is (q_0, k_0, T_0), (q_1, k_1, T_1), (q_2, k_2, T_2), $\ldots, (q_n, k_n, T_n)$, then this execution sequence is sometimes called the next $n+1$ computational steps.

Cybernetic Approach to Project Management: Where Sense Making Intelligence Is Needed

Bogdan Lent

Abstract. The original cybernetics of Norbert Wiener concerns self-regulation and equilibrium stabilisation around specified goal – mainly through negative feedback. This is an attractive preposition for project management. Yet complexity and chaos of projects are better reflected by non-linear systems, which in turn are better manageable in adaptive and self-organised distributed systems with positive feedback.

Paper presents the mental model of project management based on cybernetic system approach with several asynchronously running decentralised subsystems based on specific component-goal oriented processes.

Without claming the wholeness or completeness of the solution the indices of possible project performance improvements provide sufficient justification for continuing research in this area.

Keywords: Cybernetic project management, non-linear systems in projects, finite models, initial conditions sensibility, complex systems, negative and positive feedbacks, multiple equilibriums, L-Timer™.

1 Introduction

Project Management is about control and communication with more or less well defined goal. Wiener named this field of theory, "whether in the machine or in the animal" cybernetics after the Greek steersman (χυβερνήτης) (Wiener 1948, 1961). The general model of cybernetic system with feedback through the environment is depicted in Fig. 1.

This androgynous approach suits well the project management Humans, aided with technical means attempt to bring project towards predefined goals. The Environment which provides the Feedback is the project, System Mechanics – the project management, Goal – the project goal. Wiener tried to express his cybernetic model with one equation. As it may be possible for logical variables (and still

Bogdan Lent
University of Applied Sciences Bern, Switzerland, National Defense University
Warsaw and University of Technology and Life Sciences Bydgoszcz,
Poland, Kasetsart University Bangkok, Thailand
e-mail: `bogdan lent@lent.ch`
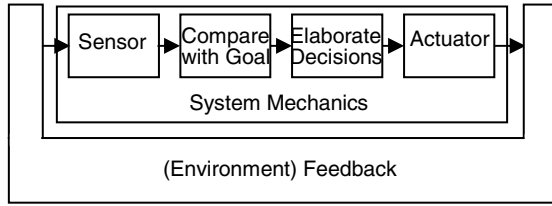
Fig. 1 General cybernetic model

complex, see (Lent 1989)), for multi value variables we obtain complex non-linear relationship (Wiener, 1948, 1961), (Kaplan, 1984, 1991). Complex systems do not lock into stable state but also do not dissolve in chaos. They store the information and exchange it. Complex systems are spontaneous, adaptive and alive (Waldrop 1992). Usually the system theories view the cybernetic systems as closed systems with predictable equilibrium; in early approach through finite number of possible states. However, the non-linear nature exposes high sensitivity to the initial conditions and multiple equilibriums. Cybernetics treats the environment as a similar system – which is not the case (each project is different). In effect a finite models of dynamic systems are limited in their predictability leading to the unavoidable imprecision. Due to the phenomenon of chance, predictability is bound to the probability and hardly deterministic in such systems (Stewart 2002). Yet, the short term predictability is nevertheless feasible (Bousquet 2009), so it makes sense to try to exercise the project management.

Von Foerster introduced 1974 the second order feedback loop: his observer (the system mechanics in the Fig. 1) is a cybernetic system with own loop itself (von Foerster 1974).By deploying a number of such second order cybernetic systems we attempt to view our first order system through the filter of particular second order sensors as shown in Fig. 2. It is a linearization of all other variables beside
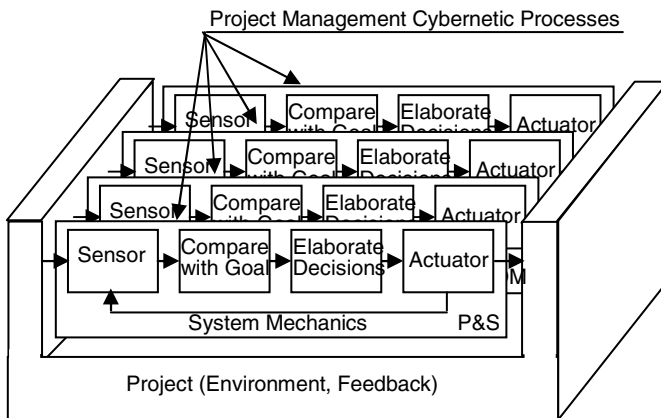
Fig. 2 Cybernetic model of Project management

those, treated by the individual cybernetic system: the project management process e.g. P&S for Planning and Scheduling or L for Leadership.

2 Mental Model of Project Management

In search of mental model we focused on easy mnemonic, which should help to manage the complexity of project management. The time clock with processes assigned to full hours appeared to be the best choice here. There are 12, 18 and 24 options of process assignment.

An evaluation of project manager activities brought the number of up to 800 single actions with numerous linear and non-linear interconnections (Rufenacht 2005). The cognitive relationships indicated the selection of 18 processes to be the best option. The heuristic process selection is based on vast experience, literature study and an analysis of the interactions between the single actions. The results were verified in several studies and practical deployment in project management daily operations since 2003 (Rufenacht 2005).

In this approach the recurrence of daily cycle secures the minimal sensitivity of each process. The mental cycle must not be diurnal: the 24 hours may go through within few minutes of project manager reflections on his activities or stretch over weeks and may occur in coincidental sequence. The key issue is recurrence itself. L-Timer™ system handles twelve administrative processes during the day time (like working day): each hour one process, in a logical sequence, which base on a macro linear interrelation; and bihourly six human factor processes (Fig. 3).
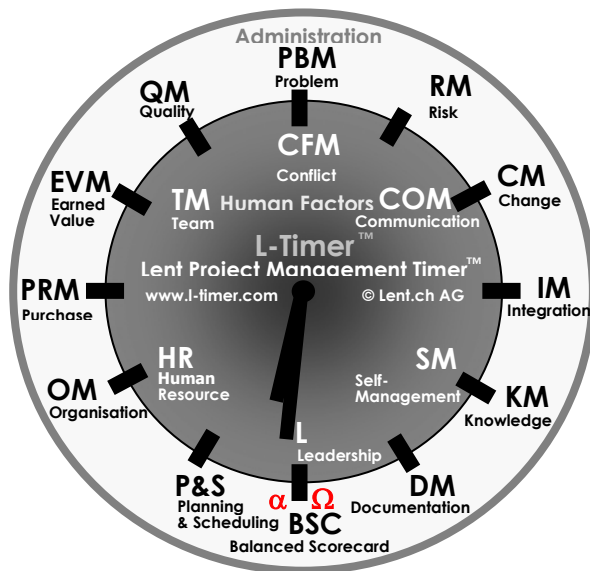
Fig. 3 L-Timer™ cybernetic processes system

The Human Factor processes are aligned with relevant administrative processes with closest links (e.g. 20:00 Human Resource Management is triggered by 8:00 Organisation Management) (Lent 2004). For readability the administrative processes are split into Table 1 and Table 2. Table 3 lists the human factor processes.

Table 1 L-Timer™ administrative processes, part 1

Process	name	Objectives
PS	Planning and Scheduling	You elaborate, structure and plan the objectives of your project. Project targets are aligned with the overall assignment specified by the customer and the higher-ranking enterprise strategy and are guaranteed over the entire duration of the project.
OM	Organization Management	You define project roles, responsibilities and the form of the organisational structure for the successful realisation of your project.
PM	Purchase Management	Through formal relationship with suppliers over all phases of the project, you secure the proper procedures and optimal results, along the formal laws, regulations and enterprise guidelines.
EVM	Earned Value Management	You control the activities in the project according to the result / deadline / cost stipulations set up in the Planning and Scheduling, with consideration for unforeseen events in the project.
QM	Quality Management	You constantly monitor project results, project processes and the other characteristics for compliance with project target stipulations, project requirements and their implementation planning, and promptly draw attention to deviations.
PBM	Problem Management	Together with your team and the applied methodology you master the technical or organisational problems within the cost and time-frame of your project.

Not coincidentally also the L-Timer™ hands point at 6:30. With the last working day results closed before at 18:00 (Balanced Scorecard) we recall the project strategy and with Leadership approach we are ready to start the day with 7:00 P&S Planning & Scheduling.

In the cybernetic system's approach the processes are not bound with specific project phases like in most recognized standards PMI (PMI 2004), IPMA (Caupin et al. 2007), but rather continuously work in loop between sensors and actuators.

Obviously we distinguish the project phases, which each process passes. The Rubicon model of four phases is the model of choice (Lent 2004).

Table 2 L-Timer™ administrative processes, part 2

Process	name	Objectives
RM	Risk Management	You minimise the overall risk to your project by permanent, creative and timely identification of potential risks, their analysis and the development of suitable countermeasures.
CM	Change Management	You ascertain, assess and decide on the implementation of proposed changes with a systematic procedure, introduce them – keeping their effects to a minimum – to the planned project handling and have the updated configuration of the system continuously under your control.
IM	Integration Management	According to the project plan and schedule you ensure that the elaborated solutions are embedded problem-free into the existing environment (organisation, human resources, applications, platforms) and that a high level of client and personnel satisfaction is achieved with its introduction.
KM	Knowledge Management	You acquire and store process experiences gained in the course of the project for its use in the current project and in other projects.
DM	Documentation Management	You ensure the documentation and archiving of project results for ease of access during project realisation, the successful placing in operation of the project results, cost-effective operation and full user satisfaction.
BSC	Balanced Scorecard	You submit the results of your project to an internationally recognised, integral and comprehensive evaluation with the aim of making a permanent, positive contribution to the implementation of enterprise strategy in your company.

We demonstrate hereafter the feasibility of this approach beyond the conventional project progress control or risk management with the process of the Human Factor group, namely the 06:00 L – the Leadership process.

Table 3 L-Timer™ Human Factor processes, part 2

Process name		Objectives
HRM	Human Resource Management	You select personnel for appointment to the formal and informal project roles best suited to their skills and experience and promote their personal further development according to the enterprise strategy
TM	Team Management	You ensure the best possible efficiency of the complete project team measured against yielded performances, staff satisfaction client satisfaction and process improvement.
CFM	Conflict Management	You promptly identify potentials for conflict in your team and in the overall project environment. You solve conflicts successfully with suitable methods and technologies.
COM	Communication Management	You master the effective communication, including that of marketing, devoted to the achievement of project goals, both in the project and its environment.
SM	Self Management	Your personal satisfaction and performance is very important. You promote it through effective self-appraisal and dealings with your own engaged resources.
L	Leadership	You skilfully and consciously control the behaviour of your team members to guarantee the achievement of the project goals.

3 Second Order Cybernetics of Project Management

The second order loop is composed by the project management cybernetic processes. Exemplary Leadership process is shown in Fig. 4. We have our time-triggered activities (initiated by 06:00) or other process initiates the Leadership, what corresponds to the changed invariants in Leadership non-linear system description. Four phases follows: L (Launch), E (Engage), A (Act) and D (Deliver), LEAD in short. In each phase the results reached are measured and corrective (negative feedback) actions are initiated. Project manager applies his third order cybernetics to aggregate the experiences and act on situation taking his mental model in account. This unpredictability of his decisions, due to the changing mental model may be viewed as the initial values change in non-linear second order cybernetic loops of the process. It implements directly the system positive feedback: by modifying and adapting the system control mechanics.
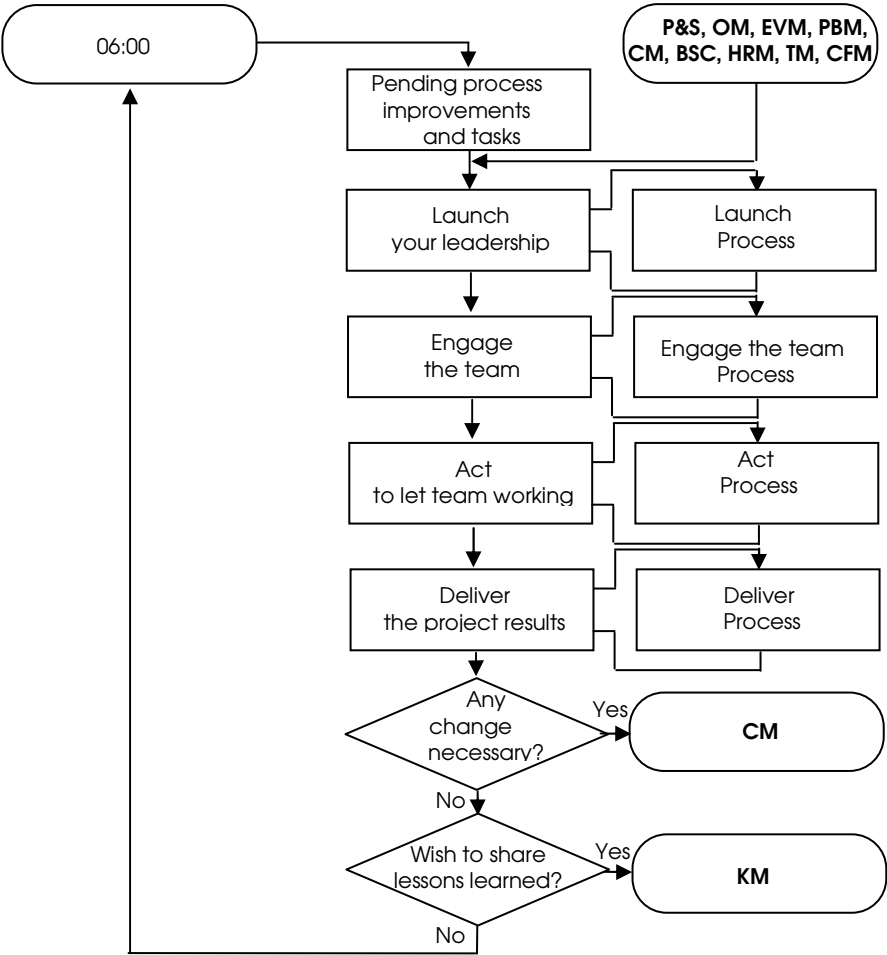
Fig. 4 Leadership process in project management

The interaction with the environment (project in this case) was evaluated a.o. by Kinicki et al. (Kinicki et al. 2011). In this model the team member (subordinate) behaviours are feedbacks to the leadership action. The reflecting awareness (Lee and Roberts 2010) allows for analysis and interpretation leading to the corrective actions which close the cybernetic loop (see Fig. 5). The situational (environmental) variables impact the goals and expectations as well as the feedbacks of the team members.

The individual project management cybernetic processes remain complex issues. For the purpose of these considerations the Gell-Mann definition of complexity is used. In his view, the simpler the system, the shorter is it's description (information content). Opposite, the longer the description, the more complex it is (Gell-Mann 1994). Systems highly ordered (perfectly linear) and chaotic (fully

non-linear) have short description, in-between lays the complexity. As we can not assume that the environment (the project) behaves along the logical rules or as expected, we face rather the complexity at the verge of chaos.
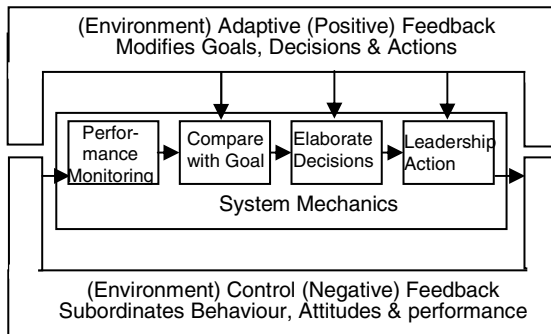
Fig. 5 Cybernetic Model of Leadership of Kinicki et al (Kinicki et al. 2011)

4   Third Order Cybernetics of Project Management: The Project Manager

Project Manager himself is the Von Foerster Observer (third level loop) in this second level project management processes' loop (von Foerster 1974). Project Manager cybernetic loop is shown in Fig. 6. It holds true for all decisions taken by project manager in any project management process.
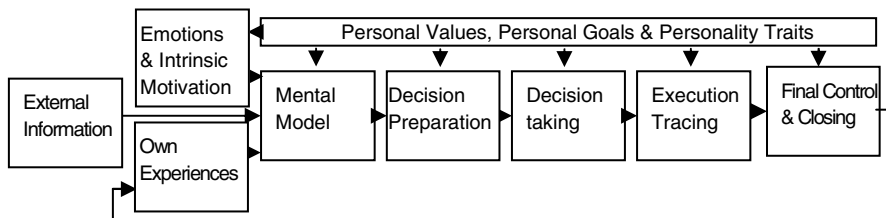
Fig. 6 Decision process

Managers see the system as linear one and try to master the negative feedback loop (e.g. by increasing the frequency of project progress control) imposing order. (Fig. 7).

The leaders, oriented towards dealing with the uncertainty of nonlinear systems, focus on positive feedback. They let the system to certain degree freely float or even intentionally destabilize, to learn the equilibriums and the resistance to change around those points. This adaptive learning let leaders to develop the

cognitive intuition (Bousquet 2009). This operation on the verge of chaos is viewed by several authors as the most successful strategy to deal with the non-linear systems (Bousquet 2009), (Kaufmann 1955), (Singh and Singh 2002). The right approach is that of manager and leader: to keep balance between positive and negative feedbacks (Bousquet 2009).
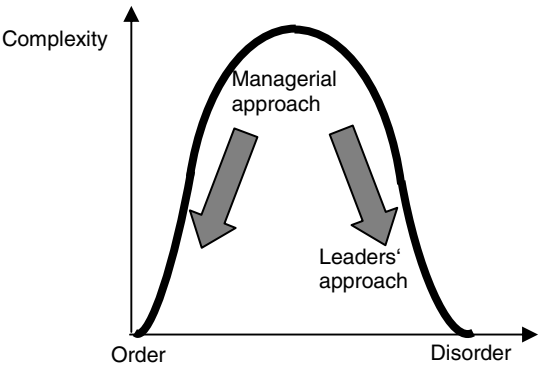
Fig. 7 Complexity adapted from (Gell-Mann 1994)

The profiles of those positive (leader) and negative (manager) feedbacks illustrate Table 4, modified from Verma (Verma 1996).

Table 4 The roles of project manager and leader modified from Verma (Verma 1996).

Project Manager	Project Leader
Planning and budgeting	Setting vision and direction
Organizing team	Inspiring team work
Staffing the roles	Aligning the team members
Controlling the results	Motivating and supporting
Sustaining the structure	Adapting the structure

We lend credibility to Singh and Singh (Singh and Singh 2002), who conclude that project managers have to balance between linear (management) systems and non-linear systems, effective in chaos and complexity management. The high degree of the complexity at the edge of chaos is simultaneously the biggest chance: the management systems handling these situations are most flexible and creative, best suited to adapt for a contingent operation and handle the unpredictability

(Bousquet 2009) To handle the last, an awareness of context and relations, even anticipation of their possibility, may be crucial to project success. Linear systems focus on quantitative analysis and project controls limiting the capability of the perception of deviations or stochastic occurrences with impact on the project fate, what may explain, why today's project are not better managed than 10 and 20 years ago.

When we consider stress situations, typical in any project day life, we observe, that our approach is rather that of problem solving than the systematic development of understanding, evaluation of alternatives, and risk analysis. We act mostly instantly and spontaneously, without questioning assumptions or implications of our action. This reaction comes from our sense making capability in view of non-linear system encounter, paired with the intuition (Thomas and Mengel 2008).

The key issue in sense making intelligence plays mental model of the situation (Fig. 8) (Lent 2009).
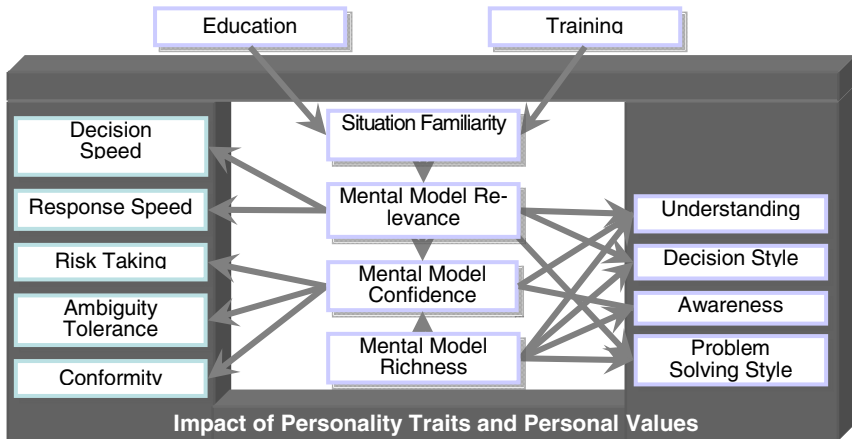
Fig. 8 Mental Model adapted from (RTO 2007)

The confidence, thus capability to develop the decision base, particularly in complex situations, depends on mental model richness and it's relevance (RTO 2007). Relevance is impacted by situation familiarity which in turn is modified by education and training. As non-linearity predominates project manager decisions, the associations training and best case education might be better choice here.

The capability to modify the mental model based on experience (training) is a positive feedback systems mechanics modification. It counterbalances the negative feedback (performance monitoring in Fig. 5). Yet the model itself is the necessary prerequisite to set the measures, wherewith the primacy of positive feedback over negative feedback is demonstrated.

5 Conclusions

Projects with specified goals and the project management tracing these goals build together the first order cybernetic system. The complexity and non-linearity of this system exclude the reasonable mathematical modelling and limit the predictability of such systems to certain probability. Only within a short time period certain predictability is achievable, thus negative feedback control might be effective.

On a long term run the balance of positive and negative feedbacks seems to be more effective. To handle it, the second and third order cybernetic systems are conceived. In the proposed mental model of project management (system mechanics in Wiener first order cybernetic model) all activities are spilt into 18 in parallel operating closed loops of second order cybernetic systems. Each of them deals with the specific issues of the process, which constitute the second order cybernetic system mechanics, treating the other project issues as invariants. Still remaining complexity and non-linearity calls for project manager abilities to act as a manager as well as a leader. Project manager is perceived as a third order cybernetic system with both the negative feedback loop (manager, striving for short term feasible control) and the positive feedback loop, bound to the leadership qualities and sense making intelligence. By handling the systems on the verge of chaos leader can better enrich his mental model identifying hidden patterns in project course and possible multiple equilibriums. Through this he wins the confidence to handle the uncertainties in project. The theoretical considerations and practice confirm overall better performance of this qualitative approach as compared to negative feedback quantitative management only.

As each project by definition differs from the others, the positive feedback loop provides the necessary adaptability and cybernetic systems evolution towards better match to project behaviours. The third order system of human handling of the project needs to tilt towards leadership positive feedback loop to be effective. Knowledge Management process KM collects and redistribute the adaptations in the second order cybernetic system - the project management. This secures the second order system evolution. The better adapted project manager and the continuously adapted project management processes are promising approach to cause also the first order cybernetic system positive evolution: this of a project and it's project management.

Author is thankful to the reviewers for taking the effort of the paper evaluation and to the editors for the opportunity to share the above views.

References

Bousquet, A.: Scientific Way of Warfare: Order and Chaos on the Battlefields of Modernity. Columbia University Press, New York (2009)

Caupin, G., et al.: International Project Management Association: International Competence Baseline (version 3). IPMA (2007), ISBN 0-9553213-0-1

Gell-Mann, M.: The Quark and the Jaguar: Adventures in the Simple and the Complex. Little Brown, London (1994)

Kaplan, F.: The wizards of Armageddon. Simon & Schuster, reissued Standford University Press, Stanford, USA (1984)

Kaufmann, S.: At Home in the Universe: The Search for Law of Self-Organization and Complexity. Oxford University Press, London (1955)

Lee, G., Roberts, I.: Coaching for authentic Leadership. In: Passmore, J. (ed.) Leadership Coaching: Working with Leaders to Develop Elite Performance, Kogan Page Ltd., London (2010)

Lent, B.: Dataflow Architecture fo Machine Control. Research Studies Press, Ltd., Taunton (1989)

Lent, B.: IT-Projekte lenken - mit System. Vieweg, Wiesbaden (2004)

Lent, B.: Human Factor Skills of Project Managers Derived from the Analysis of the Project Management Processes. PMI AGC, Bahrain (2009)

PMI, A Guide to the Project Management Body of Knowledge PMBOK Guide, 3rd edn. PMI Inc., Newtown Square (2004)

RTO Technical Repport TR-SAS-050, Exploring New Command and Control Concepts and Capabilities, NATO Neuilly-sur-Seine Cedex, France (2007)

Rufenacht, U.: Projektschlussbeurteilung P05/ProCur (Final report on Project P05/ProCur), Internal Report Swiss Ministry of Finances, Bern, Switzerland (2005)

Singh, H., Singh, A.: Principles of complexity and chaos theory in project execution: a new approach to management cost engineering. Cost Engineering 44(12), 23–32 (2002)

Stewart, I.: Does God Play Dice?: The New Mathematics of Chaos. Blackwell Publishing, MA (2002)

Thomas, J., Mengel, T.: Preparing project managers to deal with complexity – Advanced project management education. International Journal of Project Management 26(3), 304–315 (2008)

Verma, V.K.: Human Resource Skills for the Project Manager. PMI, Newtown Square (1996)

von Foerster, H.: Cybernetics of Cybernetics, Biological Computer Laboratory, University of Illinois, Urbana-Champaign (1974)

Waldrop, M.M.: Complexity, The Emerging Science at the Edge of Order and Chaos. Simon & Schuster Paperbacks, New York (1992)

Wienert, N.: Cybernetics: or Control and Communication in the Animal and the Machine. MIT, Massachusetts (1948, 1961)

Multivariate Adaptive Embedding, MAE-Process

Gerhard Sartorius

Abstract. The multivariate adaptive embedding (MAE-Process) provides an adaptive System which creates artificial neural network in the form of an appropriate model of the training-data set by using a globally optimal optimisation and acts in most cases without iterations and parameter settings. There is basically no change to input data and the training-data set is prepared by a special fitting method in order to make it treatable for spectral methods. In the working phase, new input data can be processed multivariatly by the system with good generalising properties. In combination with a Wavelet transformation (WT) for noise- and data reduction, the System performs fast and efficiently in classifying parameterised curves, such as Raman spectra.

Keywords: Machine learning / Multi-class classification / Cluster identification.

1 Introduction

The modularity of the process allows to separate the training- and working units (see fig. 1), as well as the separated carrying out of the training- and the working phase and supports the use of standard components and software. Furthermore, this modularity provides a high degree of flexibility and reduces the amount of required components when multiple working units are used. The training-data set consists of various classes of Raman spectra, measured from their respective organic substances. After designing the model of the training-data set (TDS) in the training unit (training phase), it will be transferred into one or more working units, and the working units are now able to classify untrained input Data (ID) independently (working phase) showing not only hits to a receptive area, but also a degree of membership and the tendency of a measuring point to a trained class.

Gerhard Sartorius
Lehrgebiet Informationstechnik, Fernuniversität Hagen,
Hagen, Germany
e-mail: gerhard.sartorius@t-online.de
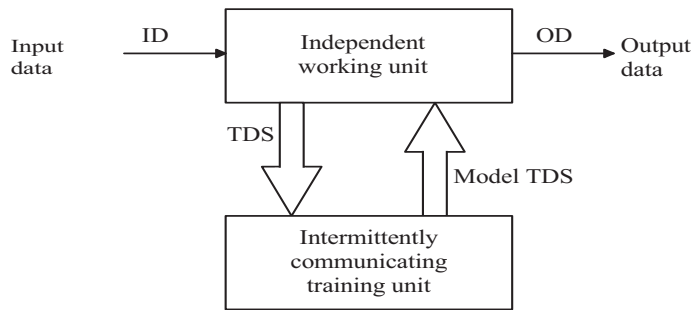
Fig. 1 Interaction between training- und working unit

List of symbols

X-space:	Input space	MF:	Manifold
Y-space:	Embedding space	ANN:	Artificial Neuronal Net
Z-space:	Feature space	WC:	Wavelet coefficient
TP	Training point	WP:	Working point, request point
X_i:	Point in X-space	a:	Index of a request point
Y_i:	Point in Y-space	i:	Index of mapping scale
Z_i:	Point in Z-space	N:	Number of X_i
D:	Dimension of X-space	d:	Dimension of Y-space,
WT:	Wavelet transform		Z-space
NN-...:	Nearest-neighbour...	TDS:	Training data set
NN-point:	Nearest-neighbour point	W_{aj}, W_{ij}:	Reconstruction weights
NN-radius:	Nearest-neighbour radius	K:	Number of NN
NN-search:	Nearest-neighbour search	η_j:	Distance to NN
NOP:	Neighborhood preserving	ROC:	Receiver Operating
	Orthogonal Projection		Characteristics
ID:	Input data	OD:	Output data
TU:	Training unit	WU:	Working unit
MU:	Measuring unit		
NR:	Neighbourhood preserving	DR:	Dimensions reduction
	relationship	SA:	Scaling and adaption

2 Advantages of the MAE-Process

Deterministic approach: The training- and working phases follow the same sequence, as shown in figure 2. The preprocessing (DP) prepares the input-data set to data X_i for the input space (X-space) of the dimensionality D, the dimensionality reduction module (DR) embeds the data Y_i in the low dimensional embedding space
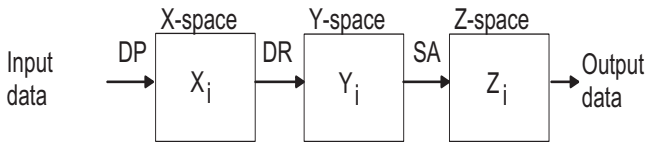
Fig. 2 MAE-processing sequence

(Y-space) of dimensionality d and finally the scaling and adaption module (SA) reproduces the output data Z_i in the feature space (Z-space) of dimensionality d. The MAE-process acts as a universal approximating method, in most cases without iterations and parameter settings.

Modularity: The modular design of the training and working units shown in figure 1 and the flexibility achieved by unsupervised learning with module DR and supervised learning with module SA (see figure 2), enable the system to create a user orientated convincing application, which shows not only hits to a receptive area, but also a degree of membership and the tendency of a measuring point to a trained class. With this contribution, any output curve for coupling to periphery equipment can be realised for display, registration or control and monitoring purposes.

Achievement of global topology: The characteristics and the intrinsic information content of high-dimensional data X_i will be preserved during data preprocessing embedding. The mapping with the dimensionality reduction is done so as to preserve the accuracy of angles and distances. The embedded data Y_i represents the global and local characteristics of the MF. This arrangement exhibits minimum energy content and represents the global optimum.
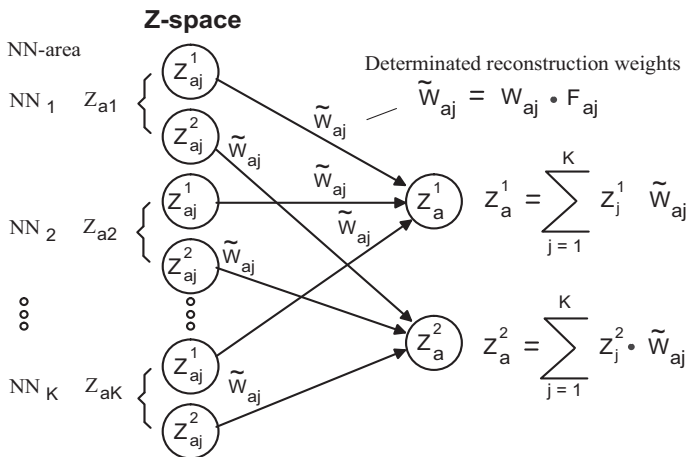
Fig. 3 Reproduction in Z-space

Application-optimised embedding: The method (NOP-method) is using the transformation matrix $\underline{V}$, formed in the training phase, inquiry points X_i are embedded via matrix-vector multiplication in the Y-space and after this scaled and adapted to be projected by using the scale matrix $\underline{F}$ in Z-space. The generalising of new inquiry points X_a can be carried out by the determination of the reconstruction weights for new untrained inquiry points in X-space and direct reproduction in combination with the scale matrix F in Z-space.

3   Processing Modules

Data preprocessing: Before the training-data set is in the form of training vectors, which represent the training points (TP), are entered into the training unit, they must be edited by the data preprocessing unit. All TPs are transferred via Wavelet transformation into a Wavelet transformed spectrum, outliers are removed and the input vectors are formed. The averaging property of the WT reduces noise, while preserving the characteristics of the original signal. The option of selecting only relevant WT-coefficients acts as a data reduction filter and prevents the overtraining of the artificial neural network by using relevant coefficients with a value higher than a predefined threshold. A further benefit of the DP is the elimination of the known problems with discontinuity of data in the MF [8]. A special fitting method is used to smooth the MF of the training-data to present them as input data X_i and to make it treatable for a dimenionality reduction method (NOP-method).

Wavelet-transformation: Preserving euclidian distance, the coefficients $WK_1, ...,$ WK_N are linearly independent by using a orthonormal basis, signals can be specifically modified, irrelevant coefficients can be ignored, coefficients with the most energy represent the main characteristics and approximate the original signal.

Forming the input vector: Because each group l has other relevant coefficients $WK(l) = \{WK_1(l), ..., WK_D(l)\}$, the input vector with the coefficients must be formed by the union set $\mathbf{x}_i := \bigcup_{l=1}^{n} WK(l)$ of the relevant coefficients $1, ..., D$ of all groups $l = 1, ..., n$. With this principle, it is possible to reproduce the TPs of all groups in the input space R^D with X_i. If for example, the WT-coefficients $k = 1, 3, 5, 7$ in group 1 and the coefficients $k = 1, 4, 5, 7, 16$ in group 2 are relevant, the dimension of the input vectors must be 6 to reproduce all vectors by using the coefficients $k = 1, 3, 4, 5, 7$ and 16 in the input space for forming the MF. Input vectors formed with this principle represent all TP, but now comprise unconnected clusters in the input space (X-space).

Forming the MF: Module DE, stage 1 of the NOP-method (Scaled and Adapted Orthogonal Neighborhood Preserving Projections), belongs to the class of spectral methods for dimensionality reduction [3, 2, 1]. To use spectral methods, the training-data set X_i must be smooth and well sampled without discontinuity in the MF. This is the prerequisite for stable numerical calculations. Outliers must be removed to build up a model for representation of the intrinsics of the MF in the input space.

Fitting method: The MF in the input space is described by distances and angles. With the Nash theorem [5], every differentiable MF can be reconstructed in euclidian space R^d. To have a small deviation, the smoothness of the MF in the input space, the maximum gradient from TP to TP is observed. When the gradient exceeds a certain value, additional points will be harmonically fitted to close the gaps and to prevent a too high of curvature of the MF. This fulfills the prerequisite of carrying out a numerically stable dimensionality reduction. The harmonically fitted values cause no significant change of the characteristics of the MF. They change the eigenvectors formed in stage 1 of the training phase little, or ideally, not at all. The smoothing is carried out by a spline interpolation for each dimension by following formula (1) in the intervals from TP to TP. The process is repeated until a predefined criterium is reached. Possible criteria are: distance from TP to TP is shorter than the maximum allowed distance or the curvature of the MF in the vicinity of the observed area is less than the allowed value.

$$S(t) = P_0(2t^3 - 3t^2 + 1) + P_1(-2t^3 + 3t^2) + T_0(t^3 - 2t^2 + t) + T_1(t^3 - t^2) \quad (1)$$

NOP: stands for *neighborhood* preserving *orthogonal projection*. Overlapping areas, defined by NN-relations, studied collectively, provide information about the global structure of the MF, which is preprocessed as above described. All clusters are linked together to one smooth MF without discontinuities and a predefined curvature which guaranties stable numeric operation. This is the prerequisite for the following operations.

Stage 1: unsupervised learning. Decorrelation DE in stage 1 of the process, which is considered during the optimization of the input structure of the MF (NOP-method) and which, in contrast to the known method (ONPP), uses the first two components for embedding (Y-space) [4], consists of five steps.

Step 1. Determination of the NN: K-nearest neighbours are identified in the X-space. The geometry of the MF, or to be exact its maximal curvature, is decisive for determining how many TPs are required to make possible a precise embedding in low dimensionality.

Step 2. Determination of the local Gram matrix $\underline{G}$: it contains the distances between training vector X_i and the NN-matrix $\underline{\eta}_j$. The local Gram matrix $G_{jk} = (\mathbf{x}_i - \eta_j) \cdot (\mathbf{x}_i - \eta_k)$ is symmetrical, positively defined, contains no negative eigenvalues and is thus invertible. The maximum distance of each TP X_i to its NN in X-space (*MaxDist2NN*) is calculated and assigned to each TP_i and stored.

Step 3. Standardization: The reconstruction weights are standardised to $\sum_{j=1}^{K} W_j = 1$. For each point X_i the NN, the number of which is K, and the distance η_j of the NN to inquiry point X_a was determined in the last step. The standardised reconstruction weights for each DP are calculated using equation (2), with $r, l, m \in \{1, ..., K\}$, K is the number of NN and j an index to identify a NN.

$$W_j = \frac{\sum_r G_{jr}^{-1}}{\sum_{lm} G_{lm}^{-1}} \quad (2)$$

Each DP is represented as a weighted linear combination of its NN.

Step 4. Optimization, determine eigenvalues: with the weights W_{ij}, calculated in the X-space the structure of the MF in d-dimensional space is reproduced. In order to do this, the calculated NN-weights for each X_i in the resulting Gram matrix are grouped together. In order to carry out the so called embedding procedure, the following cost function ($\mathbf{y}_i$, $\mathbf{y}_j$ represent data points Y_i, Y_j) with the constraints $\frac{1}{N}\sum_{i,j=1}^{N}\mathbf{y}_i\mathbf{y}_j^T = \underline{I}$ and $\underline{Y} = \underline{V}^T \cdot \underline{X}$ with $\underline{V}^T\underline{V} = \underline{I}$ must be minimised:

$$\Phi(y) = \sum_{i=1}^{N}|\mathbf{y}_i - \sum_{j=1}^{K}W_{ij}\mathbf{y}_j|^2 \tag{3}$$

Through the multiplicative composition with $\underline{X}$ and the requirement of orthonormality of the columns from $\underline{V}$ with $\underline{V}^T\underline{V} = \underline{I}$, the target function is further constrained. This leads to the following convex optimization problem[4]:

$$\tilde{\underline{M}} = \underline{X}(I-W)^T(I-W)\underline{X}^T = \underline{XMX}^T \tag{4}$$

In this equation, the quadratic form of the cost function (3) represents the global minimum. The resulting matrix $\underline{M}$ (Gram-matrix) is sparse. The calculation of its eigenvalues solves the optimization problem and requires less work than methods with full matrices. This decorrelation, which is essentially a factorization, can be carried out with standard software.

Step 5. Reproduction in Y-space: embedding in the low dimensional Y-space of dimension d is not carried out through reproduction using the linear combination $\mathbf{y}_i = \sum_{j=1}^{K}W_{ij}\mathbf{y}_j$, but through the projection with the relationship (5). The embedding of input vectors X_i in the low dimensional space is thus made possible with a matrix vector operation (5).

$$\mathbf{y} = \underline{V}^T \cdot \mathbf{x} \text{ and for all training vectors with } \underline{Y} = \underline{V}^T \cdot \underline{X} \tag{5}$$

For the statement of the problem, the reproduction of the main components as a two-dimensional model is sufficient.

$$Z_a^m = \sum_{j=1}^{K}Z_{aj}^m \cdot \tilde{W}_{aj} \tag{6}$$

The reproduction errors, caused by the DR are minimised through the special optimization with NOP in such a way, that NN-relations of the TP are maintained through the transformation and so the basis for direct embedding of the inquiry points X_a using linear combination in the feature space (Z-space) is created. Due to the special constraints of the NOP-method, the global structure of the training-data set is preserved. A further result is a special matrix $\underline{V}$, determined by the intrinsic information content of the MF, with which new untrained X_a can be embedded using vector multiplication. The resulting data arrangement in Y-space is free of redundancy and has minimal energy content.

Stage 2: supervised learning. In stage 2 the embedded TPs Y_i are rescaled in such a way, that the data, mapped in the feature space (Z-space), are adapted to the

required data structure for the respective purpose. The data structure, resulting from this stage, can consist of one or several arbitrarily continual curves, the specified progression of which reflect the intrinsic information content of the trained MF, or of ranges, which represent certain clusters and which are thus suitable for classification purposes (see figure 4). In contrast to other classification methods, all clusters are contained in the training pattern. Thus it is possible in the working phase to determine which cluster a new unknown representative X_a belongs to, in one step. In the event no match results, a display is made showing the degree of membership of the new representative.

$$F_i^m = \frac{Z_i^m}{Y_i^m}, Z_i^m = F_i^m \cdot Y_i^m \tag{7}$$

For this purpose $N \cdot d$ neurons are required at the entrance and $N \cdot d$ neurons at the exit of the ANN. Every entrance neuron Y_i^m (m indicates the dimensionality) is connected with every exit neuron Z_i^m through the relationship (7).

Receptive area of the clusters: Due to the standardization of the sum of the distances to 1, which was carried out for each X_i in the training phase with equation (2) during the calculation of the reconstruction weights, the information about the absolut distances is lost. For this reason, the closeness, the degree of membership to the receptive area of a cluster must be determined to reconstruct the ratio of distance in feature space. The distance of the most far NN to a inquiry point is necessary to realize a precise projection and this NN-point must be in the receptive area of the according cluster. If one NN-point is outlying of the receptive area, the projection will be not precise. For this reason, it is necessary to determine *ReceptiveAreaFact* with equation (8). The numerator is 2 for the diameter of the n-dimensional sphere round a point X_i in a cluster. The expression SW_{opt} in the denominator is the optimal threshold for the decision, if a point is a member of a class or not and is measured by a ROC-analysis (receiver operating characteristic) for each group in the training phase.

$$ReceptiveAreaFact = \frac{2}{SWopt} \tag{8}$$

In contrast to other classification methods, the degree of membership to the receptive area of a group can be determined with equation (9) for every inquiry point X_a in the working phase, if *MaxDistXa2NN* of inquiry point X_a is greater than *MaxDistXi2NN* x ReceptiveAreaFact, otherwise the degree of membership is one. For this, the maximum distance of each TP X_i to its NN in X-space (*MaxDist2NN*) is calculated in step 2 of stage 1 of the training phase during the calculation of the NN-weights and is assigned to the TPs and stored. This distance, defining the receptive area, is a function of the variance of the TPs. In (9) *MaxDist2NN* is the maximum distance from X_i to NN in the training phase and *MaxDistXa2NN* is the distance of the inquiry point X_a to NN in the working phase during classification.

$$DegreeofMembership = \frac{ReceptiveAreaFact * MaxDist2NN}{MaxDistXa2NN} \tag{9}$$

4 Functioning

Training phase: 1. order input data from smallest to largest distance, 2. WT-transformation of the X_i, 3. selection of relevant coefficients, 4. formation of the MF, 5. smoothing of the MF with the fitting method, 6. dimensionality reduction (unsupervised learning), formation of matrix $\underline{V}$, embedding in the Y-space, 7. scaling and adaptation (supervised learning), mapping in Z-space, 8. mapping in cluster forms with degrees of membership. The training phase is devided into the modules DP (1,..., 5), DR (6) und SA (7, 8).

Working phase: 11. WT-transformation of the input data, 12. selection of the relevant coefficients according to the criteria from the training phase, 13. formation of the input vector, 14. transformation, 15. reproduction and embedding. The working phase is devided into the modules DP (11..13), DR (14) and SA(15).
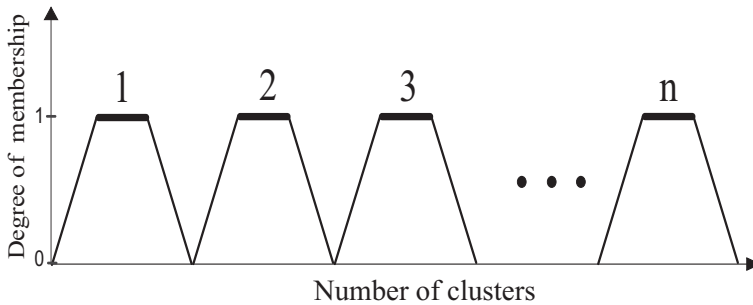
Fig. 4 Z-space: Cluster-orientated depiction in feature space with degree of membership 0...1
.

5 State of Technology

A process which transforms input data into a low dimensional space using an unsupervised learning procedure in such a way, that this arrangement displays topology-preserving properties, both locally and globally, and subsequently processes these directly into output curves for display, registration or control and classification purposes, which shows not only hits to a receptive area, but also a degree of membership and the tendency of a measuring point to a trained class using a supervised training procedure as has been introduced here, is hitherto unknown. Processes which optimise an ANN iteratively suffer under local minima (e.g. backpropagation algorithm) and do not display a minimum number of neurons. Genetic algorithms are indeed capable of finding the global optimum, however, such an adaption process takes longer than the process presented here and does not work in an optimal manner with respect to the storage base and calculation time, due to the redundant number of neurons and hidden layers. Other processes for dimensionality reduction require smooth, well-sampled input data, display only local preservation of topology (LLE) or global preservation of topology (ISOMAP), and their output configurations are

not directly usable for display, registration or control and classification purposes. With the support vector machines (SVM), which are often used for classification and are especially suitable for processing large amounts of data, as well as for classification in the working phase, only the so called support vectors (SV) must be employed. They display a global optimum and possess good generalisation properties. However, only one class can be processed with one set of SVs. The search for parameters and the determination of the kernel function is complicated and tends to be very time intensive in practice, particulary so, when different classes are to be assigned.

References

1. Bengio, Y., et al.: Spectral Dimensionality Reduction, Universit de Montreal. Internet (2006), `http://www.iro.umontreal.ca/bengioy`
2. Brand, M.: Minimax embeddings Mitsubishi Electric Research Labs, Cambridge MA 02139 USA. Internet (2006), `http://www.merl.com`
3. Brand, M.: A unifying theorem for spectral embedding and clustering, Mitsubishi Electric Research Labs, Cambridge MA 02139 USA. Internet (2006), `http://www.merl.com`
4. Kokiopoulu, E., Saad, Y.: Orthogonal Neighborhood Preserving Projections, Computer Science and Engineering Department University of Minnesota. Internet (2006), `http://lts1pc19.epfl.ch/repository/Kokiopoulou20051351.pdf`
5. Gnther, M.: Zum Einbettungssatz von J. Nash. Mathematische Nachrichten 44(1), S165–S187 (1989)
6. Roweis, S., Saul, L.: Nonlinear dimensionality Reduction by locally linear embedding. Science 290, 2323–2326 (2000)
7. Saul, L.K., Roweis, S.T.: Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds. Journal of Machine Learning Research 4, 119–155 (2003)
8. Tsai, F.S., Wu, Y., Chan, K.L.: Nonlinear Dimensionality Reduction Techniques and their Applications. EEE Research Bulletin, S52–S53 (2004)

Self-organizing Shortcuts in an Overlay Network

Lada-on Lertsuwanakul and Herwig Unger

Abstract. A decentralized algorithm for self-organize shortcuts relying on traffic in an overlay network is proposed. The idea is motivated by a bridge across the traffic jam junction in Bangkok. This method aims to improve routing performance and increase transsimission reliability in dynamic environment. The temperature field is used to present buffer usage level, so hotter peer implies more overloaded. Two groups of agents work cooperatively to construct shortcuts, called *"Bridge"*, across high buffer-usage peers. The useless bridges are removed when those shortcuts have not been used. The performance of proposed method is analyzed in the P2PNetSim, a powerful network simulator. The experimental results sustain the proposed idea that the self-organizing bridges give less routing time and loss ratio. Besides, a self-organizing grid show better load balancing in a distributed overlay system.

1 Introduction

A Peer-to-peer (P2P) system is a virtual community overlaying the basic IP network. All peers are attached to the network for the purpose of resource sharing and cooperation among peers. The P2P overlay structure is primarily organized for various specific features such as robust routing architecture, redundant storage, efficient searching and routing, load-balancing, and distributed implementation of trust and authentication [1],[2],[3],[4],[5]. The P2P overlay approaches are divided into structured and unstructured networks [6]. The structured ones are tightly controlled and content are placed at a specific location then it can find resources efficiently. However, they may cause problems if the network changes quickly and has to be reorganized permanently. On the other hand, the unstructured P2P overlay networks organize peers in a graph randomly or hierarchically. It uses flooding on the flat

Lada-on Lertsuwanakul · Herwig Unger
Department of Communication Networks, Fernuniversitaet in Hagen,
Universitaetsstrasse 27, 58084 Hagen, Germany
e-mail: {lada-on.lertsuwanakul,herwig.unger}@fernuni-hagen.de

network to find contents, so such a mechanism is not scalable and generating un-expected loads of network. The combination of advantages from both types must provide an effecitve system. Berg et. al. [7],[8] introduced decentralized algorithms that organize the participants of a network into a two- to *n*-dimensional grid with a Cartesian coordinate-system is an example. The grid structure could be constructed in a distributed manner. Such form provides multiple routes between any two nodes which beneficially and efficiently support routing and resource lookup processes. Besides, lots of literature show that the delivery time in a gird network with long links is limited [9],[10],[11],[12].

Routing, the determination of a path to transmit data between source and desti-nation, is one of the key function of all computer networks. When several paths are available particularly in any large network, the complexity of routing is to choose the "best" path in a multi-dimensional perspective. Object to clarify the best or an optimal path, the Quality of service (QoS) is arisen as a measurement to guarantee on the ability of a network to deliver predictable results. The complexity in QoS routing comes from multiple criteria, which often make the routing problems in-tractable. Current P2P situation motivates to introduce novel algorithms objecting to improve the quality of service of overlay systems considering underlay network parameters and overlay information, as well as make benefit from virtual structure.

The proposed method provides a self-organizing shortcuts, called *"Bridges"*, in an overlay system depending on dynamic network's traffic. The objective is to im-prove QoS routing. The expected results are to decrease routing time and increase transmission reliability by reducing loss ratio and delayed time due to overloaded buffers. The thermal field is used to represent buffer usage level [13], [14]. The tem-perature represents peer's buffer used-space. A higher temperature indicates less resources available and less capable of handling new data. Then hotter node shows less buffer free-space. The self-organing grid algorithm starts when a node heats up to a predefined value. Two types of agent are launched and work cooperatively to build bridges overpass traffic peers. The first type of agent is used to find appropri-ated nodes to create virtual links, and the second one are worked for distributing information for building bridges. However, some links are built but seldom used then they should not be maintained in the system. This proposed work assigns a counter to every bridge for couting unused time. And unneccessary links are re-moved when counter reaches in a predefined time. The simulation is performed in a network simulation tool, P2PNetSim. The messages are routed greedily comparing between a static grid network and a self-organizing grid network. The measurements are average routing time, delay time, and delivery ratio. In addition, load balancing is examined and discussed.

This paper is organized as follows: Section 2, previous related literatured are reviewed. Section 3 introduces the algorithm and describes how it works. Then net-work simulation and experimental result discussions are given in Section 4. Finally, the conclusions and an outlook on future research are presented in Section 5.

2   Related Works

2.1   *Routing in Mesh Overlay Networks*

Mesh topologies have been used in many areas of communication networks. The mesh structure is reliable and offers redundancy giving benefit to routing [15],[16]. In 2000, John Kleinberg [10] introduced a set of small-world network models building on the models of Watts and Strogatz [17], and proofed the one model of the set is a decentralized algorithm which is able to find a shortest path within a finite number of steps. His models base on a two-dimensional grid $(n \times n)$ in which each peer has undirected local links connect to its neighbours, and a directed long link randomly generated [10]. A long distance link is constructed from node u to random endpoint *v* with a probability proportional to $d(u,v)^{-2}$, the inverse square of the lattice distance of *u* and *v*. Links have a non-uniform distribution that prefer closer nodes than distant ones, according to power-laws. In such structure Kleinberg shows that with a greedy algorithm and using only local information, it is able to find the route of any two nodes in $O(log^2 n)$.

   Recently, some literature analyzed and extended Kleinberg's model, not only on routing algorithms but also in model construction process. Martel and Nguyen [9] expanded small-worlds to a *k*-dimensional model based on diameters of random graphs focused on uniformly distributed arcs. They introduced short paths with an expected length $O(log^{3/2} n)$ in 2-dimensional model and $O(log^{1+1/k} n)$ in the general *k*-dimensional model $(k \geq 1)$ by adding global knowledge of random links. Naor and Wieder [12] presented better delivery time with greedy algorithms by selecting the neighbour of decision node's neighbours instead. And Zou et al. [11] claimed that Kleinberg's model needs to use global information to form the structure, so that they proposed to use a cached replacing strategy instead of fix long distance links to reform the shortcuts when receiving new query requests.

2.2   *Thermal Approach*

In 2004, Unger and Wulff [18] have been introduced a routing approach in analogy to temperature fields in thermal physics to locate nodes managing contents of common interest in P2P networks. Each node features a temperature, which is an index for the activity of that node. The heat of each node radiates towards its direct neighbors and therefore influences their temperature as well. Whenever the content of a node is accessed or updated, its temperature is increased, whereas during periods of inactivity, the temperature falls of exponentially to align with the temperatures of the surrounding neighbors.

   In 2007, Baumann et al. [19] introduced the HEAT routing algorithm for large multi-hop wireless mesh networks to increase routing performance. HEAT uses anycasts instead of unicasts to make better use of the underlying wireless network,

which uses anycasts by design. HEAT relies on a temperature field to route data packets towards the Internet gateways. Every node is assigned a temperature value, and packets are routed along increasing temperature values until they reach any of the Internet gateways, which are modeled as heat sources. It is a distributed protocol to establish such temperature fields which does not require flooding of control messages. Rather, every node in the network determines its temperature considering only the temperature of its direct neighbors, which renders our protocol particularly scalable to the network size.

The thermal field represented buffer usage level has been introduced since 2009 [13], [14]. It is used to find the optimal path in a distributed system. When neighbor peer has a high temperature meaning it congests, the message-holder can decide the best route. The optimal path is balance of avoiding long queues and shortest way.

3   Algorithms

The idea of the bridge algorithm motivated by a bridge across the junction in Bangkok known as one of the heaviest traffic cites in the world as shown in Fig.1. The initial work is presented [20]. A bridge working as shortcut helps the traffic so that cars drive faster across the jam junction, also reducing traffic on the normal road. Compared to the communication system in which a traffic peer (high buffer utilization) leads to loss of messages and increase delay times. By this consequence, the bridge idea could be applied by generating logical rotes cross the congested network areas. The different to the real bridge is this virtual links constructed temporarily. When a link is unnecessary, it is removed from the system.

Fig. 1 The bridge across traffic junction in Bangkok motivates the proposed algorithm

An n-dimensional grid structured network is established. An example, the two-dimensional grid network is identified with a set of lattice, a $m \times n$ square, (i,j) : $i \in 1,2,\cdots,m,\ j \in 1,2,\cdots,n$. The lattice distance between two nodes (i,j) and (k,l) is $d((i,j),(k,l)) = |k-i| + |l-j|$. A node in the community has four bi-directed edges to its direct neighbors. The algorithm is run without centralized control. The temperature (θ) indicates the usage level of a peer's incoming- and outgoing message buffer. The temperature of a node c is referred as θ_c. The possible values of θ_c are in the range from 0 to 1, where 0 denotes an empty buffer and a value of 1 indicates that the buffer is full. The temperature is calculated as shown in Eq.1.

$$\theta_c = \frac{\text{Message in buffer}}{\text{Buffer size}}, \quad 0 \le \theta_c \le 1 \tag{1}$$

To simplify presentation, each node only uses a buffer, which is organized in a FIFO manner in our work. Hence, the temperature of that buffer is equal to the temperature of the node. Since the adaptation of grid strongly depends on θ_c being up to date, the temperature is recalculated when messages that enters or leaves a buffer.

(a) Traffic peer u find suitable nodes for generating bridges

(b) Jumper nodes create shortcuts to others

(c) Many bridges are built over a congestion area
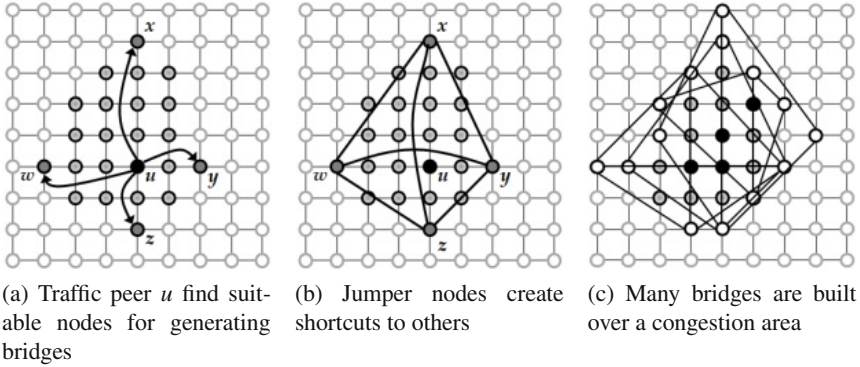
Fig. 2 The construction part of Self-organizing Bridge Algorithm

The self-organizing Bridges algorithm start when a peer constantly has high value of temperature field in a certain period. The *Hot Peer* is defined when the temperature (θ) is continued higher than a heat level (θ_H) within a time (δ_t). And *Jumper Peer*, a sutiable location to bulid bridges, has the temperature lower than a cool level (θ_L). Two groups of agents work cooperatively in the construction part of algorithm. First agent, called *Seeker*, is activated when the buffer is continuously high reached the heat point. An example is shown in Fig.2a, node u is heat constantly up to the heat level $(\theta_c \ge \theta_H)$ then four seekers are generated. The seekers are forwarded to four directions until they find jumper nodes w, x, y, and z when $\theta_w, \theta_x, \theta_y, \theta_z \le \theta_L$. The seekers die either when they find a jumper node or their life time exceeds predefined TTL value. After that, nodes w, x, y, and z generate acknowledgments and send back to the original heat node u to inform route and jumper node details.

Second agents, called *Builder*, are launched when node u received acknowledgements from four seekers within double seeker life time. Otherwise, the heat node resends a seeker to the missing seeker direction. The builder agent responses to distribute jumper details to other nodes for creating bridges over heat node u. The diagram in Fig.2(b) is a simulated diagram when logical shortcuts are created. In the final step nodes w, x, y, and z send acknowledgments back to original heat node u to confirm that the bridge creation process is completed. The network diagram in Fig.2(c) shows scenario in which many heat nodes exist.

When new shortcuts are created, the peer is also built an unused counter. Once the unused counter counts continuously to a certained period, (Δt), then that link is removed from the system.

4   Experiments

To analyze the performance of the proposed algorithm, this section discusses the experiments comparing routing performance and load balancing of a static grid structure to a self-organizing grid structure.

4.1   Simulation Setup

The simulation is performed in a network simulator, P2PNetSim. This tool allows the simulation of large distributed networks. The peers can be configured collectively and individually using XML based configuration. The peer behaviour is implemented in the Java programming language.

Network Parameter- the networks are organized into two-dimensional grid structures, each composed of 2,500 nodes (50x50). Nodes are connected to their neighbors in all four directions. The coordinate of a node is serves as its ID. The grids overlay a simulated IPv4 network. The buffer sizes and outgoing bandwidths are limited for all the peers which randoly defined by a power-law distribution. There are three types of messages; data packet, algorithm agents and acknowledgements used in experiments. The system handles both data packets and algorithm agents in First-In-First-Out (FIFO) manner, and keeps them in the same buffer while the acknowledgements are handled with priority.

Traffic pattern - traffic is generated randomly by all network nodes. The sending probabilities and intensities are distributed exponentially for both a source node generates, as well as the number of messages that can be sent per simulation time-step. All simulations generate 500,000 messages in total. The routing algorithm uses the greedy method to forward data messages to its target. Messages are routed to the closet neighbour of message-holder node compare to message's destination. The distance is measured by Euclidean Distance from node's ID.

Self-organizing Bridges Parameter- four parameters are defined; hot value, hot period, suitable thermal value, and unused period. The hot value represents a minimum level of congestion node. The hot period represents a duration of a node is constantly hot. The suitable thermal value shows a node that suited to create shortcuts. And unused period identfy a time that shortcut has not been used. (θ_H =0.7, δt =10 time-steps, θ_L =0.1, and Δt =20 time-steps).

Performance measurement - the metrics used to measure the performance using different decision methods are the following;

- loss and success ratios
- average hop-count, delay time, and routing time
- number of overloaded nodes ($\theta \geq 0.7$)
- number of bridges (shortcuts).

4.2 Simulation Results

To proof the algorithm's performance and to study the impact of bridges; loss and success ratios, number of bridges, and average routing time was measured. Furthermore, load balancing performance was evaluated in terms of number of overloaded nodes in the system. The self-organizing bridge algorithm started at simulation time-step 50.

(a) Compare Delivery Ratio   (b) Compare Routing Time
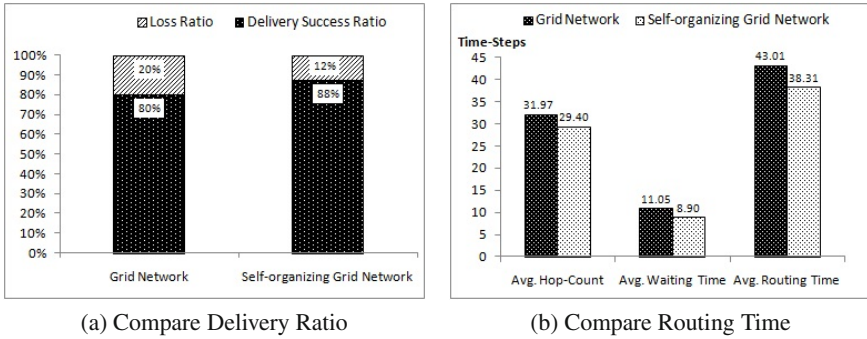
Fig. 3 The comparison of delivery ratio and routing performance

First, the delivery success ratio and routing perofrmance were analyzed as presented in Fig.3. In Fig.3a, the transmission success ratio of self-organizing grid shows better than static grid by 8%. The routing performance analyzed in Fig.3b, the comparison of average hop-count, average waiting time, and avaerage routing time are presetned. Number of hop-count, a number of nodes passed, from self-organizing grid shows superior result than static grid by 2.5 time-steps and average

delay time, a summary of waiting time in queue on the way from source to target, decreases from 11 time-steps to 9 time-steps. Lastly, the average routing time counted from source until reached target decreases from 43 time-steps to 38 time-steps. The bridges or virtual links are built across overloaded peers cause more packets reach their destination and reduce number of long-queue peers.
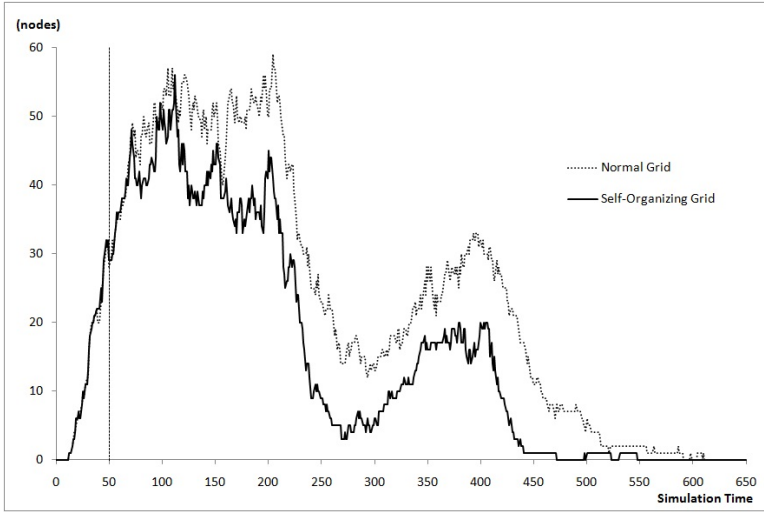
Fig. 4 Compare number of hot peers

Next, load balancing results of grid networks are compared in Fig.4. The network traffic intensity is presented by the number of hot nodes at every simulation time step. Before self-organizing grid algorithm starts, number of hot nodes from both networks are the same. Once self-organizing algorithm is run, number of overloaded node is notably decrease in self-organizing grid. The bridges makes traffic more balance load.

Fig.5 and Fig.6 present network diagram of static grid network and self-organizing grid network respectively at simulation time 50, 100, 200, 300, 400, and 500. The dark spots represents high temperature. The darker ones imply higher temperature field. At simulation time 50, both networks have similar dark points. At simulation time 100, both networks are slightly different and clearly distinguishable at simulation time 200. At simulation time 500, no black dot remains in self-organizing grid while static grid network has some.

The self-organizing grid network is analyzed more in details. Fig.7 presents number of bridges by simulation time. The graph rises from simulation time 64 when first group of bridges was created. Number of bridges increases and decreases relying on number of hot nodes supported by Fig.8. When number of overloaded node
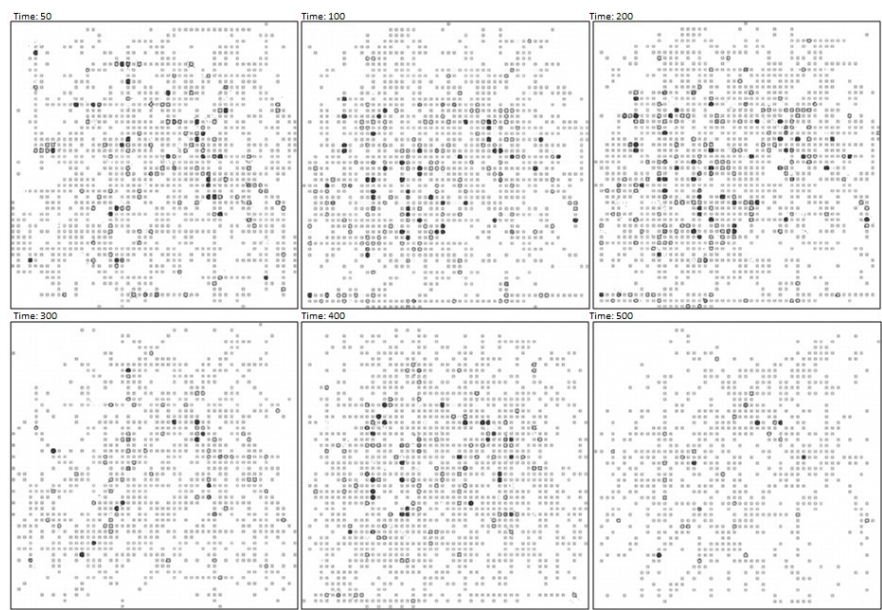
Fig. 5 Buffer level of static grid network diagram of 2,500 peers (50x50)
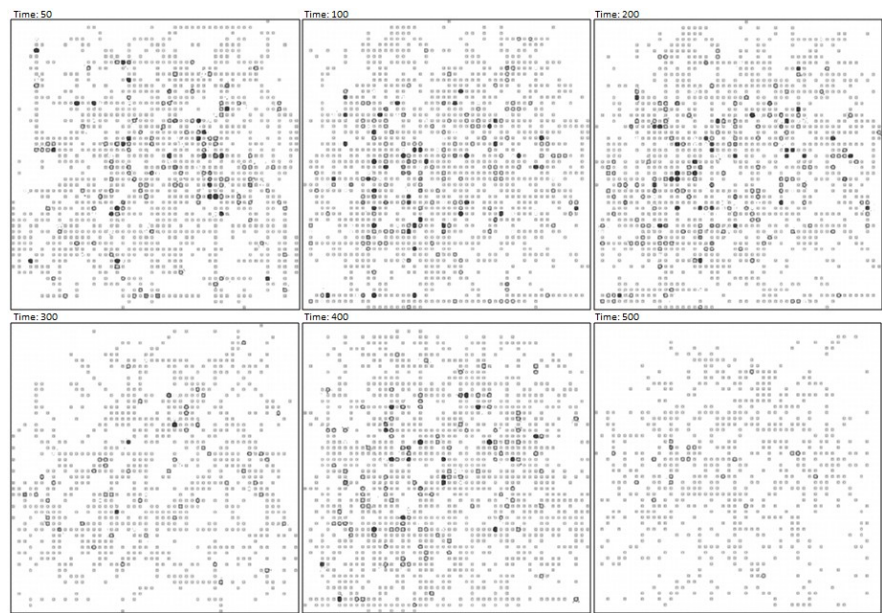
Fig. 6 Buffer level of self-organizing grid network diagra of 2,500 peers (50x50)
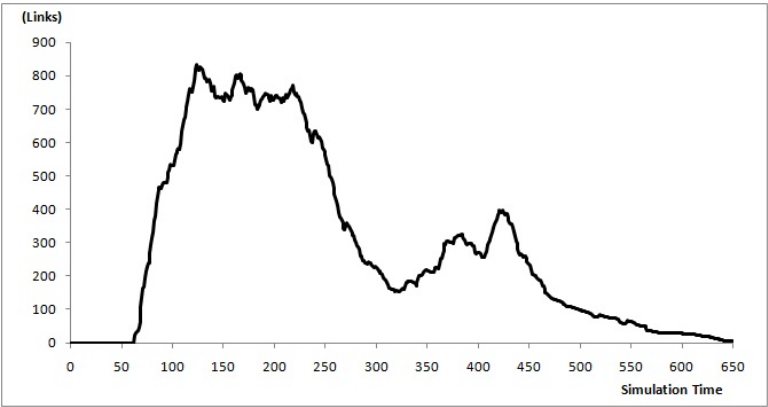
Fig. 7 Number of Bridges
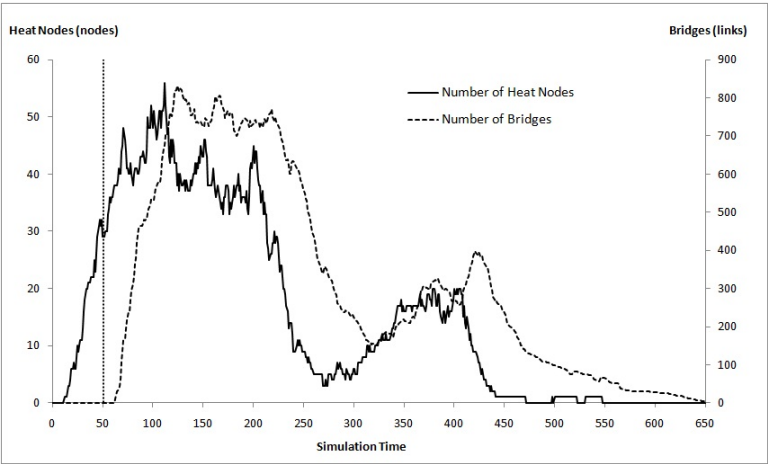
Fig. 8 Number of hot peers VS number of bridges in self-organzing grid network

is high making number of bridges rises accordingly. Whereas, number of bridges went down after number of hot nodes decreased. An example of self-organizing grid network is shown in Fig.9a which is captured at simulation time 200. Two parts of network are enlarged presenting how self-organizing grid looks like, shown in Fig.9b and c. Fig.9b presents completed bridges while Fig.9c shows incompleted and overlapped ones.
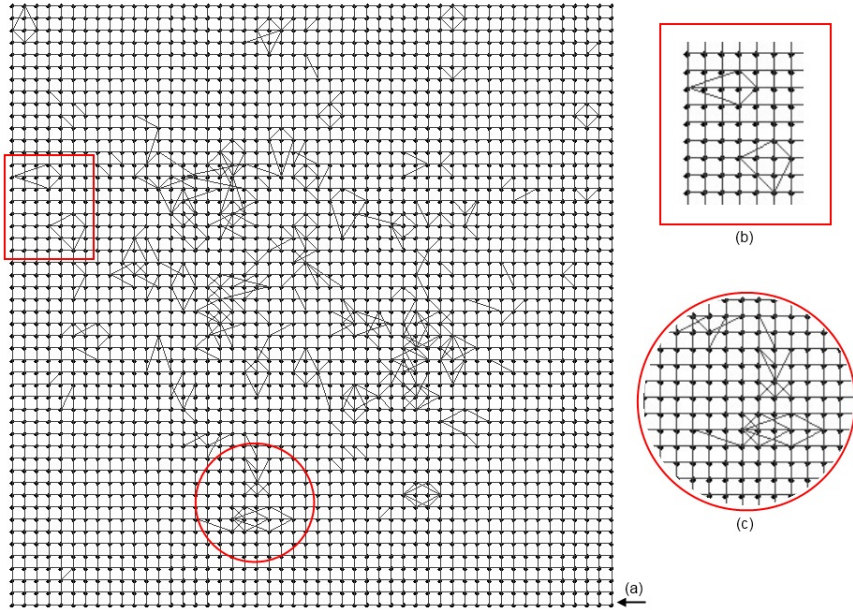
Fig. 9 Self-organizing grid network diagram at simulation time 200

5 Conclusion

This paper presents an algorithm that automatically generates shortcuts overpass overloaded areas in a grid-like overlay network. The proposed idea is motivated by bridges across the traffic junction in helping to reduce seriously traffic-loaded intersections. The congestion peers which had high buffer utilization are represented by temperature using the Thermal Field Approach. Two groups of agents cooperatively work to find suitable peers and build the bridges. The needless link is removed when it has not been used. The algorithm was testified in P2PNetSim. The experimental results have proved that the introduced algorithm is efficient. Success ratio increases, as well as average routing time decreases according to the reduction of delay time and number of hop-count. Moreover, the level of high buffer usage nodes decreases after bridges are generated. Besides, the average buffer utilization are reduced leading to a better load balancing on the systems. The virtual shortcuts are self-maintained according to traffic of networks.

Future works, the algorithm will be enhanced to establish shortcuts for a group of peers. The cooperation methods should be considered such as centralized strategy or decentralized strategy. The centralized strategy has a leader to construct and maintain bridges for the group. In contrast, all information of peers in the group is shared to find the best soultion organizing briges for group's benefits in decetralized strategy.

References

1. Rowstron, A., Druschel, P.: Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In: Proc. of the Middleware (2001)
2. Stoica, I., Morris, R., Karger, D., Kaashoek, M.F., Balakrishnan, H.: Chord: a scalable peer-to-peer lookup protocol for internet applications. IEEE/ACM Transactions on Networking (11), 17–32 (2003)
3. Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A scalable content addressable network. In: Proc. of the ACM SIGCOMM, pp. 171–172 (2001)
4. Andreolini, M., Lancellotti, R.: A flexible and robust lookup algorithm for P2P systems. In: IEEE International Symposium on Parallel & Distributed Processing, pp. 1–8 (2009)
5. Gnutella Protocol Development v0.6 (2002),
   `http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html`
6. Lua, E.K., Crowcroft, J., Pias, M., Scharma, R., Lim, S.: A Survey and Comparison of Peer-to-Peer Overlay Network Scheme. IEEE Communications Survey and Tutorial (2004)
7. Berg, D., Sukjit, P., Unger, H.: Grid Generation in Decentralized Systems. In: Proc. 2nd Int. Workshop on Nonlinear Dynamics and Synchronization 2009, Austria, pp. 95–99 (2009)
8. Sukjit, P., Berg, D.: A New, Fully Decentralized Grid Generation Method. In: Proc. of 5th National Conference on Computing and Information Technology, Bangkok, Thailand, pp. 24–28 (2009)
9. Martel, C., Nguyen, V.: Analyzing Kleinberg's (and other) Small-world models. In: Proc. 23rd ACM Symposium on Principles of Distributed Computing, Canada, pp. 179–188 (2004)
10. Kleinberg, J.: The small-world phenomenon: an algorithm perspective. In: Proc. of the 32nd Annual ACM Symposium on theory of Computing, New York, pp. 163–170 (2000)
11. Zou, F., Li, Y., Zhang, L., Ma, F.-Y., Li, M.: A Novel Approach for Constructing Small World in Structured P2P Systems. In: Jin, H., Pan, Y., Xiao, N., Sun, J. (eds.) GCC 2004. LNCS, vol. 3251, pp. 807–810. Springer, Heidelberg (2004)
12. Naor, M., Wieder, U.: Know Thy Neighbor's Neighbor: Better Routing for Skip-Graphs and Small Worlds. In: Voelker, G.M., Shenker, S. (eds.) IPTPS 2004. LNCS, vol. 3279, pp. 269–277. Springer, Heidelberg (2005)
13. Lertsuwanakul, L., Tuammee, S., Unger, H.: Routing with Temperature Field in Mesh Overlay Network. In: Proc. 9th Int. Conf. on Networks, France, pp. 285–290 (2010)
14. Lertsuwanakul, L., Unger, H.: An Adaptive Policy Routing with Thermal Field Approach. In: Proc. 9th Innovative Internet Community Systems, Germany, pp. 169–179 (2009)
15. Lee, A., Ward, P.A.S.: A Study of Routing Algorithms in Wireless Mesh Networks. In: ATNAC (December 2004)
16. Mello, A.V., Ost, L.C., Moraes, F.G., Calazans, N.L.: Evaluation of Routing Algorithms on Mesh Based NoCs, Technical Report Series No.040 (May 2004)
17. Watt, D., Strogatz, S.: Collective dynamics of small world network. Nature (393), 440–442 (1998)
18. Unger, H., Wulff, M.: Search in Communities: An Approach Derived from the Physic Analogue of Thermal Fields. In: Ramos, F.F., Unger, H., Larios, V. (eds.) ISSADS 2004. LNCS, vol. 3061, pp. 186–196. Springer, Heidelberg (2004)
19. Baumann, R., Heimlicher, S., Lenders, V., May, M.: HEAT: Scalable Routing in Wireless Mesh Networks Using Temperature Fields. In: IEEE Symposium on a World of Wireless, Mobile and Multimedia Networks (2007)

20. Lertsuwanakul, L., Unger, H., Mututanon, N., Kajadsarapadpai, P.: Bridging Overloaded Nodes in an Overlay Network by Long Distance Links. In: Proc. in 30th IASTED Conference on Modelling, Identification, and Control (AsiaMIC 2010), Phuket, Thaialnd (2010)
21. Coltzau, H.: Specification and Implementation of Parallel P2P Network Simulation Environment, Diploma Thesis, University of Rostock (2006)

Rough Sets Based Incremental Rule Acquisition in Set-Valued Information Systems

Junbo Zhang, Tianrui Li, and Da Ruan

Abstract. Set-valued information systems may evolve over time. How to acquire rules from updating information systems is vital in decision making. Rule acquisition from set-valued decision information systems needs both accuracy and coverage. To fast compute and update the accuracy and coverage, the tolerance matrix (relation matrix) and decision matrix are given when the object set varies with time. A case study on the incremental approach validates the feasibility of the proposed algorithm.

1 Introduction

Rough set theory (RST), proposed by Pawlak [10, 11, 12, 13], is a powerful mathematical tool for analyzing various types of data. It can be used in an attribute value representation model to describe the dependencies among attributes, evaluate the significance of attributes and derive decision rules [2, 5, 14, 16, 17].

An information system is a quadruple $S = (U, A, V, f)$, where U is a non-empty finite set of objects; A is a non-empty finite set of attributes; $V = \bigcup_{a \in A} V_a$ and V_a is a domain of attribute a; $f : U \times A \to V$ is an information function such that $f(x, a) \in V_a$

Junbo Zhang · Tianrui Li
School of Information Science and Technology, Southwest Jiaotong University,
Chengdu 610031, China
e-mail: JunboZhang86@163.com, jbzhang@my.swjtu.edu.cn,
     trli@swjtu.edu.cn

Da Ruan
Belgian Nuclear Research Centre (SCK•CEN), Boeretang 200, 2400 Mol, Belgium
e-mail: druan@sckcen.be

Department of Applied Mathematics & Computer Science, Ghent University,
9000 Gent, Belgium
e-mail: da.ruan@ugent.be

for every $x \in U$, $a \in A$. In many practical issues, it happens that some of the attribute values for an object are set-valued, which are always used to characterize uncertain information and missing information in information systems [18]. For an information system $S = (U, A, V, f)$, if each attribute has a unique attribute value, then S with $V = \bigcup_{a \in A} V_a$ is called a single-valued information system; if an information system is not a single-valued information system, it is called a set-valued (multi-valued) information system [1, 15]. For example, incomplete information systems can be regarded as a special case of set-valued information systems, in which all missing values can be represented by the set of all possible values of each attribute.

Due to the dynamic characteristics of data collection, the object set evolves over time. It retrains the system from scratch to obtain new knowledge when the information system varies, which is known as a non-incremental approach. However, the non-incremental approach is often very costly or even intractable. Alternatively, it can apply an incremental learning scheme [4, 9]. Nowadays, the incremental learning approaches based on RST mainly focused on these three cases: (1) The object set in the information system evolves over time while the attribute set remains constant; (2) The attribute set in the information system evolves over time while the object set remains constant; (3) The attribute values in the information system evolve over time while the object set and the attribute set remain constant. In this paper, we focus on the first case. In this case, Shan and Ziarko presented a discernibility-matrix based incremental methodology to find all maximally generalized rules [16]. Zheng et al. developed a rough set and rule tree based incremental knowledge acquisition algorithm RRIA, which can learn new knowledge more quickly [20]. Liu et al. proposed an incremental model and approach as well as its algorithm for inducing interesting knowledge [6]. Furthermore, in business intelligent information systems, Liu et al. presented an optimization incremental approach as well as its algorithm for inducing interesting knowledge [5]. Zhang et al. proposed a method for dynamic data mining based on neighborhood rough sets [17].

However, most incremental approaches focus on single-valued information systems instead of set-valued information systems. In this paper, we study the incremental approaches in the set-valued information systems. Followed by the work in [6], we first redefine the accuracy and coverage under the tolerance relation in the set-valued information system and present a concept of interesting knowledge under both accuracy and coverage. We propose the tolerance matrix and decision matrix to compute the accuracy and the coverage of the generated rules. We then develop the incremental approach and its corresponding algorithm to acquire rules incrementally under the variation of the object set.

The remainder of this paper is organized as follows. Section 2 introduces basic concepts of RST in the set-valued information system and proposes a concept of interesting knowledge under both accuracy and coverage. Section 3 presents an incremental approach and its corresponding algorithm for rule acquisition. Some illustrative examples are conducted in Section 4. The paper ends with conclusions and future work in Section 5.

2 Preliminaries

Basic concepts, notations and results of information systems as well as their extensions are briefly reviewed in this section [1, 5, 6, 7, 11, 12, 13].

2.1 Set-Valued Information Systems

Let $S = (U, A, V, f)$ be a set-valued information system, where U is a non-empty finite set of objects; A is a non-empty finite set of attributes; V is a set of attributes' values; f is a mapping from $U \times A$ to V such that $f : U \times A \to 2^V$ is a set-valued mapping.

A set-valued decision information system $(U, C \cup \{d\}, V, f)$ is a special case of a set-valued information system, where U is a non-empty finite set of objects; C is a non-empty finite set of condition attributes and d is a decision attribute with $C \cap \{d\} = \emptyset$; $V = V_C \cup V_d$, where V_C is the set of condition attribute values and V_d is the set of decision attribute values; f is a mapping from $U \times (C \cup \{d\})$ to V such that $f : U \times C \to 2^{V_C}$ is a set-valued mapping and $f : U \times \{d\} \to V_d$ is a single-valued mapping.

Definition 1. [19] In the set-valued information system (U, A, V, f), for $b \in A$, the tolerance relation T_b is defined as:

$$T_b = \{(x, y) | f(x, b) \cap f(y, b) \neq \emptyset\}. \tag{1}$$

and for $B \subseteq A$, the tolerance relation T_B is defined as:

$$T_B = \{(x, y) | \forall b \in B, f(x, b) \cap f(y, b) \neq \emptyset\} = \bigcap_{b \in B} T_b. \tag{2}$$

When $(x, y) \in T_B$, we call x and y are indiscernible or x is tolerant with y w.r.t. B. Let $T_B(x) = \{y | y \in U, y T_B x\}$. We call $T_B(x)$ the tolerance class for x w.r.t. T_B.

Definition 2. [7] Let $U = \{x_1, x_2, \ldots, x_n\}$ be a finite universe set, and X be a subset of U. Then the characteristic function $G(X)$ of X assigns 1 to an element that belongs to X and 0 to an element that does not belong to X. Thus X can be represented by an n-tuples $G(X) = (g_1, g_2, \ldots, g_n)^T$ (T denotes the transpose operation), and

$$g_i = \begin{cases} 1, & x_i \in X \\ 0, & x_i \notin X \end{cases} \tag{3}$$

Remark: Here, we denote the subset X of U as an n-column boolean vector $G(X) = (g_1, g_2, \ldots, g_n)^T$. For instance, if $U = \{x_1, x_2, x_3, x_4, x_5, x_6\}$, $X = \{x_1, x_2, x_3, x_4, x_5\}$, then $G(X) = (1, 1, 1, 1, 1, 0)^T$.

Definition 3. [8] Let $B \subseteq A$ and T_B be a tolerance relation on U. $R = (r_{ij})_{n \times n}$ be an $n \times n$ matrix representing T_B, called as a relation matrix, where

$$r_{ij} = \begin{cases} 1, & (x_i, x_j) \in T_B \\ 0, & (x_i, x_j) \notin T_B \end{cases} \tag{4}$$

Corollary 1. [3] *Let* $R = (r_{ij})_{n \times n}$ *be a relation matrix. Then* $r_{ii} = 1$ *and* $r_{ij} = r_{ji}$, $1 \leq i, j \leq n$.

2.2 Knowledge Discovery in Set-Valued Information Systems

Definition 4. Given a set-valued decision information system $S = \{U, C \cup D, V, f\}$ where $C = \{c_1, c_2, \cdots, c_p\}$ and $D = \{d\}$. $\forall x \in U$, we generate a decision rule in the following way: as the predecessor of the decision rule we take the conjunction $\bigwedge_{c \in C} (c = f(x, c))$ and as the successor of the rule we take decision attribute d with value $f(x, d)$. Hence, the constructed decision rule for the object x is of the form:

$$(c_1 = f(x, c_1)) \wedge (c_2 = f(x, c_2)) \wedge \cdots \wedge (c_p = f(x, c_p)) \rightarrow d = f(x, d).$$

Definition 5. Suppose a set-valued decision information system $S = (U, A, V, f)$, $U = \{x_1, x_2, \cdots, x_n\}, A = C \cup D. TC = \{T_C(x_1), T_C(x_2), \cdots, T_C(x_n)\} = \{T_1, T_2, \cdots, T_n\}$ is a union of tolerance classes where $T_i = T_C(x_i)$ is the tolerance class of x_i with respect to T_C; $U/D = \{D_1, D_2, \cdots, D_m\}$ is a partition of decision attributes, where D_j $(j = 1, 2, \cdots, m)$ is a decision class. $\forall T_i \in TC, \forall D_j \in U/D$, the support, accuracy and coverage of $T_i \rightarrow D_j$ are defined respectively as follows:

  Support of $T_i \rightarrow D_j$: $Supp(D_j|T_i) = |T_i \cap D_j|$;
  Accuracy of $T_i \rightarrow D_j$: $Acc(D_j|T_i) = |T_i \cap D_j|/|T_i|$;
  Coverage of $T_i \rightarrow D_j$: $Cov(D_j|T_i) = |T_i \cap D_j|/|D_j|$.
where $| \bullet |$ denotes the cardinality of the set.

Here, we use the matrices to simplify the problem. Hence, the support matrix, the accuracy matrix, the coverage matrix are shown as follows [6].

$$Supp(D|T) = \begin{pmatrix} Supp(D_1|T_1) & Supp(D_2|T_1) & \cdots & Supp(D_m|T_1) \\ Supp(D_1|T_2) & Supp(D_2|T_2) & \cdots & Supp(D_m|T_2) \\ \vdots & \vdots & \vdots & \vdots \\ Supp(D_1|T_n) & Supp(D_2|T_n) & \cdots & Supp(D_m|T_n) \end{pmatrix} \quad (5)$$

$$Acc(D|T) = \begin{pmatrix} Acc(D_1|T_1) & Acc(D_2|T_1) & \cdots & Acc(D_m|T_1) \\ Acc(D_1|T_2) & Acc(D_2|T_2) & \cdots & Acc(D_m|T_2) \\ \vdots & \vdots & \vdots & \vdots \\ Acc(D_1|T_n) & Acc(D_2|T_n) & \cdots & Acc(D_m|T_n) \end{pmatrix} \quad (6)$$

$$Cov(D|T) = \begin{pmatrix} Cov(D_1|T_1) & Cov(D_2|T_1) & \cdots & Cov(D_m|T_1) \\ Cov(D_1|T_2) & Cov(D_2|T_2) & \cdots & Cov(D_m|T_2) \\ \vdots & \vdots & \vdots & \vdots \\ Cov(D_1|T_n) & Cov(D_2|T_n) & \cdots & Cov(D_m|T_n) \end{pmatrix} \quad (7)$$

Definition 6. $\forall T_i \in TC$, $G(T_i)$ is an *n*-column boolean vector of T_i. Let $GT = (G(T_1), G(T_2), \cdots, G(T_n))$ be an $n \times n$ Boolean matrix, called a tolerance matrix.

Lemma 1. *Given a set-valued decision information system $S = (U, C \cup D, V, f)$ and the tolerance relation T_C. Let $GT = (G(T_1), G(T_2), \cdots, G(T_n))$ and $R = (r_{ij})_{n \times n}$ be an $n \times n$ matrix representing the tolerance relation. Then*

$$GT = R \tag{8}$$

which means the tolerance matrix is equal to the relation matrix in the set-valued information system.

Definition 7. $\forall D_j \in U/D$, $G(D_j)$ is an n-column boolean vector of D_j. Let $GD = (G(D_1), G(D_2), \cdots, G(D_m))$ be an $n \times m$ boolean matrix, called a decision matrix.

Lemma 2. *Let $GT = (G(T_1), G(T_2), \cdots, G(T_n))$, $GD = (G(D_1), G(D_2), \cdots, G(D_m))$ and $R = (r_{ij})_{n \times n}$. Then the support matrix can be computed as follows.*

$$Supp(D|T) = GT \bullet GD = R \bullet GD \tag{9}$$

where $\bullet$ is dot product of matrices.

Corollary 2. *Suppose $G(D_j) = (d_{1j}, d_{2j}, \cdots, d_{nj})^T$ $(j = 1, 2, \cdots, m)$. Then $\forall T_i \in TC$, $\forall D_j \in U/D$, $i = 1, 2, \cdots, n$, $j = 1, 2, \cdots, m$, the following results hold:*

(1) $Supp(D_j|T_i) = |T_i \cap D_j| = \sum_{k=1}^{n} r_{ik} d_{jk}$;
(2) $Acc(D_j|T_i) = |T_i \cap D_j|/|T_i| = \frac{\sum_{k=1}^{n} r_{ik} d_{jk}}{\sum_{k=1}^{n} r_{ik}}$;
(3) $Cov(D_j|T_i) = |T_i \cap D_j|/|D_j| = \frac{\sum_{k=1}^{n} r_{ik} d_{jk}}{\sum_{k=1}^{n} d_{kj}}$.

Lemma 2 and Corollary 2 illustrate the method of computing support, accuracy and coverage matrices by using dot product of matrices.

The support matrix, accuracy matrix and coverage matrix help us extract the useful rules from the set-valued decision information system, and the definition of interesting knowledge is shown as follows.

Definition 8. [6] $\forall T_i$ $(i = 1, 2, \cdots, n)$, $\forall D_j$ $(j = 1, 2, \cdots, m)$, if $Acc(D_j|T_i) \geq \alpha$ and $Cov(D_j|T_i) \geq \beta$ hold, we call the rule $T_i \rightarrow D_j$ a kind of interesting knowledge where $\alpha \in (0.5, 1)$ and $\beta \in (0, 1)$.

The threshold $\alpha > 0.5$ is inspired by the criterion of a simple majority rule, that is, the accuracy of the rule which is more than 0.5 can be executed.

3 Incremental Rule Acquisition Based on Matrix

We discuss about the change of rules in dynamic set-valued decision information systems when the object set evolves over time. First, we present some assumptions and the basic structure of our approaches in Section 3.1; Second, we propose the incremental approaches as well as its analysis process when single object enters into or gets out of the information system in Sections 3.2 and 3.3, respectively; Finally, we give the incremental algorithm for rule acquisition when the object set evolves over time in Section 3.4.

3.1 Assumptions and Basic Structure for the Incremental Approach

The updating of the object set is divided into two parts: the immigration of the object and the emigration of the object [6]. The former case means the object enters into the system at time $t+1$, and the universe is expanding; the latter case means the object gets out of the system, and the universe is contracting at time $t+1$.

Suppose the incrementally learning process of interesting knowledge is from time t to time $t+1$. To describe a dynamic information system, we denote a set-valued information system at time t as $S=(U,C\bigcup D,V,f)$, with a tolerance class set $TC=\{T_1,T_2,\cdots,T_n\}$ and a decision class set $U/D=\{D_1,D_2,\cdots,D_m\}$, where U is a non-empty finite set of objects at time t. At time $t+1$, some objects enter the system while some go out, so the original set-valued information system S will be changed into $S'=(U',C'\bigcup D',V',f')$. Similarly, the accuracy matrix and coverage matrix at time t are denoted as $Acc^{(t)}(D|T)$ and $Cov^{(t)}(D|T)$, respectively, and the ones at time $t+1$ as $Acc^{(t+1)}(D'|T')$ and $Cov^{(t+1)}(D'|T')$, respectively. According to Definition 8, the rule $T_i \rightarrow D_j$ is interesting if $Acc^{(t)}(D_j|X_i) \geq \alpha$ and $Cov^{(t)}(D_j|T_i) \geq \beta$ at time t; the rule $T_i' \rightarrow D_j'$ is interesting if $ACC^{(t+1)}(D_j'|T_i') \geq \alpha$ and $Cov^{(t+1)}(D_j'|T_i') \geq \beta$ at time $t+1$.

With these stipulations, we focus on two aspects: one is to discuss how to update rules when single object enters into or gets out of the information system; the other is to design its corresponding incremental algorithm for rule acquisition.

3.2 Single Object Enters into the Set-Valued Decision Information System

Assume a new object x_{n+1} enters into the information system. Then $U' = U \cup \{x_{n+1}\}$.

(a) Updating the Tolerance Matrix (Relation Matrix)

For the new object x_{n+1}, its tolerance class is $T_{n+1}=T_C(x_{n+1})=\{y \in U'|yT_Cx_{n+1}\}$.

At time t, the relation matrix is $R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{bmatrix}$. Then at time $t+1$,

$$R' = \left[\begin{array}{c|c} R & Q^T \\ \hline Q & r_{n+1,n+1} \end{array}\right] = \left[\begin{array}{cccc|c} r_{11} & r_{12} & \cdots & r_{1n} & r_{1,n+1} \\ r_{21} & r_{22} & \cdots & r_{2n} & r_{2,n+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} & r_{n,n+1} \\ \hline r_{n+1,1} & r_{n+1,2} & \cdots & r_{n+1,n} & r_{n+1,n+1} \end{array}\right], \text{ where } Q = (r_{n+1,1},$$

$r_{n+1,2},\cdots,r_{n+1,n})$.

(b) Updating the decision matrix

At time t, the decision matrix $GD = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} \end{bmatrix}$.

Case 1: Forming a new decision class.

In this case, at time $t+1$,

$$GD' = \left[\begin{array}{c|c} GD & P_2^T \\ \hline P_1 & d_{n+1,m+1} \end{array} \right] = \left[\begin{array}{cccc|c} d_{11} & d_{12} & \cdots & d_{1m} & d_{1,m+1} \\ d_{21} & d_{22} & \cdots & d_{2m} & d_{2,m+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} & d_{n,m+1} \\ \hline d_{n+1,1} & d_{n+1,2} & \cdots & d_{n+1,m} & d_{n+1,m+1} \end{array} \right],$$

where $P_1 = (d_{n+1,1}, d_{n+1,2}, \cdots, d_{n+1,m})$ and $P_2 = (d_{1,m+1}, d_{2,m+1}, \cdots, d_{n,m+1})$.
Namely, $d_{n+1,m+1} = 1$, $d_{n+1,j} = 0$ $(j = 1, 2, \cdots, m)$ and $d_{i,m+1} = 0$ $(i = 1, 2, \cdots, n)$.

Case 2: Not forming a new decision class.

At time $t+1$, $GD' = \left[\begin{array}{c} GD \\ \hline P \end{array} \right] = \left[\begin{array}{cccc} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} \\ \hline d_{n+1,1} & d_{n+1,2} & \cdots & d_{n+1,m} \end{array} \right],$

where $P = (d_{n+1,1}, d_{n+1,2}, \cdots, d_{n+1,m})$.
That is, $\exists j^* \in \{1, 2, \cdots, m\}$ such that $x_{n+1} \in D_{j^*}$. Therefore, $d_{n+1,j^*} = 1$ and $d_{n+1,j} = 0$ $(j = 1, 2, \cdots, m, j \neq j^*)$.

3.3 Single Object Gets Out of the Set-Valued Decision Information System

An object x_{i^*} $(i^* \in \{1, 2, \cdots, n\})$ gets out of the information system. $U' = U - \{x_{i^*}\}$. Thus, we just delete Line i^* and Column i^* of the relation matrix and Line i^* of the decision matrix. Then,

$$R' = \begin{bmatrix} r_{11} & r_{12} & \cdots & \cancel{r_{1i^*}} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & \cancel{r_{2i^*}} & \cdots & r_{2n} \\ \vdots & \vdots & \cdots & \vdots & \ddots & \vdots \\ \cancel{r_{i^*1}} & \cancel{r_{i^*2}} & \cdots & \cancel{r_{i^*i^*}} & \cdots & \cancel{r_{i^*n}} \\ \vdots & \vdots & \cdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & \cancel{r_{ni^*}} & \cdots & r_{nn} \end{bmatrix} \text{ and } GD' = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \cancel{d_{i^*1}} & \cancel{d_{i^*2}} & \cdots & \cancel{d_{i^*m}} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} \end{bmatrix},$$

where the elements with "-" in the matrix are deleted.

Remark: After updating the relation matrix and decision matrix, according to Lemma 2 and Corollary 2, we update the support, accuracy and coverage matrices and acquire rule incrementally by Definition 8.

3.4 An Incremental Algorithm for Rule Acquisition in the Set-Valued Information System

In this subsection, we design an increment algorithm (Algorithm 1) corresponding to Section 3.2 and Section 3.3.

Algorithm 1. An incremental algorithm for rule acquisition in the set-valued information system

---

Input: (1) A set-valued decision information system $S = \{U, C \cup \{d\}, V, f\}$;
        (2) Two thresholds α and β.
Output: Support matrix, Accuracy matrix, Coverage matrix, Rules at time t and $t+1$, respectively.

1 **begin**
    `// At time t, construct the relation matrix and decision matrix,`
    `compute the support matrix, accuracy matrix and coverage matrix,`
    `and output the rules.`
2   **Construct** the relation matrix $R = (r_{ik})_{n \times n}$ according to Definition 3 and Corollary 1.
3   **Construct** the decision matrix $GD = (d_{kj})_{n \times m}$ according to Definition 7.
4   **Compute** the support matrix $Supp^{(t)}(D|T) = R \bullet GD$ according to Lemma 2.
5   **Compute** the accuracy and coverage matrices $Acc^{(t)}(D|T)$, $Cov^{(t)}(D|T)$ according to Corollary 2.
    In detail, $Acc^{(t)}(D_j|T_i) = \frac{\sum_{k=1}^{n} r_{ik}d_{jk}}{\sum_{k=1}^{n} r_{ik}}$, $Cov^{(t)}(D_j|T_i) = \frac{\sum_{k=1}^{n} r_{ik}d_{jk}}{\sum_{k=1}^{n} d_{kj}}$ $(i = 1, 2, \cdots, n, j = 1, 2, \cdots, m)$.
6   **for** $i = 1$ **to** n **do**   `// Rule acquisition`
7   **for** $j = 1$ **to** m **do**
8   **if** $Acc^{(t)}(D_j|T_i) \geq \alpha$ *and* $Cov^{(t)}(D_j|T_i) \geq \beta$ **then**
9   **Output** the rule $T_i \to D_j$.
10   **end**
11   **end**
12   **end**
    `// At time t+1, update the relation matrix and decision matrix,`
    `compute the support matrix, accuracy matrix and coverage matrix,`
    `and output the rules.`
13   **if** *A new object x_{n+1} enters into the information system* **then**
14   **Let** $n' = n + 1$.
15   **Update** the relation matrix according to (a) in Section 3.2, $R' = \left[\begin{array}{c|c} R & Q^T \\ \hline Q & r_{n+1,n+1} \end{array} \right]$.
        `// Update the decision matrix according to (b) in Section 3.2`
16   **if** *forming a new decision class* **then**
17             $GD' = \left[\begin{array}{c|c} GD & P_2^T \\ \hline P_1 & d_{n+1,m+1} \end{array} \right]$.
18   **Let** $m' = m + 1$.
19   **else**
20             $GD' = \left[\begin{array}{c} GD \\ \hline P \end{array} \right]$. **Let** $m' = m$.
21   **end**
22   **else if** *An object x_{i^*} gets out of the information system* **then**
23   **Let** $n' = n - 1$.
24   **Update** the relation matrix R' and decision matrix GD' according to Section 3.3.
25   **end**
26   **Compute** the support matrix $Supp^{(t+1)}(D'|T') = R' \bullet GD'$ according to Lemma 2.
27   **Compute** the accuracy and coverage matrices $Acc^{(t+1)}(D'|T')$, $Cov^{(t+1)}(D'|T')$ according to Corollary 2.
28   **for** $i = 1$ **to** n' **do**   `// Rule acquisition`
29   **for** $j = 1$ **to** m' **do**
30   **if** $Acc^{(t+1)}(D_j'|T_i') \geq \alpha$ *and* $Cov^{(t+1)}(D_j'|T_i') \geq \beta$ **then**
31   **Output** the rule $T_i' \to D_j'$.
32   **end**
33   **end**
34   **end**
35 **end**

4 An Illustration

Table 1 presents a set-valued information system $(U, C \cup D, V, f)$, where $C = \{c_1, c_2, c_3, c_4\}$, $D = \{d\}$ and $U = \{x_1, x_2, x_3, x_4, x_5\}$. Therefore, $T_1 = T_C(x_1) = \{x_1, x_2, x_4\}$, $T_2 = T_C(x_2) = \{x_1, x_2, x_3, x_4, x_5\}$, $T_3 = T_C(x_3) = \{x_2, x_3, x_5\}$, $T_4 = T_C(x_4) = \{x_1, x_2, x_4, x_5\}$, $T_5 = T_C(x_5) = \{x_2, x_3, x_4, x_5\}$. $U/D = \{D_Y, D_N\} = \{\{x_1, x_2, x_4, x_5\}, \{x_3, x_4\}\}$.

Then, the relation matrix $R = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}$ and the decision matrix $GD = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$.

Thus, $Supp^{(t)}(D|T) = R \bullet GD = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 3 & 2 \\ 2 & 1 \\ 3 & 1 \\ 2 & 2 \end{bmatrix}$.

Consequently, the accuracy matrix and the coverage matrix at time t are computed, respectively, as follows.

$$Acc^{(t)}(D|T) = \begin{bmatrix} 2/3 & 1/3 \\ 3/5 & 2/5 \\ 2/3 & 1/3 \\ 3/4 & 1/4 \\ 2/4 & 2/4 \end{bmatrix} = \begin{bmatrix} 0.67 & 0.33 \\ 0.6 & 0.4 \\ 0.67 & 0.33 \\ 0.75 & 0.25 \\ 0.5 & 0.5 \end{bmatrix} \tag{10}$$

$$Cov^{(t)}(D|T) = \begin{bmatrix} 2/3 & 1/2 \\ 3/3 & 2/2 \\ 2/3 & 1/2 \\ 3/3 & 1/2 \\ 2/3 & 2/2 \end{bmatrix} = \begin{bmatrix} 0.67 & 0.5 \\ 1 & 1 \\ 0.67 & 0.5 \\ 1 & 0.5 \\ 0.67 & 1 \end{bmatrix} \tag{11}$$

We then consider the case of the object set varying at time $t + 1$. Suppose at time $t + 1$, (1) a new object x_6 (where $c_1 = \{1\}, c_2 = \{1\}, c_3 = \{0, 1\}, c_4 = \{0, 1\}, d = N$) enters into the system; (2) x_5 gets out of the system.

Table 1 A set-valued information system

U	c_1	c_2	c_3	c_4	d
x_1	$\{0\}$	$\{0\}$	$\{1,2\}$	$\{1,2\}$	Y
x_2	$\{0,1,2\}$	$\{0,1,2\}$	$\{1,2\}$	$\{0,1,2\}$	Y
x_3	$\{1,2\}$	$\{1\}$	$\{1,2\}$	$\{1,2\}$	N
x_4	$\{0,1\}$	$\{0,2\}$	$\{1,2\}$	$\{1\}$	N
x_5	$\{1,2\}$	$\{1,2\}$	$\{1,2\}$	$\{1\}$	Y

(1) A new object x_6 enters into the system

According to Section 3.2, we update the relation matrix and decision matrix as follows. Since $T_6 = T_C(x_6) = \{x_2, x_3, x_5, x_6\}$ and $f(x_6, d) = N$, then $Q = (0, 1, 1, 0, 1)$, $r_{66} = 1$ and $P = (0, 1)$ (Case 2 in Section 3.2: not forming a new decision class).

Thus,

$$R' = \left[\begin{array}{c|c} R & Q^T \\ \hline Q & r_{66} \end{array}\right] = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix} \text{ and } GD' = \left[\begin{array}{c} GD \\ \hline P \end{array}\right] = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

(2) An object x_5 gets out of the system (Continuation of (1))

By Section 3.3, we just delete Line 5 and Column 5 of the relation matrix and Line 5 of the decision matrix. Then,

$$R' = \begin{bmatrix} 1 & 1 & 0 & 1 & \cancel{0} & 0 \\ 1 & 1 & 1 & 1 & \cancel{1} & 1 \\ 0 & 1 & 1 & 0 & \cancel{1} & 1 \\ 1 & 1 & 0 & 1 & \cancel{1} & 0 \\ \cancel{0} & \cancel{1} & \cancel{1} & \cancel{1} & \cancel{1} & \cancel{1} \\ 0 & 1 & 1 & 0 & \cancel{1} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix} \text{ and } GD' = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ \cancel{1} & \cancel{0} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

Therefore, $Supp^{(t+1)}(D'|T') = R' \bullet GD' = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 2 & 3 \\ 1 & 2 \\ 2 & 1 \\ 1 & 2 \end{bmatrix}.$

Hence, the accuracy matrix and the coverage matrix at time $t + 1$ are computed, respectively, as follows.

$$Acc^{(t+1)}(D'|T') = \begin{bmatrix} 2/3 & 1/3 \\ 2/5 & 3/5 \\ 1/3 & 2/3 \\ 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix} = \begin{bmatrix} 0.67 & 0.33 \\ 0.4 & 0.6 \\ 0.33 & 0.67 \\ 0.67 & 0.33 \\ 0.33 & 0.67 \end{bmatrix} \qquad (12)$$

$$Cov^{(t+1)}(D'|T') = \begin{bmatrix} 2/2 & 1/3 \\ 2/2 & 3/3 \\ 1/2 & 2/3 \\ 2/2 & 1/3 \\ 1/2 & 2/3 \end{bmatrix} = \begin{bmatrix} 1 & 0.33 \\ 1 & 1 \\ 0.5 & 0.67 \\ 1 & 0.33 \\ 0.5 & 0.67 \end{bmatrix} \qquad (13)$$

We use (10) and (11) to construct the 2-dimensional value pairs $(Acc^{(t)}(D_j|T_i), Cov^{(t)}(D_j|T_i))$ at time t. Similarly, we use (12) and (13) to construct the 2-dimensional value pairs $(Acc^{(t+1)}(D'_j|T'_i), Cov^{(t+1)}(D'_j|T'_i))$ at time $t + 1$. Obviously, different thresholds of accuracy and coverage lead to different rules derived. For example, we

set two thresholds $\alpha = 0.6$ and $\beta = 0.5$. Then, the rules $T_1 \rightarrow D_Y$, $T_2 \rightarrow D_Y$, $T_3 \rightarrow D_Y$ and $T_4 \rightarrow D_Y$ are interesting knowledge at time t according to (10) and (11). At time $t+1$, according to (12) and (13), the rules $T_1 \rightarrow D_Y$, $T_2 \rightarrow D_N$, $T_3 \rightarrow D_N$, $T_4 \rightarrow D_Y$, $T_6 \rightarrow D_N$ are interesting knowledge.

5 Conclusions

We proposed a concept of interesting knowledge under both accuracy and coverage in the set-valued decision information systems. The tolerance matrix and decision matrix compute the accuracy and the coverage of the generated rules. Aiming at the variation of the object set, we proposed the incremental approach and its corresponding algorithm to acquire rules incrementally. We validated the proposed approach and algorithm by a case study. A future work is to test the efficiency of the proposed algorithm, and to develop new approaches for dynamically learning interesting knowledge in hybrid (numerical and categorical) information systems.

References

1. Guan, Y., Wang, H.: Set-valued information systems. Information Sciences 176(17), 2507–2525 (2006)
2. Hu, Q., Yu, D., Liu, J., Wu, C.: Neighborhood rough set based heterogeneous feature subset selection. Information Sciences 178(18), 3577–3594 (2008)
3. Kolman, B., Busby, R.C., Ross, S.C.: Discrete Mathematical Structures, 5th edn. Prentice-Hall, Inc., Upper Saddle River (2003)
4. Li, T., Ruan, D., Wets, G., Song, J., Xu, Y.: A rough sets based characteristic relation approach for dynamic attribute generalization in data mining. Knowledge-Based Systems 20(5), 485–494 (2007)
5. Liu, D., Li, T., Ruan, D., Zhang, J.: Incremental learning optimization on knowledge discovery in dynamic business intelligent systems. Journal of Global Optimization 8, 1–20 (2010), doi:10.1007/s10898-010-9607-8
6. Liu, D., Li, T., Ruan, D., Zou, W.: An incremental approach for inducing knowledge from dynamic information systems. Fundamenta Informaticae 94(2), 245–260 (2009)
7. Liu, G.: The axiomatization of the rough set upper approximation operations. Fundamenta Informaticae 69(3), 331–342 (2006)
8. Liu, G.: Axiomatic systems for rough sets and fuzzy rough sets. International Journal of Approximate Reasoning 48(3), 857–867 (2008)
9. Michalski, R.: Knowledge repair mechanisms: evolution vs. revolution. In: Proc. ICML 1985, pp. 116–119 (1985)
10. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data, System Theory, Knowledge Engineering and Problem Solving, vol. 9. Kluwer Academic Publishers, Dordrecht (1991)
11. Pawlak, Z., Skowron, A.: Rough sets and boolean reasoning. Information Sciences 177(1), 41–73 (2007)
12. Pawlak, Z., Skowron, A.: Rough sets: Some extensions. Information Sciences 177(1), 28–40 (2007)

13. Pawlak, Z., Skowron, A.: Rudiments of rough sets. Information Sciences 177(1), 3–27 (2007)
14. Qian, J., Miao, D., Zhang, Z., Li, W.: Hybrid approaches to attribute reduction based on indiscernibility and discernibility relation. International Journal of Approximate Reasoning 52(2), 212–230 (2011)
15. Qian, Y., Dang, C., Liang, J., Tang, D.: Set-valued ordered information systems. Information Sciences 179(16), 2809–2832 (2009)
16. Shan, N., Ziarko, W.: Data-based acquisition and incremental modification of classification rules. Computational Intelligence 11(2), 357–370 (1995)
17. Zhang, J., Li, T., Ruan, D., Liu, D.: Neighborhood rough sets for dynamic data mining. In: World Conference on Soft Computing, San Francisco, May 23-26 (2011)
18. Zhang, W., Ma, J., Fan, S.: Variable threshold concept lattices. Information Sciences 177(22), 4883–4892 (2007)
19. Zhang, W., Mi, J.: Incomplete information system andits optimal selections. Computers & Mathematics with Applications 48(5-6), 691–698 (2004)
20. Zheng, Z., Wang, G.: RRIA: A rough set and rule tree based incremental knowledge acquisition algorithm. Fundamenta Informaticae 59(2-3), 299–313 (2004)

A Virtual Supply Chain Model for QoS Assessment

Roman Gumzej and Brigita Gajšek

Abstract. A hybrid virtual supply chain model for the assessment of Quality of Service (QoS) in supply chains is being presented. Due to the large complexity of the problem domain a combined – analytical and simulation – model has been recognised as the best solution for QoS evaluation within a supply chain. The main problem of building such a model is the level of detail, which on one hand is limited by the transparency of business processes of contributing partners and on the other by the desired relevancy of the gathered results. Our model builds on data that are usually made public or are at least subject to contract terms among partners in a supply chain and provide a "good-enough" foundation for QoS evaluation.

1 Introduction

When devising models of real-life processes, we generally speak of two kinds or types of models – static and dynamic. Static models are deterministic models whose elements and their interrelations can be expressed algebraically – by mathematical formulas. The laws given by them apply at all times and the results are non-volatile. Dynamic models are meant for problems, where the time dimension is also important. A typical characteristic of dynamic models is also the stochastic nature of their element's properties and volatility of their interrelations, both affecting the results, which are valid at a given time or for a given time frame.

Another characteristic of models pertains to the results we wish to obtain. Quantitative models - mathematical models – are based on cardinally measurable values. They are based on a set of variables that vary over a specific domain, while quantitative and causal relationships have been defined between these variables (Bertrand and Fransoo, 2002).

Roman Gumzej · Brigita Gajšek
University of Maribor – Faculty of Logistics
Mariborska cesta 7, SI-3000 Celje, Slovenia
e-mail: roman.gumzej@fl.uni-mb.si

The causal relationships of qualitative models are not being quantified; instead, they are described based on ordinal or nominal scalable data. Qualitative models can extend quantitative ones to enhance the comprehensibility of a problem.

Quantitative models are either descriptive or prescriptive. Prescriptive – optimization – models comprise in addition to causal relationships also target relations that determine an optimal configuration (Evans and Olson, 2001). They are usually modelled by equations' systems and solved analytically; therefore, they are considered analytical models. Hence, the optimal solution can be obtained by convergence directions and/or heuristic algorithms (Simchi-Levi et. al., 2000).

The alternatives to prescriptive models are descriptive models or analysis models. They don't comprise target relationships; hence, they cannot render an optimal system configuration (Evans and Olson, 2011). Instead, they render performance indicators based on known causal relationships. They can also be used for comparison among different process-designs, to study process-behaviour at different points (in time) and to predict future process-behaviour.

Descriptive models can be analytical or simulation models. Typical examples of descriptive, analytical models in supply chain management (SCM) are queuing models. Methodical alternatives to analytical models are simulation models. Here, the causal relationships are represented formally/logically by dynamic simulation models with emphasis on the stochastic nature of processes and their temporal behaviour. Computer simulation refers to methods for studying a wide variety of models of real-world systems by numerical evaluation where software is used to imitate the system's operational characteristics, often over time. Simulation experiments enable the analysis of complex system behaviour of different system configurations and states.

Our system model is based on analytical methods and a simulation model. It is quantitative, dynamic, stochastic and descriptive in its nature. We speak of a hybrid model combining approaches for reduced complexity, enhanced comprehensibility, introducing stochastic processes by means of statistical distributions of process variables, and reduced effort to build/maintain the model by joining analytical and simulation approaches. For problems of large complexity like ours hybrid models are a good alternative to pure analytical or simulation models. One of the first logistics applications thereof dates back to 1972 (Nolan and Sovereign, 1972).

The evaluation of our model renders performance indicators that pertain to the QoS among nodes of a supply chain, defined in (Gumzej and Gajšek, 2011). The fine-tuning of our model occurs in a closed loop after the evaluation step based on thresholds from some reference values given for our QoS indicators. Due to the complex nature of the problem domain, this step requires human (analysts) intervention and is thus done "off-line".

2   Model Composition for Complex Supply Chain Analysis

Our hybrid supply chain model is characterized by:

- A complex system architecture with supply chain processes, characterized by possibly large number and variability of elements

- A balance is sought among detailed analysis of dynamic stochastic processes and resources/time required
- No constraints are imposed to the number of elements in the system to better represent dynamical interrelationships in the entire system
- On the aggregation-level only basic classification of elements according to their roles in the supply chain is foreseen; however, no constraints are applied concerning the choice of design alternatives

The necessary steps to build our model are:

1. Design and analyse the goods/services flow as a queuing system; for complexity reduction only the most critical process-steps are modelled
2. Processes are represented by discrete events in our simulation model
3. Supply chain nodes are represented by decomposable simulation nodes with attributes of combined – simulation and analytical models

The criteria for the choice of the mentioned model-construction granularity are mainly its complexity and the urgency for optimization of its processes – mass customization (Davis, 1996).

Our simulation model is an open mass service model of type G/G/m. Its service-nodes (SN, see Figure 1) are modelled with respect to general distributions of arrivals and processing times with m parallel processes at each node. The input-flow is non-ordinal and non-homogeneous. We have different input (IBS) and output (OBS) batch sizes. There are in general an infinite number of places in each node's queue, which of course is unrealistic, but due to planned demand this is not a serious limitation and we can limit them at any time without changing the design. When assigning service-units (SU) to input requests the Just-In-Time (JIT) scheduling policy (Gumzej and Lipičnik, 2009) with resource limitations applies. The model is resolved by the decomposition of service-nodes to service-units where their distributions of preparation, processing and out-of-order times apply. The overall simulation time is given in days and is predetermined by the analyst.
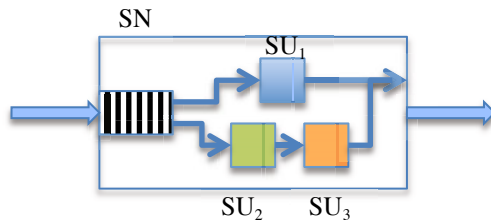
Fig. 1 Service-node structure

The output data from the simulation for each service node are: timeliness, correctness, availability, flexibility, robustness and processing cost.

2.1 *Analytical Model*

To evaluate a supply chain design a large number of relevant criteria (assets, service-levels, pass-through times, loads, costs, etc.) need to be considered by multi-criteria evaluation. Since the various parameters are in general independent, the optimization of individual values does not lead to a global optimum. Hence, multi-criteria evaluation is usually applied to render an adequate solution. There are a number of approaches to do this – pre-emptive optimisation, service-level concept, target weighing and goal programming. In our combined approach target weighing and goal programming are used.

Considering all listed goals, the virtual supply chain model for supply chain QoS evaluation needs to fulfil the following conditions:

- Process-orientation and consideration of all listed QoS parameters with appropriate handling of competing goals
- Consideration of the supply chain strategy and risk factors of processes
- Practical applicability and acceptance, being supported by the model's simplicity and integration of the well-established SCOR model

To describe the structure of our model it is reasonable to use the terminology introduced by the Supply Chain Council (hereinafter referred to as SCC) in the form of Supply Chain Operations Reference Model (hereafter referred to as the SCOR Model). This model has been used by practitioners for many years, the SCOR-model provides a unique framework that links business processes, metrics, best practices and technology features into a unified structure to support communication among supply chain partners and to improve the effectiveness of supply chain management and related supply chain improvement activities (SCC, 2008). SCOR is based on five distinct processes: *Plan, Source, Make, Deliver* and *Return* (see Figure 2).

According to the SCOR model, various activities between any two organizations are done by SOURCE (at the first organization) and DELIVER (at the second organization). SOURCE stands for processes that procure goods and services to meet planned or actual demand. DELIVER encompasses processes that provide finished goods and services to meet planned or actual demand, typically including order management, transportation management, and distribution management. In relation with the SOURCE process we have included: receive and verify products; authorize supplier payments; supplier agreements. In relation with the DELIVER process we have included: processing customer inquiries and orders; invoicing customer.

MAKE stands for processes that transform a product to a finished state to meet planned or actual demand. These processes require resources, time to prepare and time to process. These processes are prone to outages, resulting in unavailability or reduced processing capacity. In a case where we have parallel processors parallel processes are possible. In relation with the MAKE process we have included: time to prepare, time to process, associated costs and out-times distributions.

Reverse flow is perceived by the RETURN processes in connected organizations. In relation with the RETURN process we have included: returns of defective products; returns of products for maintenance, repair, overhaul; returns of excess products. The RETURN combines processes associated with returning or receiving returned products for any reason. These processes extend into post-delivery customer support (SCC, 2008).

According to SCC (2008), from the list of sub-processes and associated tasks only those were selected that involve the exchange of information, products, services, documents and finances between entities of different organizations.

Source: Supply Chain Council, 2010

Fig. 2 The SCOR Structure

The main purpose of business between two organizations is sourcing. Links between them are therefore an ideal spot to place probes for cyclically checking whether the objective is reached again and again. Table 1 presents the main operational transactions between two organizations.

Table 1 Operational transactions between observed organization and its supplier

		Supplier		
Observed organization	1.	⚙ request for quotation (starts exchange)		
	2.	⚙ confirmation of receiving the request for quotation (AVAILABILITY)	OR	⚙ no confirmation of receiveing
	3.	⚙ quotation (confirmation of the possibility of realization) (promised RELIABILITY)	OR	⚙ rejection of the possibility of realization
	4.	—•←— agreement (source of reference values: payment, delivery, reclamation period)		
	5.	⚙ order (starts exchange)		
	6.	⚙ confirmation of receiving the order (AVAILABILITY)		
	7.	⚙ confirmation / rejection of the possibility of realization (promised RELIABILITY)		
	8.	⚙ confirmation of shippment (AVAILABILITY)		
	9.	⚙ delivery (TIMELINESS, CORRECTNESS) (proof of promised RELIABILITY)		
	10.	⚙ confirmation of delivery	OR	⚙ complaint to quality, quantity, …
	11.	⚙ invoice		⚙ confirmation of receiving the reclamation (AVAILABILITY) ⚙ confirmation on resolving the situation (promised RELIABILITY)
	12.	⚙ payment (LIABILITY) (proof of promised RELIABILITY)		⚙ resolve complaint (LIABILITY) (proof of promised RELIABILITY) ⚙ confirmation of delivery ⚙ invoice ⚙ payment (LIABILITY) (proof of promised RELIABILITY)

The communication between first organization Source Process and second organization Deliver Process is excellent if the individual line items making up an order are all perfectly matched and on time. In this case there is no need for alarms or searching for causes of irregularities.

Dialogue and exchange between the two organizations must be made in the shortest time possible, because these are necessary activities that do not add direct value to the consumer. Observed organization, anyone except the third tier supplier, repeatedly sends requests for communication to suppliers. These are request for quotation, order, confirmation of delivery, complaint to quality, quantity, etc. Expected results are exchange of information, materials and so on.

To compare alternative supply chain designs multi-criteria decision-making is used by adoption of the weighted goal programming approach. The target function for supply chain QoS is formulated in the form of Equation 1:

$$Q_a = \min\left(\sum_g p_g \sigma G_a\right) \xrightarrow{\min_a} Q \tag{1}$$

minimised across all alternatives a $\in$ *{1, 2, …, A}* at each node, where $p_g \in (0, 1)$ is the ponder for the parameter *G* and *sigma G* is its deviation, calculated from Equation 2:

$$\sigma G_a = \sqrt{\frac{1}{N} \sum_i \sum_j \max(g_{a,i,j} - \hat{g}_{a,i,j}, 0)^2} \tag{2}$$

where $g_{a,i,j}$ denotes the measured value of G at probe $x_{i,j}$, $\hat{g}_{a,i,j}$ the reference value for $g_{a,i,j}$ at that probe and N the respective number of probes.

The measured QoS of a supply chain is eventually determined by the minimisation of the Q values as shown in Equation 1. The minimisation encompasses all possible alternatives at each node and may vary in time.

The relevant parameters for our QoS evaluation have been identified in (Gumzej and Gajšek, 2011) and comprise correctness, dependability and cost indicators.

Organization's ability to react on a sent request within or outside a given time frame can be in our case described by AVAILABILITY. Good AVAILABILITY can be evidenced by different kinds of confirmation of receiving sender's document (request for quotation, order, shipment…).

Organization is RELIABLE when it is able to perform its required functions under stated conditions for a specified period of time. It appears that it is able to realize any offer, order, complaint or reclamation. On the connection between any two organizations submissions of tenders, order confirmations, revised/new items as an answer to complaint have to be checked.

Everything exchanged must be delivered on predetermined time, complete and undamaged. We have to check TIMELINESS (on time) and CORRECTNESS (all orders (lines) are complete and are undamaged). Status of CORRECTNESS is

permitted to be FALSE in the predetermined time window. When the time window for delivery is closed also TIMELINESS becomes FALSE. In the latter case TIMELINESS and CORRECTNESS are at the lowest value. When each first delivery is on time and all order lines show complete and undamaged status, no complaints are present, TIMELINESS and CORRECTNESS are at the highest value.

SAFETY and SECURITY are inside the delivering organization in questionable state/condition, if the first time exchange is not on time and complete and undamaged, or if communication exceeds the available time. They cannot be measured on any link but if the link is weak it can be such because of a deteriorating state of safety and security in a delivering organization. If all connections are rated as good and acceptable, well-established safety in security can be deduced within organizations in a supply chain – supply chain is SAFE and SECURE until the first occurrence of a delay, incomplete delivery or damage. SAFETY and SECURITY of any supply chain as whole have only two states: SAFE/SECURE or UNSAFE/UNSECURE. At occurrence of the second mentioned status organisations at the beginning of identified weak links have to consider the improvement of their SAFETY in SECURITY.

Any organization is LIABLE as an individual if it has acquired certificates for quality, business excellence… Existence of certificates is usually checked before the first business contract is signed. After an established partnership, higher LIABILITY has links with on-time payments, and those who responsibly react to the damage caused to customers or suppliers, reacting to complaints and reclamations and resolve them within a reasonable time. Each resolved complaint and reclamation is a proof of LIABILITY. Subject of a complaint or reclamation can be not conducted payment for rendered services or delivered goods, damaged or inappropriate goods, failure to comply with delivery dates.

For the sake of simplification of our model and to avoid interrelationships that exist among individual indicators, we are defining only a few that shall form the basis for the evaluation of Q, namely:

C and T – representing the proportions of correctly and on time completed processes, namely:

$$C = \frac{c}{d} \qquad \text{and} \qquad T = \frac{t}{d} \tag{3}$$

where c represents the number of fulfilled demands, t the number of timely fulfilled demands and d the total number of demands. Timely fulfilled demands are demands whose processing time $(t_{out} - t_{in})$ is less than the time specified for delivery.

A – representing availability, being the proportion of time a node is in a functioning condition – available to its demands, namely:

$$A = 1 - \frac{MTTR}{MTTF} \qquad (4)$$

where MTTF represents its average Mean-Time-to-Failure and MTTR its average Mean-Time-to-Repair.

F – representing flexibility, representing the proportion of possible input source alternatives at any node – input source redundancy, namely:

$$F = 1 - \frac{1}{n} \qquad (5)$$

where n represents the average number of possible input sources.

R – robustness representing the average redundancy of processors at any node – proportion of processing redundancy, namely:

$$R = 1 - \frac{1}{m} \qquad (6)$$

where m represents the average number of parallel processing units.

P – price is calculated based on the cost of processing at each node, where the sum of processing costs of individual units processing transactions is collected, namely:

$$P = CM + PQ \cdot cp + (TS + TP) \cdot ct \qquad (7)$$

where *CM* represents the cost of material, *PQ* the processing quantity, *cp* the cost per processing unit (product/service cost), *TS* the time to process (service time) and *TP* the time to prepare with the appropriate *ct* cost of product/service processing time.

2.2   Simulation Model

Our simulation model has to enable the monitoring of supply chain behaviour in time and hence builds on network dynamics theory. The general input data to parameterise our model are the time period (frame) in which we observe the monitored network. For any supply chain node and/or its unit we need to know the distributions of preparation times, service times and MTTF/MTTR times.

For each PRODUCT/SERVICE the following data must be available: description, required quantity (for the observed time frame), distribution of demand, production site and its composition (e.g. its Bill of Material (BOM), providing the mapping of components vs. product in the form of N:1 relation).

For each SUPPLY CHAIN NODE the following data are available: description, number of parallel processes, availability (% of the observed time frame) based on MTTF and MTTR of its processing units.

Our INPUT and OUTPUT MATERIAL FLOWS are characterised by the batch sizes: production batch size (number of products of the same type/period) and transport batch size (number of products being transported together to the next node).

SERVICE/PRODUCTION is described by the work-schedule (supply chain logic), where the individual activities carry the following attributes: deadline, time to prepare, service time, production node/unit assignment.

Our simplified model structure is shown in Figure 3. Based on customer orders, transactions are being generated containing information on the time of order, product/service, quantity and time to deliver. Customer orders are formed according to the "demand to forecast strategy" based on previously assembled order distributions. Our model is evaluated considering these data and the outcome represents customer satisfaction – quality of our supply chain.
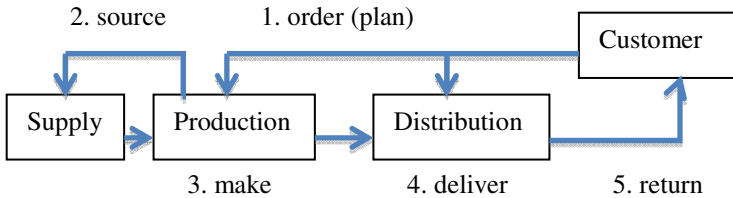
Fig. 3 Our SCOR model structure

3 Classifications of Supply Chain Nodes and Transactions

Changed supply chains models are putting the relevant quality criteria into a new perspective. The quality of products and services are not regarded solely from the producer-consumer perspectives. Since customer-supplier relations exist throughout the entire supply chain and can be established between any pair of elements, quality depends not only on the quality policy of the producers but on quality of each link in the supply chain. Supply chain element stands for an organisation, which shall be divided into several entities (e.g. inbound warehouse, production facility, output storage…). To take into account various possible geographic and logical distributions of organization units one must be careful in determining the sources of information, material, service and so on, Figure 4. When analysing weak links one shall wish to consider more closely the units of the pertaining organisation.
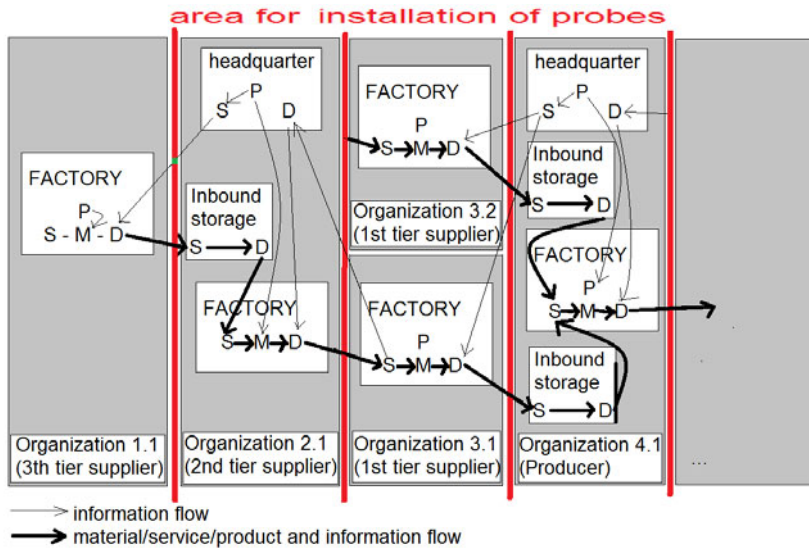
Fig. 4 Graphical presentation of a part of a supply chain

For the purpose of building our experimental model we need to define, in addition to the stated QoS Criteria, a list of possible classes of organizations in the structure of a supply chain and a list of possible interactions between all conceivable links between the various aforementioned nodes.

Basic organizations of supply chains are third tier suppliers, second tier suppliers, first tier suppliers, producers, distributors, retailers and consumers. A first tier supplier is a supplier that invoices the customer for goods and services rendered directly by that supplier. A second tier supplier is a supplier that invoices the first-tier supplier for goods and services rendered. A third tier supplier is a supplier that is more than one step away from the first tier supplier, providing goods and services to second tier suppliers. First tier supplier's vendors are second tier, and all other vendors are third tier. Every organization has impact on overall QoS and is as such taken into the consideration.

On the connections between the above listed organizations in a supply chain we shall observe the physical, documentation and cash flows. From the viewpoint of supply chain physical flow we can include raw materials, goods, semi-finished products and finished products flow. Information flow can be established with data attached on objects in physical flow or with data included in content of various documents. Cash flow is established for the purpose of payment of purchased materials, goods and products and for the reimbursement of costs for non-quality aforementioned items. Waste materials, inadequate semi-products, spent products are traveling backwards the supply chain for recycling, reclamation or destruction.

Transactions between pairs of organizations are conducted on different levels. Three most frequently mentioned are: operational, tactical and strategic. In our research we will focus on transactions on the operational level. From the examination of several case studies we have learned that the initial transactions among

newly connected organizations are performed at this level. Although later on, when two organizations develop trust, they also develop mutual activities at the tactical and strategic levels, links carrying transactions between partner organisations in a supply chain are represented on the operational level. Transaction processing activities are usually started sequentially, but their processing considering the supply chain logic is parallel and distributed. They are predominantly started by the organization that is (in the structure of the supply chain) positioned closer to the consumer or by the consumer itself.

4 Conclusion

In this article a supply chain (network) model is laid out being the foundation of our virtual supply chain model. In its two-fold structure it encompasses the evaluation of QoS indicators from the static (analytical) part of our model, and dynamic (simulation) part of our model. Its exploration should render useful QoS indicators for individual supply chain nodes as well a benchmark of excellence for the entire supply chain.

Literature

[1] Bertrand, J.W.M., Fransoo, J.C.: Modelling and simulation: Operations management research methodologies using quantitative modelling. International Journal of Operations and Production Management 22(2), 241–264 (2002)
[2] Davis, S.: Future Perfect. Addison-Wesle, Reading (1996)
[3] Evans, J.R., Olson, D.L.: Introduction to Simulation and Risk Analysis, 2. Aufl. Prentice Hall, Upper Saddle River (2001)
[4] Gumzej, R., Gajšek, B.: ntroducing Quality of Service Criteria into Supply Chain Management for Excellence. International Journal of Applied Logistics (IJAL) 2(1), 1–16 (2011)
[5] Gumzej, R., Lipičnik, M.: Anwendung von Methoden der Logistik und Netzplantechnik zur präzedenz- und ressourcenbeschränkten Ablaufplanung von Echtzeitsystemen. In: Fachtagung des GI/GMA-Fachausschusses Echtzeitsysteme (real-time), Boppard, 19. und 20. Software-intensive verteilte Echtzeitsysteme (In-formatik Aktuell), pp. 49–56. Springer, Heidelberg (2009)
[6] Nolan, R.L., Sovereign, M.G.: A Recursive Optimization and Simulation Approach to Analysis with an Application to Transportation. Management Science 18(12), 676–690 (1972)
[7] SCC, Supply-Chain Operations Reference-model, Internet page (2008),
`http://supply-chain.org/f/SCOR90OverviewBooklet.pdf`
(obtained May 27, 2011)
[8] Simchi-Levi, D., Kaminsky, P., Simchi-Levi, E.: Designing and Managing the Supply Chain: Conceptys, Strategies, and Case Studies. Irwin McGraw-Hill, New York (2000)

Modelling and Simulation of Coupled Systems-Water and Energy - Case Study of the Water Reservoir System of the Rimac River Catchments

Gloria Robleto, Manfred Schütze, and Edy Godoy

Abstract. Approx 80 % of the fresh water consumption of the megacity of Lima (Peru) is covered by the reservoir system of the Rimac river catchments. A further task is the production of electrical energy by hydropower plants. Lima, with its 8 million inhabitants, is situated on the very dry pacific coast of Peru. Continuous water supply is a challenging task. Water supply and energy production lead to operational conflicts regarding the water releases of the reservoir system. Currently, the decision about the water releases is based on a general discharge Plan, precipitation forecast and a simplified reservoir model. This contribution presents the first working steps for the optimisation of the operation of the reservoir system based on a more detailed modelling approach. The model of the reservoir system has been simulated for two different hydrological periods (dry and humid) including the water discharge plan of the operating company.

Keywords: Energy, Modelling, Operation, Reservoir systems, Water supply.

1 Introduction

The pacific coast of Peru is one of the driest areas in the world. The fast growing capital Lima with its currently 8 million inhabitants needs a large amount of

Gloria Robleto · Manfred Schütze
Ifak, Institut für Automation und Kommunikation e.V. Magdeburg,
Werner-Heisenberg Str.1, 39106, Magdeburg, Germany
e-mail: gloria.robleto@ifak.eu, manfred.schuetze@ifak.eu

Edy Godoy
EDEGEL S.A.A, Av. Victor Andrés Belaúnde N° 147 Torre Real 4,
Piso 7-San Isidro, Lima, Perú
e-mail: egodoy@edegel.com

drinking water. The most important fresh water source for water supply is the reservoir system of the Rimac river catchments located in the Andean mountains. The reservoir system consists of 21 reservoirs, canals, tunnels and natural rivers for the water transport (see Fig. 1).
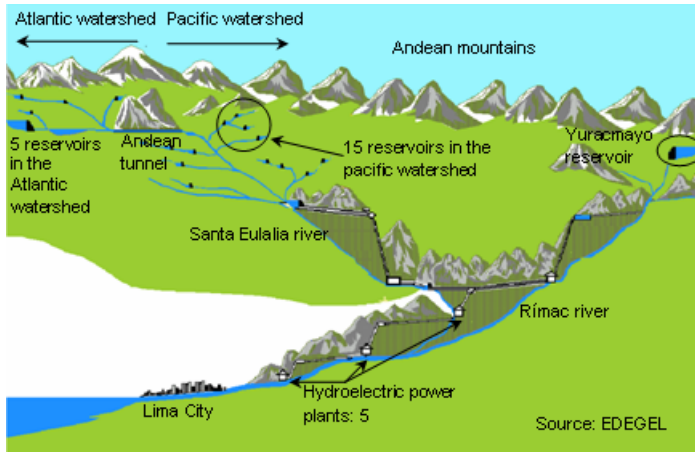
Fig. 1 Water reservoir system of the Rimac river catchments (adapted from EDEGEL)

During the wet weather season in the Andeans (from December to April), the reservoirs are refilled. At the end of this period, all 21 reservoirs are nearly completely filled. During the subsequent dry season (from Mai to November), the stored water has to cover most of the water demand of Lima as well as the water needs for the energy production by hydropower plants. Responsible for the whole reservoir system management are SEDAPAL (water company of Lima) and the electricity company EDEGEL S.A.A. (which operates also the hydropower plants). During the dry weather season, the water releases of the reservoirs are agreed upon by both companies by using a general discharge plan, which includes operation procedures and operating policies. The decisions on the water releases of the total reservoir system are additionally supported by long term experiences of the employees, a model based precipitation prognosis and a simplified model approach (see Fig. 2).

The target is to achieve a predefined storage volume at the end of November in order to ensure a complete refill of the system during the subsequent wet weather period. For normal and humid hydrological years, the target volume is set to 140 Hm^3 and to a value between 70-80 Hm^3 for dry hydrological years. Due to the fast growing number of inhabitants of Lima and due to the expected climate changes (assumed less precipitation during the wet weather period), the established approach of decision making is believed to be insufficient in the near future. In order to support operation and management of the reservoir system, a detailed model for the operation of the reservoir system, which includes control algorithms for water releases, will be set up.
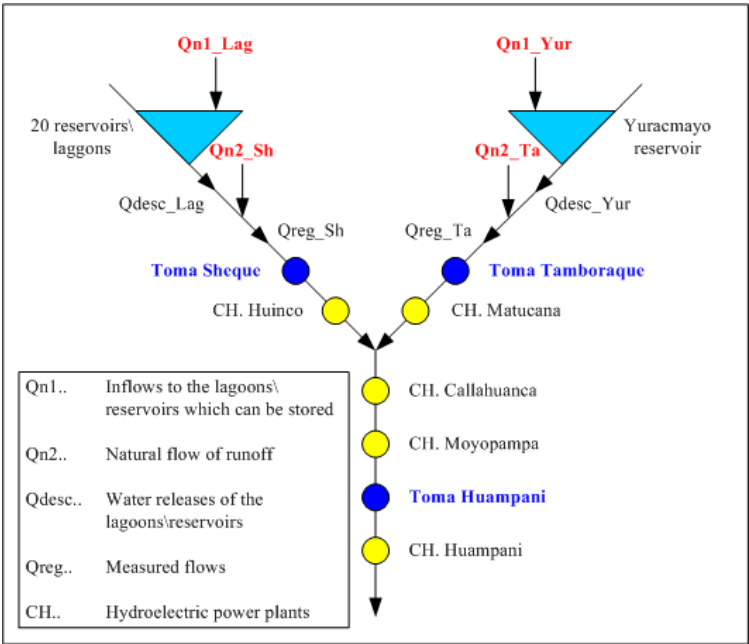
Fig. 2 Simplified diagram of the EDEGEL water supply reservoirs system (adapted from EDEGEL)

2 Operation of Reservoir Systems

In general, the management and operation of reservoir systems has to consider the effects of external drivers (e.g. climate conditions) and social, economical, political and environmental restrictions. An efficient operation has to consider different, and often competing, objectives (e.g. water supply, flood protection, energy production, ecological demands like fish migration, etc.). This requires the definition and implementation of a set of operation procedures, operating policies and schedules or management plans (Draper and Lund 2004). Operating rules and optimisation methods have been successfully applied not only for single reservoir systems, but also for complex multiple reservoirs systems organised in series, in parallel or a combination of both (see Trott and Yeh 1973; Becker and Yeh 1974; Murray and Yakowitz 1979; Chu and Yeh 1978; Yeh 1985; Karamouz et al. 1992; Esat and Hall 1994; Wardlaw and Sharif 1999; Hashemi et al. 2008). For many practical situations (e.g. real-time reservoir operations, planning stage of proposed reservoirs), operating rules are also established (Tu et al. 2003). Lund and Guzman (1999) reviewed single-purpose operating rules for reservoir systems in series and in parallel for water supply, flood control, hydropower, water quality and recreation. The proposed rules are useful for real time operation, supporting not only the simulation of reservoir systems operation in real time, seasonal

and long term, but also for better understanding of the functioning of reservoir systems. Draper and Lund (2004) and Brass (2006) mentioned reviews of reservoir operation rules that can be found in the literature, including the simplest reservoir operation rule, the so called standard operating policy (SOP). This determines the water releases of a reservoir as a function of the total water which is currently available. The target is to store as long as enough water is available. Otherwise, the amount of water available will be discharged.

The described operating rules represent predominantly a guide for reservoir system operation and can not be probably directly applied to real water reservoir systems, because each reservoir system is unique and has its own requirements and restrictions, see for example the reservoir system of the Rimac river basin presented in Section 1.

3   Modelling and Simulation of Reservoir Systems

Modelling and simulation are methods that are especially appropriate for examining, testing and evaluation measures for large and/or sophisticated systems. For modelling and simulation of reservoir systems many examples exist targeting the improvement of operation procedures. Labadie (2004) and Brass (2006) provide a review about the available computer simulation models. Many existing simulation models have been developed for particular systems, but many have also been developed for public use or general purpose (Labadie 2004).

For modelling and simulation of the reservoir system of the Rimac river catchments, the software package SIMBA (Simulation of biological wastewater system) has been used. SIMBA is an integrated dynamic system simulation program for sewer systems, wastewater treatment plants, sludge treatment and receiving water bodies. The software package is based on MATLAB/Simulink and has been developed by ifak Magdeburg (ifak 2009). In engineering practice and in research, the software package supports the analysis of the whole urban wastewater treatment. Several features of this software proved to be particular useful has been used for the present this work. The one is based on the module library for hydrological modelling of sewer networks. This has been adapted to modelling of reservoir systems. The other feature is the possibility to express control algorithm by using the industry standard IEC 1131-3 (IEC 61131-3 2002). This standard offers 5 programming languages for design and implementation of control algorithms. Especially the programming language Structured Text (IEC 61131-3 ST) is used more and more. The simple programming language Structured Text provides a good compromise of an easy-to-understand programming language and high flexibility and powerful capabilities to express control algorithms. It has shown to be useful in several applications of simulation and implementation of control concepts (Ogurek et al. 2008). By using this module, the operating rules of the discharge plan of EDEGEL could be integrated and implemented (see Fig. 3).

```
IEC 61131 Function Block (structured_text)                                    □ ▣ ✖

 1    FUNCTION_BLOCK default
 2    VAR_INPUT
 3       QInflow_5LagMantaro:              LREAL;  (*Inflows [m3/d]*)
 4       QInflow_Yur:                      LREAL;
 5       V_5LagMantaro:                    LREAL;  (*initial Water volume in m3*)
 6       V_Yur:                            LREAL;  (*initial Water volume in m3*)
 7       Qn2:She:                          LREAL;
 8       Qn2_Ta:                           LREAL;
 9    END_VAR
10    VAR_OUTPUT
11      T1_Qdesc_5LagMantaro:              LREAL; (*water releases in m3/d*)
12      T1_Qdesc_Yur:                      LREAL;
13    END_VAR
14    VAR
15       Vu_5Lag_Mantaro:                  LREAL:=157.09E6;(*Reservoir capacity in m3
16       Vu_Yur:                           LREAL:=48.30E6;
17    END_VAR
18
19    IF Qn2_She+Qn2_Ta>=Qreg_She+Qreg_Ta THEN
20       T1_Qdesc_5LagMantaro:=0.00;
21       T1_Qdesc_Yur:=0.00;
22    ELSE
23          IF Qn2_She<Qreg_She AND Qn2_Ta>=Qreg_Ta THEN
24             IF V_5LagMantaro>0.58 THEN
25                T1_Qdesc_5LagMantaro:=0.65*(Qreg_She-Qn2_She); (*prioridad 1, lagun

Debug mode: ☐

Sample time [d]: -1                                        New   Import ...  Export ...
```

Fig. 3 Some operating rules of the discharge plan in the IEC Module

In addition, for the determination of the accumulative monthly energy production in the selected years, an energy equation, relating flow to generate energy applied by EDEGEL, has been integrated into the model.

4 Modeling and Simulation of the Water Reservoir System of the Rimac River Catchment

Modelling, simulation and control of the reservoir system has been carried out with the program SIMBA, described above, taking into account the discharge plan of EDEGEL. Due to the complexity of the system and for a first test of the control of the system, considering the real operating rules, a simplified model of the real system has been set up. The simplification of the system consisted mainly in the division of the entire reservoir system into 4 groups of reservoirs (5 reservoirs of the Atlantic watershed, 9 and 6 reservoirs of the Pacific watershed and the Yuracmayo reservoir, see Fig. 4), a more detailed schema than the simplified water reservoir schema of EDEGEL presented in the Fig. 2. The structure of the simplified water reservoir system modelled is presented in the Fig. 4.
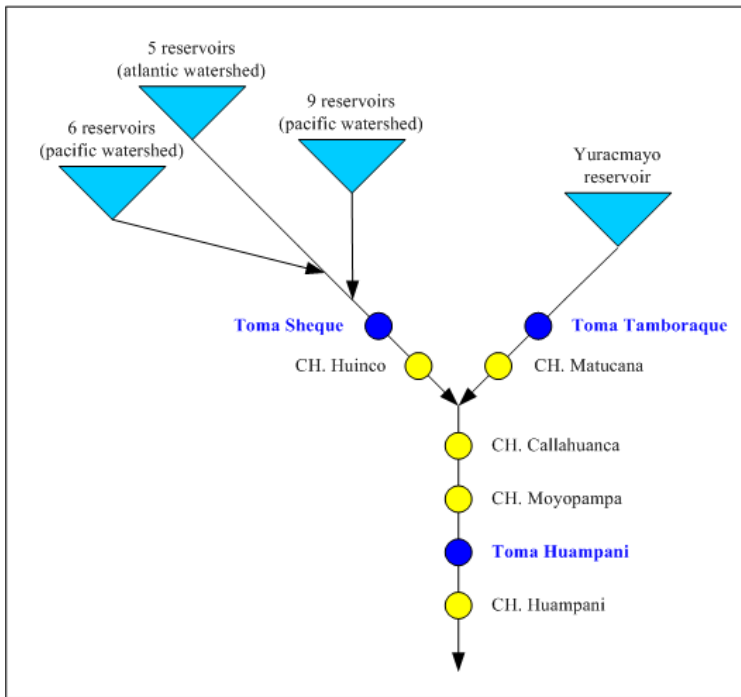
Fig. 4 Simplification of the real water reservoir system (source Godoy 2011)

4.1 Selection and Simulation of Two Past Years as Basis Cases

In order to create base cases of the past operation of the system and to test the integrated operating policies of the discharge plan for the control of the system, a dry hydrological year (2004) and a wet hydrological year (2009) have been selected. The results of the simulation of the selected years have been compared with the historical data of EDEGEL. The compared results were the average monthly water releases of the reservoirs, stored water volume in the reservoirs at the end of each month, the monthly average regulated flows measured in the measuring points "Toma Sheque" and "Toma Tamboraque" (see Fig. 4) and the monthly accumulative energy production in each hydroelectric power plant.

4.1.1 Data Base and Input Data for the Simulation of the Basis Cases

For the simulation of the water reservoir system, the historical data of EDEGEL for the year 2004 and 2009, as well as most of the operating rules of the discharge plan have been considered. In addition, in order to achieve a better approximation of the real management of the reservoir system, the implementation of additional operating rules has been carried out. The additional operating rules have been based on the historical water releases of the reservoirs between 1997 and 2009.

Additional input data for the simulation were: monthly average of natural inflows to the reservoirs (Qn1_5lag, Qn1_9lag, Qn1_6lag and Qn1_Yur), monthly average of natural flows of runoff (Qn2_Ta, Qn2_Sh), initial stored water volume in the reservoirs, the accumulative monthly energy production and the monthly average regulated flows measured in the measuring point "Toma Sheque" and "Toma Tamboraque" (Qreg_She, Qreg_Ta). The reference values for the regulated flows were 20 m³/s (January - April and December) and for the months from May to November, the values of the regulated flows measured in the considered years.

5   Results and Discussion

The results of the simulation of the base cases for the year 2004 and 2009 are listed in Tables 1 and 2. The results include the monthly average discharges of the 5 reservoirs (Qd5lag) of the Atlantic watershed, the discharges of the 9 and 6 reservoirs (Qd9lag, Qd6lag) of the Pacific watershed and the discharges of Yuracmayo reservoir (QdYur), the regulated flows measured in the measuring points "Sheque" and "Tamboraque" (QregSh, QregTa), the stored water in the reservoirs (V5lag, V9lag, V6lag and VYur) and the total accumulative monthly energy production. (E. prod.)

Table 1 Simulation results (year 2004)

Year 2004	J	F	M	A	M	J	J	A	S	O	N	D
Qd5lag (m³/s)	4,4	1,4	0,0	0,0	1,5	4,0	4,6	4,9	5,7	5,1	3,2	0,7
Qd9lag (m³/s)	0,7	0,2	0,0	0,0	0,2	0,6	0,7	0,7	0,9	0,8	0,5	0,1
Qd6lag (m³/s)	1,7	0,5	0,0	0,0	0,6	1,5	1,8	1,9	2,2	2,0	1,2	0,3
QdYur (m³/s)	1,4	0,0	0,0	0,0	0,0	0,5	1,2	1,7	2,0	1,9	1,4	0,2
Qregsh (m³/s)	9,8	10,1	11,8	10,1	9,1	9,2	9,7	10,0	10,1	9,6	9,2	9,6
QregTa (m³/s)	11,3	13,7	16,3	14,0	10,9	8,4	7,3	6,6	6,6	7,4	9,8	13,2
V5lag (Hm3)	64,3	72,6	94,0	117,5	128,5	128,3	126,5	122,6	114,3	108,7	108,4	115,4
V9lag (Hm³)	4,2	6,2	9,8	12,2	12,9	12,2	11,5	10,4	8,9	8,0	8,7	11,3
V6lag (Hm³)	17,1	20,0	24,7	28,4	30,7	31,1	30,5	27,4	22,6	18,5	18,1	21,8
VYur (Hm³)	11,8	18,2	26,4	29,9	33,1	34,0	32,1	28,6	24,2	21,1	21,5	28,4
E. prod. (GW)	265,0	253,1	304,4	277,3	255,7	222,9	222,8	218,6	212,6	222,1	243,0	282,5

In order to evaluate the simulation results presented in Tables 1 and 2, a comparison between the obtained results with the measured historical data of EDEGEL has been made. Figures 5, 6, 7, 8, 9 and 10 show a comparison between the water releases, stored water volume of the 20 reservoirs (the sum of the results of the 5 reservoirs of the Atlantic watershed and the 15 reservoirs of the Pacific

Table 2 Simulation results (year 2009)

Year 2009	J	F	M	A	M	J	J	A	S	O	N	D
Qd5lag (m³/s)	0,0	0,0	0,0	0,0	0,7	4,1	4,7	5,2	6,1	6,2	4,4	0,9
Qd9lag (m³/s)	0,0	0,0	0,0	0,0	0,1	0,6	0,7	0,8	0,9	0,9	0,7	0,1
Qd6lag (m³/s)	0,0	0,0	0,0	0,0	0,3	1,6	1,8	2,0	2,4	2,4	1,7	0,4
QdYur (m³/s)	0,0	0,0	0,0	0,0	0,0	0,1	0,7	1,7	2,3	2,1	1,3	0,1
Qregsh (m³/s)	12,8	18,6	32,9	32,2	19,8	12,0	11,1	11,0	11,9	12,8	13,8	14,8
QregTa (m³/s)	16,9	23,3	28,6	25,8	20,6	13,6	10,1	9,3	9,2	9,5	11,9	16,8
V5lag (Hm3)	106,8	136,3	157,1	157,1	157,0	151,0	141,2	129,8	116,9	106,0	103,7	122,7
V9lag (Hm³)	10,4	15,5	19,6	19,6	19,6	19,6	19,6	18,8	17,4	16,2	17,2	19,6
V6lag (Hm³)	31,0	38,7	48,2	56,8	57,4	56,9	54,9	51,2	45,5	40,5	38,8	44,0
VYur (Hm³)	16,3	25,9	37,6	45,8	48,3	48,3	48,0	45,0	40,7	36,9	37,0	45,7
E. prod. (GW)	310,3	319,3	395,1	387,4	360,2	287,1	267,1	260,8	259,4	276,9	296,5	328,9

Fig. 5 Comparison between the simulation results and the measured monthly average water discharges of the 20 reservoirs (year 2004)

watershed) and of the Yuracmayo reservoir and the monthly accumulative energy production and the measured historical data for the years 2004 and 2009. It can be seen from the figures that the operating rules implemented in the model result in a good approximation of the real operation of the system. The obtained simulation results follow, in most of the months, the trend of measured data in the reservoir system. Some observed peaks (e.g. water discharges of the 20 reservoirs in the month of April, Fig. 6.) could not be simulated, because these are the results of spontaneous decisions taken by the managers or by the operators of the system.

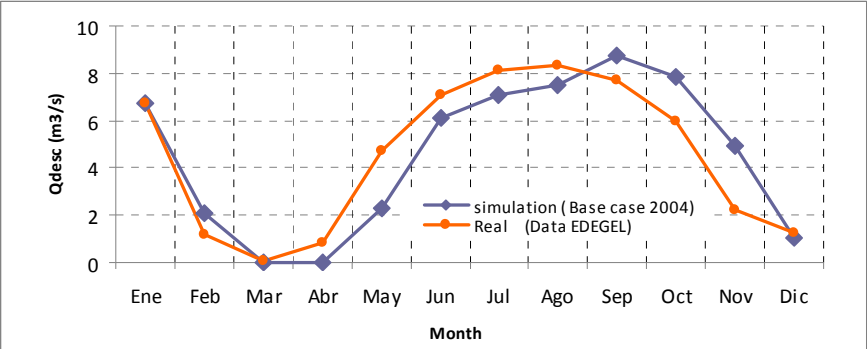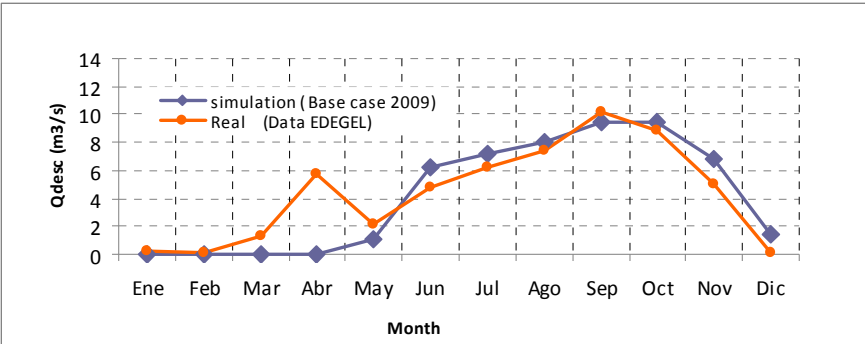**Fig. 6** Comparison between the simulation results and the measured monthly average water discharges of the 20 reservoirs (year 2009)
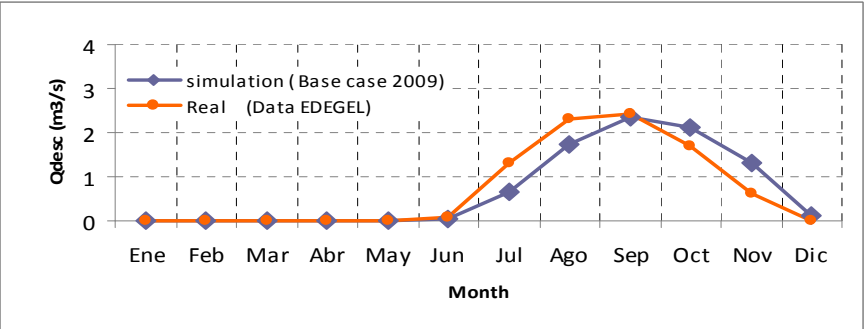
Fig. 7 Comparison between the simulation results and measured monthly average water discharges of the Yuracmayo reservoir (year 2009)
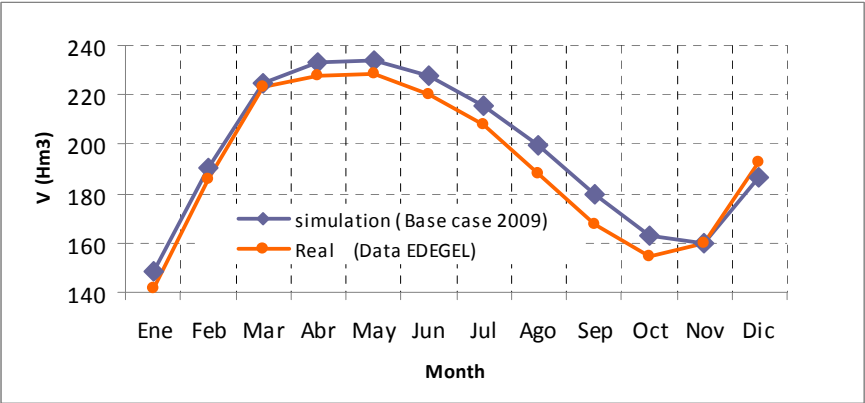
Fig. 8 Comparison between the simulation results and the measured data of stored water volume in the 20 reservoirs (year 2009)
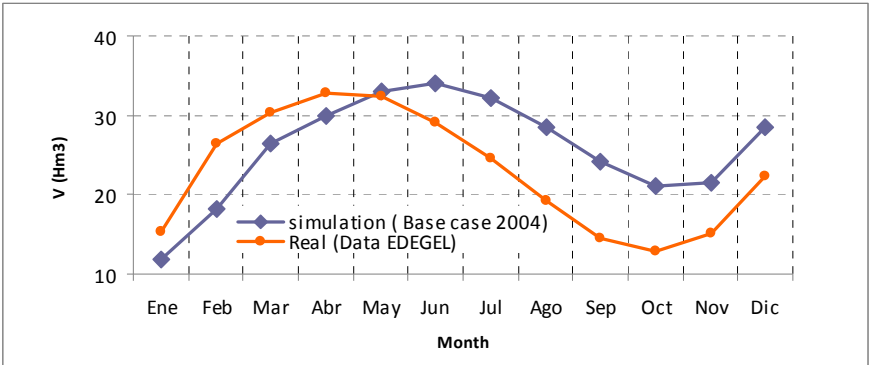
Fig. 9 Comparison between the simulation results of stored water volume in Yuracmayo reservoir and the measured data (year 2004)
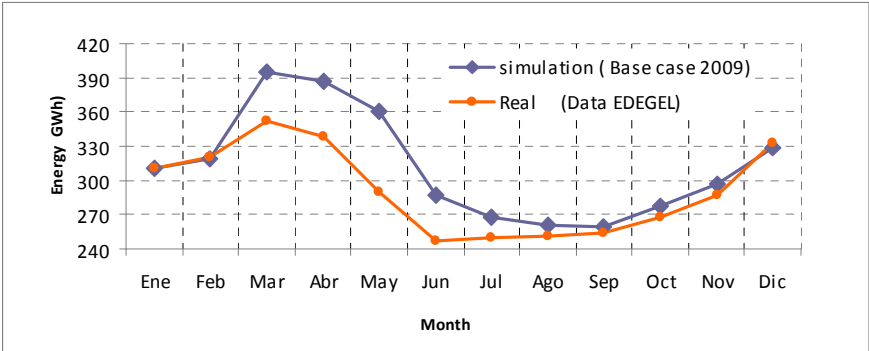
Fig. 10 Comparison between the simulation results of monthly accumulative energy generation and the measured data (year 2009)

6   Conclusions

Taking into account the simulation results obtained, the control of the reservoir system using the operating rules (of the discharge plan and the additional operating policies) integrated into the model provide a first insight into the operation of the reservoir system for the years under consideration (2004 and 2009). The general behaviour of the operation was represented well. Some peaks caused by specific or spontaneous decisions could not be simulated, because they are not included in the normal operating discharge plan. Nevertheless, the simulation could validate the modelling of the discharges in the Rímac river basin and the water volume stored in the reservoir system. Furthermore, a good approximation of the monthly accumulative energy generation has been achieved.

In order to validate and to adapt the operating rules integrated into the model, the simulation of other periods with different conditions (e.g. semi-dry years, normal years, semi-wet years) is considered necessary. Also, the potential to improve the operation of the reservoir system will be analysed. Furthermore, operation of reservoir system will be put in context with the overall water management of the megacity of Lima (Schütze et al. 2011).

References

Becker, L., Yeh, W.: Optimisation of Real Time Operation of a Multiple Reservoir System. Water Resources Research 10, 1107–1112 (1974)

Brass, C.: Betriebsoptimierung von Talsperrensystemen mittels Stochastisch Dynamischer Programmierung (SDP) unter Berücksichtigung veränderlicher Ziele und Randbedingungen. Dissertation, Ruhr-Universität Bochum (2006)

Chu, W., Yeh, W.: A Nonlinear programming algorithm for real time hourly reservoir operations. Water Resources Bulletin, American Water Resources Association 14, 1048–1073 (1978)

Draper, A., Lund, J.: Optimal Hedging and Carryover Storage Value. Journal of Water Resources Planning and Management 130, 83–87 (2004)

Esat, V., Hall, M.: Water resources system optimisation using genetic algorithms: Hydroinformatics '94. In: Esat, V., Hall, M. (eds.) Proceedings of the First International Conference of Hydroinformatics, Balkema, Roterdam, The Netherlands, pp. 225–231 (1994)

Godoy, E.: Modelling and control of the control reservoir system of the Rimca river catchment, Final Report (2011)

Hashemi, M., Barani, G., Ebrahimi, H.: Optimisation of Reservoir Operation by Genetic Algorithm considering inflow probabilities (Case Study: The Jiroft Dam Reservoir). Journal of Applied Sciences 8, 2173–2177 (2008)

IEC 61131-3, Programmable controllers - Part 3: Programming languages, 2nd Edn., International Electrotechnical Commission, Genf (2002)

ifak, SIMBA6 – Simulation of Wastewater systems, Reference and Tutorial, ifak – Institut für Automation und Kommunikation e. V. Magdeburg, Germany (2009)

Karamouz, M., Houck, M., Delleur, J.: Optimisation and Simulation of Multiple Reservoir Systems. Journal of Water Resources Planning and Management 118, 71–81 (1992)

Labadie, J.: Optimal operation of Multireservoir Systems: State-of-the-Art-Review. Journal of Water Resources Planning and Management 130(2), 93–111 (2004), doi:10.1061/(ASCE)0733-9496

Lund, J., Guzman, J.: Some Derived Operating Rules for Reservoirs in Series or in Parallel. Journal of Water Resources Planning and Management 125, 143–153 (1999)

Murray, D., Yakowitz, S.: Constrained Differential Dynamic Programming and its Application to Multireservoir Control. Water Resources Research 15, 1017–1027 (1979)

Ogurek, M., Alex, J., Schütze, M.: Simulation als Basis für Entwicklung, Test und fehle-
      rarme Inbetriebnahme von Automatisierungssystemen komplexer Prozesse, EKA 2008,
      Magdeburg (2008)
Schütze, G., Robleto, G., León, C., Rodriguez, I.: Modelling and scenario building of urban
      water and wastewater systems – Addressing water shortage in Lima. In: IWA/IAHR In-
      ternational Conference on Urban Drainage, Porto Alegre (September 2011)
Trott, W., Yeh, W.: Optimisation of Multiple Reservoir Systems. Journal of Hydraulics Di-
      visions 99, 1865–1884 (1973)
Tu, M., Hsu, N., Yeh, W.: Optimisation of Reservoir Management and Operation with
      Hedging Rules. Journal of Water Resources Planning and Management 129(2), 86–97
      (2003), doi:10.1061/(ASCE)0733-9496
Wardlaw, R., Sharif, M.: Evaluation of genetic algorithm for optimal reservoir system op-
      eration. Journal of Water Resources Planning and Management 125, 25–33 (1999)
Yeh, W.: Reservoir Management and operation models: A State of the Art Review. Water
      Resources Research 21, 1797–1818 (1985)

Non-recursive Interference Calculi - A Mathematical Calculus Immanent in Nervous Activity

Gerd Karl Heinz

Abstract. Interference networks (IN) and interference systems have a comparable mathematical-physical background, reaching from photonic wave interference in optics over signal interference in digital filters (FIR, IIR), wave interference in Radar- or Sonar- devices to ionic pulse interference in nerve nets. Special properties of IN are short wavelength, relative timing and non-locality of function. Behind concepts of cybernetics and informatics, we find in interference integrals a hidden functional principle for nerve nets, the non-locality of function. The paper highlights integration methods in non-recursive IN. It reflects on simulation movies on the web [9].
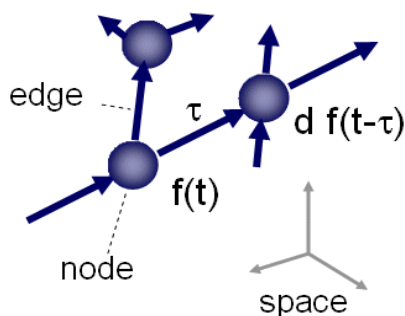
Fig. 1 Non- recursive interference network (IN). Any time function $f(t)$ arrives a next node delayed by τ and damped by d in a form $d*f(t-\tau)$.

Any process, that uses as input a set of time functions (acoustic, ionic, electric or photonic) and delivers as output 'images' or 'maps' denotes an interference process. The paper observes interference networks without feedback between nodes (non-recursive IN); the edges carry mono-directional time-functions.

The idea of modeling such processes bases on the conjunction (addition/multiplication) of spatial delayed, *synchrotopic* time functions at locations or nodes, and the integration of interference values. Synchrotopy combines in Greek the words *syn-* for together, *chronos* for time and *topos* for location [7].

Gerd Karl Heinz
Gesellschaft zur Förderung angewandter Informatik e.V. (GFaI)
Volmerstr.3, 12489 Berlin
e-mail: heinz@gfai.de

In difference to solutions of differential equations at single nodes (wave equation) the interest of IN is to understand the function of networks.

In interference systems, the type of operation, the type and parameters of time-function, the type of non-linear exponation or integration and the current channel number 'interfere'.

Various works showed properties of interference integrals ('wave images'): In acoustic imaging [4], for nerve nets [8], for high-speed microcontrollers [7]. Simplicity of the element-by-element integration process on the one hand; and complexity to find possible parameterizations on the other constrained earlier publication.

We suggest a simplest model. It is a non-recursive conjunction without feedback; delays shift time- and space-functions without feedback.

Reflecting McCulloch/Pitts paper [12] "A logic calculus immanent in nervous activity" interference systems or networks have to contain more then a logic calculus; physical aspects (delaying wave space) and additional mathematical aspects (integration – this paper) play an important rule too.

Analyzing interference nets, we find different philosophical aspects compared to computer science and neuroscience. While state machines (Boolean nets, also McCulloch/Pitts) use concepts, that *associates a 'function' or 'signal' physically to an output of a cell or gate*, wave spaces and interference nets show a different functional organization and mapping. Thinking about Radar maps or acoustic images, the *relativity of timing of signals defines the function at a certain place; a function does not necessarily associate with the output of a gate or cell*. Function has no locality. Any gate or cell output can carry very different signals in sequence. Non-local functionality needs something like a dense gate-ground (the retina for example). This seems to be the main aspects to clarify retarding progress: Every philosophic assumption about nerve nets up to now assumes a computer-like binding of function to an axonal output like "PORTB3 - LED on/off".

 "A profound revolution lurks in our basic concept of how the information-bearing elements of the nervous system communicate." T.H. Bullock, [1]

Inspecting a homogeneous photonic wave field on a photo plate, we can shift the plate before exposure – the resulting (wave-) image after exposure is independent of the shift and of the exact location of the net parts.

If we change the position or location of some pins of a bus in our PC, the system would not work further. For the airplane-helicopter crash [7], the absolute place of the crash plays no rule. If machines start and fly with identical relative timing, motion, wind, height and direction, they could meet again.

In IN the type of conjunction (multiplicative or additive), the time function type (positive or symmetrical), the way to calculate the detection (integration or averaging), the time function properties (single pulses or sinoidal forms) and the channel numbers influence the properties of integrals (maps, images) [7].

For example, multiplication of symmetrical time functions works best for two, but not for more channels. Additive conjunction is just satisfying for many channels but works not satisfying for two.

Zero Delay Prohibited

Any incoming time-function $f(t)$ leaves the edge delayed by τ and damped by a factor d (mostly $d=1$)

$$f(t) \quad \rightarrow \quad d*f(t-\tau) \tag{1}$$

If not otherwise noted, the edge delays the transmitted signal. Each node remarks a location with a spatial position or location within the network, defining the delay properties of the net. Information needs time to bridge spaces. Infinite fast transmission (without a physical delay) is impossible. In opposite to electric schematic drawings, where edges mark equipotential connections (electric nodes), interference nets have only edges with (distributed or fixed) delays. Comparable to electrics, the equipotential assumption of nodes is valid only within a node. Every edge has a delay. By analogy, non-local calculations are impossible. For example, the convolution integral needs delayed inputs. The IN-representation splits into different nodes and external delays.

Space- and Time Function, Wave Function

Public wave function definition (Wikipedia) is different to the model proposed here. Things are much simpler. We suggest only a dense meshed network and any time function that can freely expand into all directions.
Time function and space function show different views of the same thing:

- A time function $g(t-\tau)$ is valid for a single point (oscillogram).
- A space function $\psi = f(vt-r)$ is valid for a single time step (ocean wave).

Space and time function join by analogy with velocity $v = r/\tau$, delay τ, radius r.
If a time function is leaving one-dimensional space, it becomes a wave. Main characteristic of a wave is that all points of a certain peak have the same delay to the source location.

Only the calculation of radius r in space function $\psi = f(vt-r)$ has to change to show one- or higher dimensional waves. In $r = x-x_0$ the index x_0 denotes the locations of the source point of wave, while x denotes the actual location. For Euclidian space for example the radius r is the well known Euclidian distance

$$r = x-x_0 \qquad\qquad\qquad \textit{one dimensional space} \tag{2}$$

$$r = \sqrt{(x-x_0)^2 + (y-y_0)^2} \qquad \textit{two dimensional space} \tag{3}$$

$$r = \sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2} \quad \textit{three dimensional space} \tag{4}$$

If we follow a single peak of the time function through any space, we find it as a wave [8]. Consequently, a wave characterizes nothing more and nothing less, then a space-function in spherical dimension higher one.

If we inspect any system with the property to delay signals flowing through, we can regard the signals to be wave-like. The form of expansion of waves depends on the spherical measure or norm [7]. Euclidian space geometry produces round waves. Manhattan geometry shows squared waves. Inhomogeneous space geometry of nerve nets shows broken, distorted or discontinuous waves [10].

Independent of the measure, all nodes (pixel etc.) under the wave $\psi = f(vt-r)$ have a common property: they have the same delay distance (delay) to the source point of the wave $f(t)$.

To finish the wave concept, any single node can overlay different waves coming from different origins [4]. We find this kind of wave propagation in various types of substrates for different carriers of waves; we find it in air or water (electric-,

Fig. 2 Example of a 2-dim. Euclidian wave function $\Psi = f(r,t) = f(vt-r)$. With $v=r/\tau$ it relates to time functions waves of type $g(t-\tau)$.

photonic-, acoustic waves), on integrated circuits (electric waves) or in nerve nets (ionic waves). By contrast, Andrew Packard's squid experiments on nerve nets [13] had shown that nerve systems do not mesh all nodes necessarily to all sources. Inspecting his records it seems, we find yellow and red waves interacting independent of black waves. We talk here about different layers.

Suggesting a dense meshed network with intrinsic delays between nodes, if we run the parameter vt, the waves come in motion. Running it forward, the wave runs forward with wave peak in front and wave tail back. If we run the time parameter backward, the wave runs backward showing the source point. Using a negative delaying substrate [4], we get waves going inward with tail outside.

Compare with examples and movies on the web page [9]. Under the movies lays the Scilab [14] source code.

Race Circuits

If the geometric wave length $\lambda = vT$ comes in the range of the size of a circuit or arrangement, we talk about 'race circuits'. Interference nets are race circuits.

Functions of race circuits need any kind of integration processes. If we use a paper sheet instead of the photo plate, after exposure no image remains. The photo image needs integration over lots of waves to form the interference integral – the image – within the light sensing material of the photo plate.

In case of electronic race circuits, race conditions occur in the range of femtoseconds for each state. Radar, electric interferometers, reflectometers or femtosecond-meters use the idea in several ways. Each pixel of any optical image shows

properties of interference integration. Fast integrated circuits show more and more the problem of race conditions, because each single wire delays the carried signal relative to the length. Last not least acoustic camera or Sonar uses the interference of sound waves.

Delays and Waves

The main important property of interference systems is the restriction that every signal needs time to reach a destination at a different location [5].

For the exposure of a photo plate, a small deformation of the delaying wave space by a minimal mismatch of orthogonal direction to the axis destroys the image.

The delay assumption transforms a simple time function into a *wave function* [7]. The interference network becomes a *wave space*. We talk about homogeneous or inhomogeneous wave spaces.

While nerves need inhomogeneous wave spaces, acoustic, optic or electric net models use mostly a homogeneous, Euclidian geometry. Dependent of application we find different distance measures for delay, for example Euclidian, Manhattan or chaotic (nerves) [7].

Node Structure of Non-Recursive Nets

Independent of network type the signal processing subdivides into the processes of input weighting, conjunction and transfer.

Separation of interference network nodes into the three parts weighting, conjunction and integration allows descriptions of different processes and types of nets: ionic, electric, photonic or acoustic nets. Modeling the dendritic tree of a nerve, the edge between weighting and conjunction can also have a delay. The conjunction- operator Ψ symbolizes a substitute for possible conjunctions of time functions in interfering systems of different kind like $\Psi = \{\Pi, \Sigma, \Lambda$ (AND), V (OR), max, min, exp$\}$. A general model of node function [6, 7], that fits some cases, is
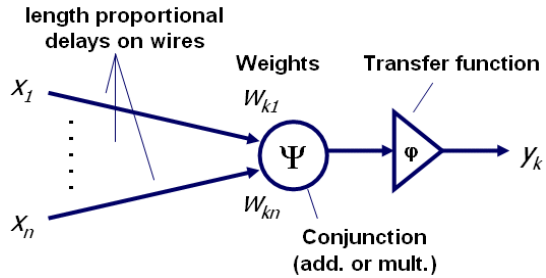
Fig.3 Net node structure

$$y_k(x,y,z,t) = \varphi(\overset{n}{\underset{j=1}{\Psi}} w_{kj} x_j(t - \tau_{kj})) \tag{5}$$

Transfer functions can be fuzzy sets, threshold functions or integral functions.

Interaction of Time Function and Processing

It might be trivial to underline, that the type of time function (bipolar-symmetric or monopolar-asymmetric/positive) has influence on possible calculations.

Multiplication of bipolar signals on more then two channels is impossible. Using many channels and multiplication, any negative part swaps the interference integral. Hence, computation of many unipolar signals is restricted to addition (acoustic cameras) or to multiplication of monopolar signals (nerve system).

By opposite, addition produces integral values also, if only a single time function has a value and all other deliver zero. For acoustic imaging this means that every still location in a resulting map gets a residual interference integral value.

Using two channels, multiplication has interesting properties. It is in common use for signal processing (modulation, frequency doubling etc.) and for any kind of computer algebra (logical AND).

Last not least we know from Fuzzy Sets, the kind of conjunction (addition – multiplication) influences the probability, that any stochastic computation shows the maximum (addition, OR) or the minimum (multiplication, AND).

1   Interference Detection: Integral or Average?

Signal processing and electronics use a wider terminology for 'integration' compared to integrals in mathematics. In electronics, we call any accumulation of values an integration process. To register and count very fast and small spikes, it is possible to observe integrals. Using integrative solutions we differ between two general types of interference detection, that is

- Interference integration and
- Interferential averaging.

While acoustic cameras use an averaging process to reconstruct maps with dB-values, any photo film or plate shows an integral behavior. As longer is the exposure time, as brighter becomes the film.

Anti-Parallel Addition of Waves - d'Alembert Waves

A first public solution of the wave equation shows d'Alembert's formula. However, the solution restricts interference to a one-dimensional principle with anti-parallel data flow. It is dedicated to analyze concurrent edges and violates restrictions for non-recursive interference nets.
A solution of the one-dimensional wave equation

$$\frac{1}{c^2}\frac{\delta^2 u}{\delta t^2} = \frac{\delta^2 u}{\delta x^2} \tag{6}$$

is d'Alembert's formula, for example in the form

$$u(t,x) = u_1(ct+x) + u_2(ct-x). \tag{7}$$

Two time-functions $u(x,t) = u_1(x,t) + u_2(x,t)$ interfere here on a one-dimensional sphere (wire, axon etc.). Neglecting the details, d'Alembert's formula allows also this solution. The public solution $u(x,t) = u_1(x+ct) + u_2(x-ct)$ shows one forward $u_1(x+ct)$ and one backward traveling wave $u_2(x-ct)$ [9]. The wave trail runs there in front of the wave. We use the form with trails at the end of waves.

Any interference integral appears only, if we introduce a non-linearity that enforces the place of wave meeting. Integration of added time functions shows always a *residual interference* value at locations, where one of the time functions is zero. That means, the integral value at any location tends to increase independent of other properties.

Multiplication of Waves

Multiplication of positive time functions removes the integral at all locations, where one of the time functions is zero. The approach

$$u(t,x) = u_1(ct+x) * u_2(ct-x) \tag{8}$$

has special potential for demonstrational purposes [9] and in signal processing, for example for modulation, demodulation or coding (AM, FM, GPS). Using single waves, the interference integral appears at the place of wave meeting, see animations. If one of the contra-directional flowing time functions comes later, the meeting place of time functions shifts to a location in direction of the delayed function.

2   Pixel Calculus and Time-Parallel Data Flow

To overcome the restrictions of one-dimensional contra- directional waves, we apply a node-convention. In opposite to anti-parallel calculation it simplifies the task. The node abstraction suggests, that spherical arriving waves show a time-parallel data flow. It allows the calculation of waves in *n*-dimensional networks.

Fig. 4 Time-Parallel Data Flow (Pixel Calculus). The example demonstrates the time parallelism: For numerical addition we can write $y1 = f3+g4+h2$, $y2 = f4+g5+h3$, $y3 = f5+g6+h4$ etc. In Scilab we write only $y = f+g+h$, if y,f,g,h are equal sized vectors.
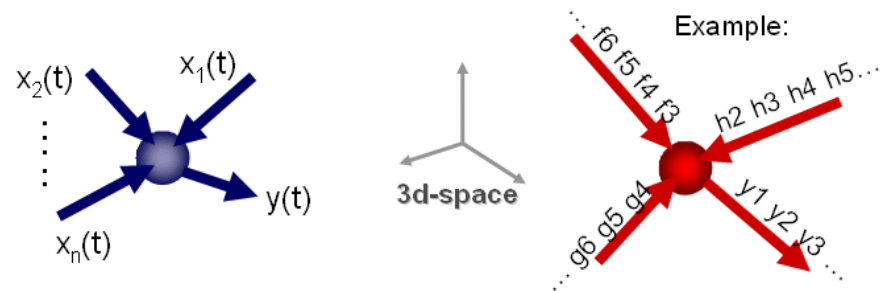
At each node of the IN, the *multi-directivity in space* corresponds now to a *parallelism of data flow in time dimension.*

If time functions are lists, they relate to element-by-element computation. Scilab [14] writes element-by-element multiplication of vectors with dot-star, for example $c = a .* b$.

3 Examples for Interference Systems

3.1 Open Integration - Photography, Ultra Short Time Measurement

Supposing, two spiking time-functions (*a*, *b*) in the range {0…1} with zero-level {0} and information level {1} meet in a way comparable to Fig.1. The gate inputs are a_i and b_i, the gate delivers the output $x_i = a_i\&b_i$ (Boolean AND-operation), *i* is the gate index.

Fig. 5 Example of a Boolean race logic with additional delay difference 2τ (!) and integrator (compare with data set in Tab.1)

Resulting x_i spikes apply in electric interference applications generally to short, to register them. To get a possibility for registration, we have to integrate over x_i to get a higher *accumulation y_i*. The slope adds each result of x_i to y_i (integration) for each location or net node (we have three here):

$$x_i = a_i \& b_i \quad \Rightarrow \quad y_i = y_i + x_i \quad \textit{(integration)} \tag{9}$$

The value of y_i shows the interference integral level. It shows, how many Ones meet at the gate.

To see the effect of any change in position, the τ in Fig.1 delays the related time function *a* or *b* by two time steps, Tab.1. We suppose infinite time function sequences. Any vector *a*, *b*, *x*, *y* is a small part of a infinite long sequence. The normal position of gates may be in the middle (x_2, y_2). Additional delays mark the left or right positions (x_1, y_1) and (x_3, y_3).

Left and right positions can symbolize two cases:

- one of the time functions *a* or *b* is delayed (here by two steps)
- the middle gate moves the position to left or right (with additional delay)

Both cases are equivalent. Each 'hit' of Ones in (*a=1* & *b=1*) produces a result $x = 1$ and increases the respective accumulator *y* by one.

Using a time function *a* and a time function *b* that shifts on the time scale (left/right) by two steps ($\tau \sim n = 2$) relative, the effect of delay shift on the interference integral can be studied by example, Tab.1 and Fig.1.

In case of maximum interference of Ones between *a* and *b* (middle position, x_2) the accumulation is maximal ($y_2 = 6$). A variation of delay between inputs *a* and *b* by two steps shows, that the interference integral value *y* varies (y_1, y_3). The value of the accumulator y_i marks the number of hits.

Table 1 Interference integrals of two time functions. Example for an 'open' integration process for logic gates (data set for Fig.4)

Direction of time flow →

```
a(t)    =[ 0 0 0 1 0 0 1 1 1 0 0 0 0 0 1 1 0 0 0 ]
b(t-2)  =[ 0 0 1 1 0 0 0 0 1 1 1 0 0 0 0 1 1 1 0 ]
x₁      =[ 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 ]
y₁      =[ 0 0 0 1 1 1 1 1 2 2 2 2 2 2 2 3 3 3 3 ]  = 3

a(t)    =[ 0 0 0 1 0 0 1 1 1 0 0 0 0 0 0 1 1 0 0 0 ]
b(t)    =[ 1 1 0 0 0 0 1 1 1 0 0 0 0 0 1 1 1 0 0 0 ]
x₂      =[ 0 0 0 0 0 0 1 1 1 0 0 0 0 0 1 1 1 0 0 0 ]
y₂      =[ 0 0 0 0 0 0 1 2 3 3 3 3 3 3 4 5 6 6 6 6 ]  = 6

a(t-2)  =[ 0 0 0 0 1 0 0 1 1 1 0 0 0 0 0 1 1 0 0 ]
b(t)    =[ 1 0 0 0 0 1 1 1 0 0 0 0 1 1 1 0 0 0 0 ]
x₃      =[ 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 ]
y₃      =[ 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 ]  = 1
```

Photonic Interference and a Photonic Paradox

Inspecting light waves and their interference integrals (images) we find a paradox. The addition of many time functions at a detecting field (retina, photo plate, screen) produces in theory no interference integral, if we compare photonic waves with symmetric, sinoidal waves, see the movie in [9] (Sinoidales Integral bei Addition, #*sinadd*). If a single wave has at a certain location an integral value of zero, also the addition of many waves delivers zero. To get an image, we have to suppose any nonlinearity. The detecting field produces the solution: it has to transform the linear overlay by a nonlinear operation using the squaring terms 'energy' or 'intensity'.

Light waves interfere on a photo film, producing the image as interference integral. As longer is the exposure time, as brighter is the exposure. Film exposure appears proportional to shutter opening time.

Let *x(t)* be the sum of all light waves reaching the target place, the resulting time function *y(t)* of the film is the integral over time. Suggesting the time function *x(t)* of light consists of *periodic, symmetric waves* we know from DC-current, the *interference integral is always zero* [9]. Positive integral parts delete the negative,

$$y(t) = \int_{-\infty}^{\infty} x(t)\, dt = 0. \tag{10}$$

To get any exposure different to zero, the observed time function *x(t)* cannot be *periodic and area-symmetric* to time axis, see the animation films [9]. The interference integral *disappears for symmetric time functions*.

If the time function integral *y(t)* has to be different to zero, we have to suppose a non-linear rectifier (film, CCD, eye) within the chain, squaring the time function

$$Y(t) = \int_{-\infty}^{\infty} (x(t))^2 \, dt \neq 0. \quad \text{(energy, intensity)} \tag{11}$$

If the effect of the positive part of wave to any photo sensor or film is identical to the effect of a even high negative part, we talk about energy or intensity.

Overexposure of Photo Plates and Integral Divergence for Open Integration

As longer is the integration time, as more increases the value of any interference integral for a photo film. The value depends directly from the length of measuring interval. The interference integral accumulates to a *steady growing* time-function of the pixel or sensor. We will call it an *'open' integration process*. The accumulation grows to infinite for infinite long time or sequences. If not zero, the resulting integral is divergent in most practical cases

$$f(t) \rightarrow \infty \quad \textit{for} \quad t \rightarrow \infty. \tag{12}$$

If the Ones are interfering Planck photons, we have to stop the integration process after a certain time to avoid overexposure.
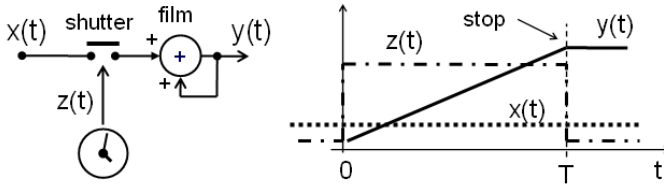
Fig. 6 Equivalent circuit for open loop integration of photographic films

Investigation of photography brought the *shutter* to stop overexposure of interference integration process on a photo plate. Open integration stops after the exposure time. The accumulator resets and the next image can start by opening again the shutter.

3.2 Integration with Loops – CCD, Nerves

To limit the level of exposure, we can stop the exposure at the exposure maximum of the sensor or we can downscale the sensitivity relative with the maximum of the integral, for example using a moving average algorithm. Processes use sometimes a servo loop; we call them 'loop integration'.

Stop-Loop Integration – CCD Camera Chipsets

Automatic exposure systems in CCD-camera chip sets stop the light exposure, if the first of the sensor pixels of the camera reaches the integration maximum. If $y(t)$ exceeds *max*, the comparator sends a one. If not, it sends a zero value to the output. The shutter closes the input for all pixels, if maximum (max) exceeds. The accumulator value of each pixel represents now the light intensity.
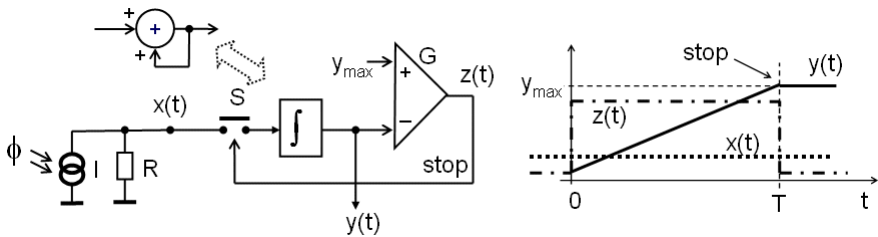
Fig. 7 Equivalent circuit for servo loop integration of CCD camera chip sets: photo-cell signal *x(t)* integrated signal y(t), comparator G. Shutter opens when *y(t)* exceeds the maximum value y_{max}.

After reading the pixel values by the computer, the next image exposure starts with a reset of the accumulator value to *y(t) = 0* (not drawn).

Pulse Frequency Integration (with loop) – Eyes, Nerves

Translating the open integration method to our eyes means, as longer we look on an object, as more lightness our eyes would get. We know this is not true. We can look hours on an object without overexposure of retina. Nerves and eyes use a different integration method.

While video cameras (CCD) conserve the accumulator until the computer has read it, by opposite nerve cells destroy the accumulator value just in the minute, the accumulator is full. It seems to make no sense.

But integration time reflects now the input current. A high input current (high optical flux) brings a short integration interval, a low current a long integration interval. In the minute, the accumulator value $y(t)$ at comparator *comp* is greater the maximum value y_{max}, the comparator output changes to high, resetting the accumulator.
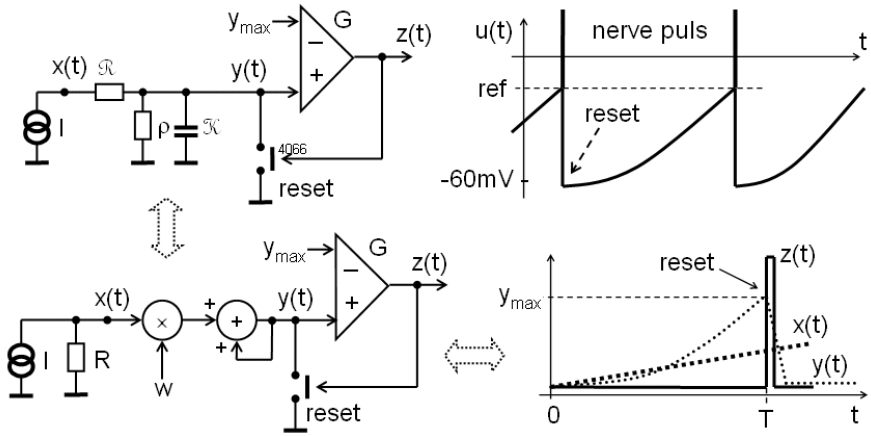
Fig. 8 Integrate-and-fire-neuron [2] as integrator in IN. First row: Lapique's R-C network of 1907 [11] enhanced by a comparator and a reset-loop. Right: Pulse principle of nerve. Second raw: Equivalent circuit with weighed input, defining pulse frequency. Right: Spike generation of the circuit.

While the accumulator clears, the comparator changes back to zero state. A small spike appears at the comparator comparable to an output spike of a nerve cell just in the minute, the accumulator is full.

If the input current loading the capacitor $K = C$ is i, the voltage over C grows with

$$\frac{du}{dt} = \frac{i}{C}.$$

(13)

The fire frequency f for constant i is for $du = y_{max} = u_{max}$:

$$f = \frac{1}{dt} = \frac{i}{du\ C} = \frac{i}{u_{max}\ C}$$

.(14)

The brilliant idea behind nerve function is, to output the *strength of accumulation as a duration or pulse frequency f*. In neuro-computing, His idea [11] later comes back as 'integrate and fire neuron', compare for example [2]. Although his idea is over 100 years old, the specific association of spiking behavior with *interference integration* and the properties to *avoid overexposure* and to *code the intensity to pulse length* seems not to be well known.

4 Averaging Integration Methods

We found, interference integration incorporates the danger of divergence of the result. Looking for optimum integration methods, we find a class of "means".

Denoted by Wikipedia[1] very different means plays a technical rule. We would limit the choice to typical interference methods.

The principle of 'mean' reflects the integration with reset loop. Reset appears, if the value is maximal. The number of samples reflects the current.

By difference mean methods add a number of samples and divide the result by the number. For a linear ramp, the result is identical to the area integral.

4.1 Means Used for Acoustic Cameras

The reconstruction process [4] adds first the delay-corrected channels k of microphones to get a single, averaging time function (bipolar) reconstruction $p_i(t)$. It represents the approximation of the original time function at the respective location i of the origin

$$p_i(t) = \frac{1}{n} \sum_{k=1}^{n} p_k(t + \tau_{ik}).$$ (15)

Integration in form of quadratic mean produces a single value $P_i(t)$ for each pixel (noise pressure as color) for a given time point t

$$P_i(t) = \sqrt{\frac{1}{n} \sum_{j=1}^{n} p_{ij}^2}$$ (16)

The equation adds for a given time point t the squares of the next n samples producing a slot for each t. In practice, the t repeats at a distance of T.
A second mean $Q_i(t)$ allows a piecewise integration over some slots or frames to smoothen the acoustic film

$$Q_i(t) = \frac{1}{m} \sum_{k=1}^{m} P_{ik}.$$ (17)

Using linear microphones and linear ADC's, the arithmetic mean in acoustics appears with the unit for sound pressure in Pascal (mostly in mPa). The sound level L is the logarithmic measure of the sound pressure related to a reference value, the threshold of human hearing $p_0 = 20\mu Pa$

$$L = 20\log(\frac{P_i(t)}{p_0}) \; dB.$$ (18)

For a continuous function the interference integral would appear with sample frequency f_s and sample time $T = n/f_s$ in the form

$$P_i(t) = \sqrt{\frac{1}{T} \int_0^T p_i^2(\tau) \, d\tau}.$$ (19)

[1] http://en.wikipedia.org/wiki/Mean

Sample p(t)   Interval mean P(t)

n=5

m=3

Film slot mean Q(t)

t
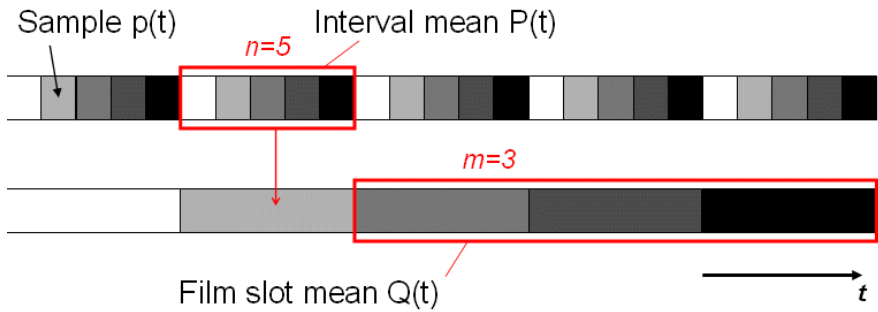
Fig. 9 Means used for Acoustic Camera.

5 Conclusion

If a time function is leaving one-dimensional space, it becomes a wave. Main characteristic of a wave is that all points of the peak have the same delay to the source location.

Different, interaction aspects between type of conjunction, time function and type of integration influence interference integrals.

Defining the node-edge construct, calculations of interference conjunctions can work time- parallel. In vector notation, the output is a simple element-by-element conjunction of input vectors. The result of integration process is a number, that defines in images the lightness or color of the node or pixel.

Interferential functions correspond to relativity of wave expansion, not to gate location. Non-locality of interferential function reasons circuits, which deliver different behavior compared to state machines.

To highlight locations of wave interference, the specific integral calculus has to be non-linear (multiplication, exponation).

To process bipolar signals, sign switching restricts multiplication to two channels. Higher channel numbers need addition. Usability of multiplication for more then two signals restricts the time-functions to be monopolar (nerve).

Like acoustic time functions, photonic time function waves are symmetric. The conjunction process has additional type. Hence, acoustic and photonic processes need asymmetric integration. The mapping (image projection) needs any form of a non-linearity (quadratic mean; intensity, energy, quadrature).

Open integration produces integral divergence. To use the divergence technical, a time trigger has to stop the process.

Using a comparator resetting the integral value, the run-time characterizes the integral value. Analyzing the loop process, we find nerve pulse generation as a special case to code the integral value at the output of nerve cells in time-domain (frequency).

Using open integration with time limit, the quadratic mean technique of Acoustic Cameras appears as a specific integration method.

We could observe, that the interference network approach allows the calculation of systems in different fields (digital gates, light and acoustics). Pulse frequency coding of nerve cells shows a special case of interference integration.

References

[1] Bullock, T.H.: Signals and signs in the nervous system: The dynamic anatomy of electrical activity. Proc. Natl. Acad. Sci. 94(1), 1–6 (1997),
     `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC33648/`

[2] Gerstner, W., Kistler, W.M.: Spiking Neuron Models. Single Neurons, Populations, Plasticity.Cambridge University Press (2002), ISBN 0 521 89079 9,
     `http://icwww.epfl.ch/~gerstner/SPNM/node26.html#section-IF-nonlinear`

[3] Hamilton, W.R.: Theory of Systems of Rays. In: Wilkins, D.R. (ed.) Transactions of the Royal Irish Academy, vol. 15(1828), pp. 69–174 (1828),
     `http://www.maths.tcd.ie/~dwilkins/`

[4] Heinz, G.: Zur Physik bildgebender Rekonstruktion akustischer Bilder und Filme im Zeitbereich. 33. Deutsche Jahrestagung für Akustik, Uni Stuttgart (2007),
     `http://www.gfai.de/~heinz/publications/papers/2007_DAGA_Reko.pdf`

[5] Heinz, G.: Relativität elektrischer Impulsausbreitung.- Schlüssel zur Informatik biologischer Systeme. 39. IWK TU Ilmenau 27.-30.9, S. 238-245 (1994),
     `http://www.gfai.de/~heinz/publications/papers/1994_IWK.pdf`

[6] Heinz, G.: Introduction to Wave Interference Networks. In: Proceedings of Workshop 2010 Autonomous Systems, Camp de Mar, Mallorca, Spain, October 24-29, Shaker-Verlag (2010),
     `http://www.gfai.de/~heinz/publications/papers/2010_autosys.pdf` ISBN 978-3-8322-9514-1, ISSN 1866-7791

[7] Heinz, G.: Interference Networks as a Generalizing Signal Theory. Plenary invited lecture. In: 7th Int. Conference on Computing and Information Technology IC2IT, May 11-12, King Mongkut's University of Technology, Bangkok (2011),
     `http://www.gfai.de/~heinz/publications/papers/2011_bangkok.pdf`

[8] Heinz, G.: Neuronale Interferenzen, p. 300 (1993) (manuskript),
     `http://www.gfai.de/~heinz/publications/NI/index.htm`

[9] Heinz, G.: Interference integral animations (1995-2009),
     `http://www.gfai.de/~heinz/publications/animations/index.htm`

[10] Heinz, G.: Interferential Projections in Inhomogeneous Space, web-page (1997) on,
     `http://www.gfai.de/~heinz/historic/distort/distort.htm`

[11] Lapicque, L.: Recherches quantitatives sur l'excitation électrique des nerfs traitée comme une polarisation. J. Physiol. Pathol. Gen. 9, 620–635 (1907)

[12] McCulloch, W., Pitts, W.: A Logical Calculus of the Ideas Immanent in Nervous Activity. Bulletin of Mathematical Biophysics 5, 115–133

[13] Packard, A.: Organization of cephalopod chromatophore systems: a neuromuscular image-generator. In: Abbott, N.J., Williamson, R., Maddock, L., Cephalopod Neurobiology, pp. 331-367. Oxford Press (1995),
`http://www.gfai.de/~heinz/historic/biomodel/squids/squids.htm`

[14] Scilab 3.1.1:
`http://www.scilab.org/communities/developer_zone/scilab_versions`

Self Organization in Enterprise 2.0 Communities: To Introduce a New Experts' Exchange Application

Gerald Eichler

Abstract. Web-based communities are a real example for self organisation of people. While everybody talks about Web 2.0 applications, the Enterprise 2.0 discussion runs very slowly due to several social and cultural aspects as well as the contradiction between communities without hierarchies versus well-structured companies. By analysing common things and differences of private and business communities, basic mechanisms are discovered. With SPREE – the knowledge exchange network – an Enterprise 2.0 application created by Deutsche Telekom Laboratories is investigated during the three phases of its pilot trial in the knowledge domain of recommendation. Strengths and weaknesses are analysed and recommendation, how to accompany the introduction of a thematic specific social network is given. By founding a spin-off both, the enterprise and the community track of SPREE will be continued.

1   The Community as an Autonomous System

Usually, an autonomous system (AS) is defined as "a collection of routers under a single administrative authority, using a common Interior Gateway Protocol for routing packets".[1] But the term AS is also useful to define in analogy a community for this paper as "a loose group of people, under a common code of conduct, using Internet based communication tools".

The occurrence of online web communities has neither been planned from the beginning nor really controlled during its existence. However, the largest existing

---

Gerald Eichler
Deutsche Telekom AG, Laboratories, Information Relevance
Deutsche-Telekom-Allee 7, D-64295 Darmstadt/Germany
e-mail: gerald.eichler@telekom.de

[1] http://dictionary.reference.com/browse/autonomous+system

communities e.g., *Facebook, YouTube* and *twitter* are driven by hidden business aspects nowadays. Most business models are based on **reactivity**, as proactivity would require a clear future vision which is not given due to the basic self-organizing character of worldwide communities.

1.1 The Community Application Hype Cycle

As for other products, hype cycles are typically for communities seen as a system. Some well-known communities e.g., *SecondLife* or *MySpace* almost died out.

Most successfully in terms of survival time are communities which follow the principles of common public licences like *Wikipedia* or *Fotocommunity*. Equality of members and voluntariness of membership is sometimes covered by social pressure. More and more community members become aware of the fact that ownership and rights of contents is a cruel issue.

The principle of equality contains the important aspect to unify two roles within one person: to be consumer and producer of contents, the community is working on. Don Tapscott introduces therefore the new term *prosumer* [1]. A certain ratio of production to consumption, which ranges from 1:5 to 1:50 is required to guarantee the survival of a community as system. Booth, too many producers (overwhelming diversity) and too many consumers (no new contents) clearly fail over time.

1.2 Community Specific Paradigms

The life time of online communities, respectively the life cycles of representative community applications is influenced by three paradigms. It started with the browser era and internet cafes in the 1990s, where the local installation of applications on personal computers became obsolete.

1. **Browser paradigm:** Installation-free use of browser clients increases the flexibility of communities due to device independent access.
2. **Mobility paradigm:** The value of a community increases by its mobility due to the availability of powerful mobile devices (smart phones) for everybody.
3. **Identity paradigm:** Anonymity and secure identities are two opposite but driving factors for the increasing value of a community.

Ray Kurzweil analysis many development trends over the time. For the life cycle of a paradigm three phases are characteristic: firstly, slow growth as early phase of an exponential growth; secondly, rapid grow, as late phase of exponential growth and thirdly, a levelling off. This results in a characteristic **S-curve** [2]. An ongoing exponential sequence is made up of a cascade of S-curves.

All above postulated paradigms characterize a certain time period of community. While smart phones twitter made just possible in the early 2000s, the 2010s will be characterized by new business relationships due to offering of secure restricted identification (RI) like the new German identity card, elektronischer Personalausweis (ePA) does [3].

2 Private and Business Communities

Business communities can be defined by two different characteristics. Either the acting community members follow strong business aims, like known from customer relationship management (CRM) systems, or the content objects are business goods with a defined value. Such goods are not limited to material objects but can be ideal objects like intellectual property rights (IPR) too.

To firstly distinguish communities by its primary aim, there should be a division between private communities => **Web 2.0** philosophy and business communities => either **Enterprise 2.0** or pure business philosophy. Looking into the character of the goods which are exchanged between the members of the community it becomes clear, they might be very similar regarding the applied exploitation methods as depicted in Fig. 1.
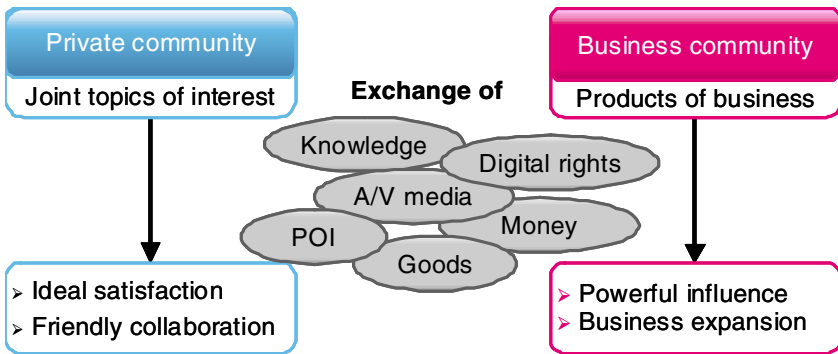
Fig. 1 Comparison of private and business communities

The appropriate selection of **communication channels** is the central core of a community. Often there is no need to create new communication models. With sms, instant messaging, email, blog or forum there are already many asynchronous options, the users are already familiar with. **Chat** which is from its origin synchronous can be extended to an asynchronous tool. This makes it very interesting for communities, where members are not always present.

2.1 Common Trends in and Driving Factors for Communities

Environmental factors influence the character and further development of online communities. Starting from the social behaviour of the *Net Generation* or *Generation Games* [1] there are new expectations of the users on the one hand and new technical options on the other. Table 1 gives an overview of those trends.

Table 1 Influence factors for online communities

Usability aspects	Technology aspects	Personalization aspects	User expectations
Always on	Online webspace	Context awareness	Free of charge
Multi-modality	Browser features	Personal profiles	Ready to use
Intuitiveness	Smart phones and	Social ranking	Self-explaining
Ease of use	netbooks	Anonymity on demand	Single sign on

Complex and monolith applications are widely ignored. Instead of purchasing a new software bundle, plus a multi-step setup procedure including installation, configuration and licence agreement, pure platform independent browser applications, paired with HTML 5 offers or smooth apps for the most common smart phone architectures including micro-payment are expected and accepted. Apple's iPhone, Google's Android and Microsoft's WindowsPhone are the answers of the new global players to win customers' vendor loyalty back. Beside this, application developers become a new type of community on its own, as users know best what they really want or need next.

2.2 Social Enterprise Communities

While private communities define mostly their own rules in a very practical and open manner, enterprise community often have to fit a given framework of rules. There are much more technical, security and management issues than expected. The last sometimes reach a level of **political issues**. There is a contradiction in philosophy: while community often think hierarchy less most companies think in clear hierarchies [4].

In the following chapter the side effects are evaluated by following step by step a real trial.

3 The Introduction of a New Application in Business Context

When introducing a new **social** oriented application in an enterprise context many parties have to agree. In this case social stands for the relevance of personal data. To get the final management agreement, security experts, data protection entities, operation services and finally the workers' council have to give their written commitments. This process takes typically several months, when starting from the scratch. Fig. 2 summarises invocation factors.

But always keep in mind: finally, only the staff judges over victory or defeat of a new application where the participation works on a volunteer base.
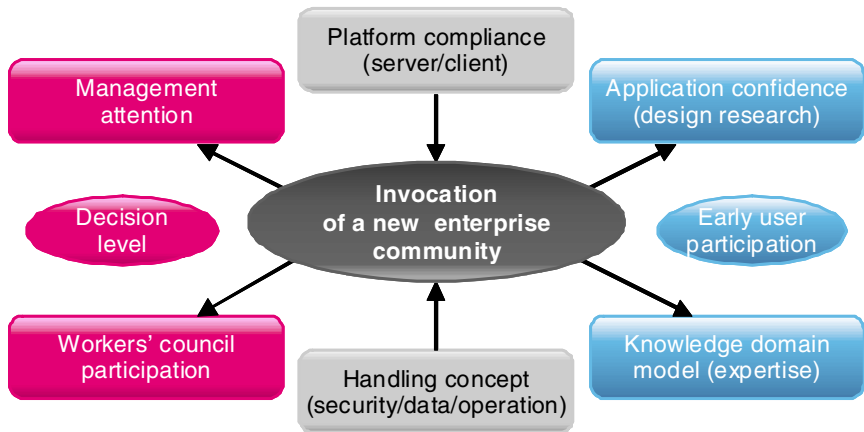
Fig. 2 Factors of invocation for a new enterprise community application

3.1 *SPREE – The Knowledge Exchange Network*

Although SPREE is not an acronym, the short name of an Enterprise 2.0 community research project within Deutsche Telekom Laboratories was given a meaning for the second pilot trial phase at Product and Innovation (P&I) [5] by *Social Pilot for Recommendation Experts' Exchange*. Recommendation is an important topic that touches many different products and therefore departments within several Deutsche Telekom business units. SPREE was developed to allow efficient interaction between experts without borders and hierarchies.

The basic *SPREE principle* is rather simple

> By asking a new **question** in natural language, a **classification** mechanism is deployed. Multiple **experts** are addressed and notified based on their profiles to get into contact with the questioner by means of asynchronous **chat**.

3.2 *Knowledge Domain Modelling by Taxonomies*

Within SPREE profiles are defined as a qualified instance of the underlying taxonomy, describing the recommendation domain. Neither the number of branches (dimension) nor the number of levels (deepness) is limited. However, the number of nodes, called **concepts**, should be in an adequate relation to the size of the community. As the community grows the taxonomy will be more detailed or extended.
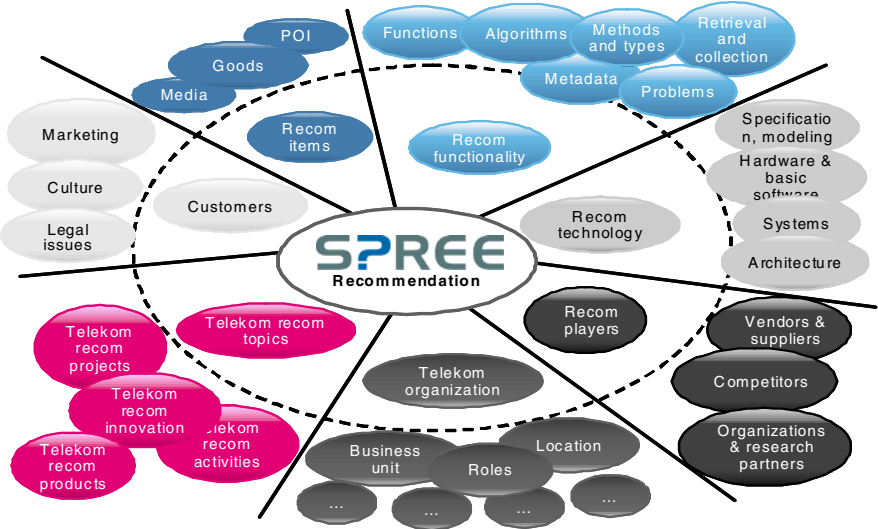
Fig. 3 Taxonomy of the SPREE recommendation domain at Deutsche Telekom P&I

Colours in Fig. 3 show the different orthogonal dimensions of the recommendation domain. Orthogonal stands for the independence of the concepts of the different dimensions. While classical taxonomies lead to a unique classification, the SPREE case supports multi-classification enriched by a flexible relevance weight per concept.

Ontologies would be the next logical step. But due to their complexity in edge attributes, the performance of matching is reduced. However, current research works on such issues.

4 The 3x3 Trial Model

As private communities can grow smoothly from the very beginning, business communities usually require an excellent planning. Therefore, several separate pilot **trial periods** are recommended before starting a rollout. Each period was introduced by a scenario definition and careful requirement analysis. Table 2 shows some characteristics of version 1, 2 and 3.

Table 2 SPREE release candidates

Version (year)	V1 (2007/08)	V2 (2009/10)	V3 (2010/11)
Focus	Rapid prototype	Enterprise version	Community version
Target network	Internet	Intranet	Internet
Software status	Open source	Licence	Licence
Design target	None	CI/CD compliance	Logical simplicity

Table 3 documents accompanying tasks which were executed during the second SPREE pilot trial with version 2 in Winter/Spring 2010.

Table 3 Three phases of the SPREE pilot trial

Trial preparation	Trial execution	Trial evaluation
Paper model prototyping	Training offer	Statistics
Design research analysis	Moderation	Questionnaire
Thematic modelling workshop	Regular newsletter	Interview
Application feature selection	Portal integration	Management decision proposal

The following subchapters will stress important findings during the three phases.

4.1 SPREE Trial Preparation

To start a SPREE trial with a ready to use application, two important decisions are required: the topic (knowledge domain) has to be fixed and an appropriate target group needs to be identified. While the management has to be convinced of savings of time and money, colleagues can be inspired by understanding their topics and daily problems.

To win reliance in a new application is not a technical issue but a psychological process. The following offers can support this pretension:

- Trial participation on volunteer base
- Feature offers with freedom of choice
- Transparency of business logic process steps
- Full control over personal data at any time
- Pseudonym as visualized user name if wished
- Clear and minimized role concept
- Intuitive icons for status indication
- Compliance with company corporate design and IT workplace settings.

The trial participants were given the full set of features (Fig. 4), knowing in advance that only a limited set of functions will be used actively. To emphasize the community feeling, rating and highscore were activated after several discussions.

4.2 SPREE Trial Execution

To start a trial is the most critical phase, as a community needs a critical mass to succeed. Announcements were published on the Intranet, a first newsletter was individually distributed by email, a moderated opening session was initiated and the trial was integrated in the corporate portal as separate entry.
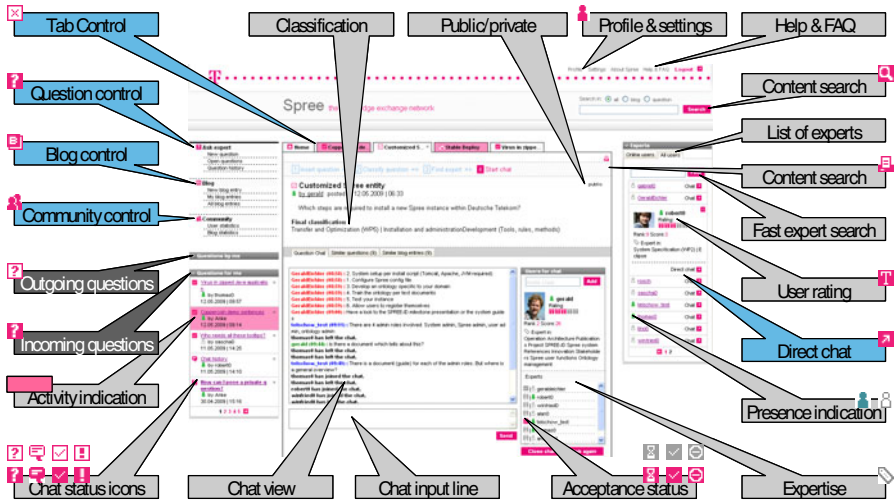
Fig. 4 Set of SPREE functions presented as browser application in Telekom look and feel

For moderation and guidance three dedicated SPREE users having a real face behind and representing different business units, were introduced to the community:

1. *Mr.Spree*: the project owner by Deutsche Telekom Laboratories to be the first address for any new ideas and lead trough the recommendation domain
2. *Spree.Pilot*: the guide by Deutsche Telekom P& I to monitor the trial goals and be a trusted person
3. *Spree.Help*: the contact person by T-Systems GmbH for any usability question.

Private chats using SPREE itself were the mostly applied means of support. From the users' point of view SPREE offers only two roles: the questioner and the expert. Each registered user is a questioner per default. In addition a user becomes an expert by simply filling its profile, which is a collection of expertise, selected from the modelled taxonomy.

So the principle of equality of community members is retained. The principles of freedom of choice and voluntariness were also offered in operation details:

- Users were allowed to **decline** chat invitations when they were addressed by SPREE as expert.
- Users who were not addressed by SPREE as an expert were allowed to **join** "open questions" by simply participating in the chat.\

"Closed questions" could (but not must) be rated by the questioner. Each involved expert got the same rating on a five-T-scale. So the given expertise of the small chat community was rated, not individuals.

   Out of interesting chat dialogues instantly blog entries (edited micro articles) could be created by both, the questioner and the expert. This is an important long term added value as SPREE not only addresses experts but also finds "similar questions" end "related micro articles" to any new incoming question.

4.3   SPREE Trial Evaluation

The **usage statistics** over the trial period are a measure of the community development and activity. Per timeframe (day, week or month) the following parameters were recorded (Fig. 5):

- New registrations
- Logins
- New and closed questions
- New blog entries (micro articles)
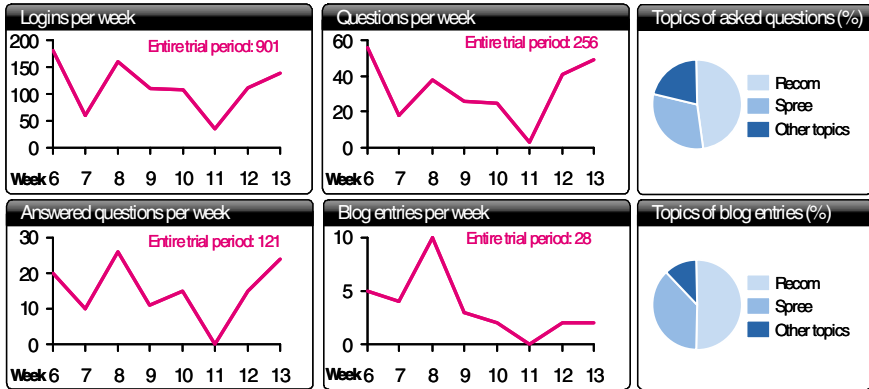
Fig. 5 SPREE trial activities over the time of the trial period (February till April 2010)

   When not driving a trial by active moderation during public holidays like week 11 in Fig. 5, the **activity** goes down. By seeding new interesting topics and announcing the official end of the trial period, activities increased again (week 12/13).

   The analysis of discussed **contents** over the trial showed that about 50 % of the discussions dealt with the modelled knowledge domain of recommendation.

   The history of **highscores** was not tracked but found a very important motivation for the trial participants. By browsing the highscores, only the ten highest records plus the own personal score were displayed. The long tail was hidden.

By asking the trial participants to fill a **questionnaire**, feedback on the following criteria was collected (numbers in brackets indicate the number of questions):

- Overall: general idea of SPREE as community tool (5)
- Matching: quality, expertise & matching (7)
- Interaction: usage & communication (8)
- Satisfaction: usability & experiences (7)
- Diversity: usage & functions (14)
- Expectations: future functions (7).

The social result: daily collaboration by means of Enterprise 2.0 support is still on an early cultural stage where strong project thinking is manifested.

5 The Future of SPREE

The new *SPREE Gesellschaft für Informations- und Wissensmanagement mbH*[2], founded as a spin-off as result of the SPREE project series in April 2011, will follow three tracks: offer SPREE as both, a community and an enterprise product with different skin and feature sets (examples in Fig. 6) {product}, the integration of the intelligent SPREE matching core into existing portals {service}, and running SPREE servers {operation}.
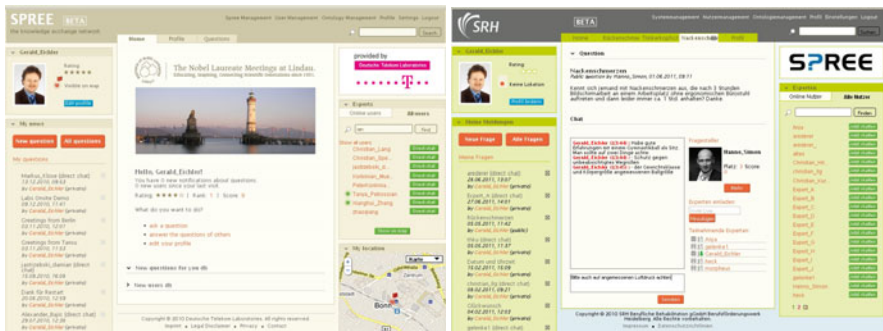
Fig. 6 Two SPREE community skins. (a) The Lindau Nobel Laureates Meeting SPREE community provided by Deutsche Telekom Laboratories, URL: `http://www.spree-lindau.de/` (b) SPREE pilot for medical pain management by SRH.

5.1 Summary on Community Application Success Factors

From the social point of view SPREE establishes easily **shortcuts** in a given community. **Chat** is an appropriate communication way to work as contact builder

---

[2] `http://www.spree-gmbh.de/`

between community members on base of a friend-of-a-friend recommendation. A **social graph** as introduced in the SPREE Prototype version [6] by the DAI-Labor of TU Berlin visualizes new relationships.

To support both, **public and private chats** is essential even in a business community. New private contacts strengthen the business success. Other SPREE trials have shown that a slight part of private topics in the model of the knowledge domain, like hobbies drastically increase the overall usage of the system and reduce contact fears.

Business communities do not work as good as private communities in the sense of autonomous systems. **Simplicity in concept** and **intuitivism in use** are key success factors to win members' confidence. Such an example is the four step SPREE core process:

New question => classification => expert selection => asynchronous chat

Although step two and three are optional and can be skipped, users prefer the feeling of having control of everything. Additional available SPREE **functions** (see Fig. 6a) e.g., twitter channel integration and user **localisation** [7] were not tested within the internal trial, but make such an application more attractive for the community.

There is no installation at all is required to run SPREE as a client. But already the registration is an inhibition threshold. For automatic authentication the user's email is recommended as it is a unique identifier if not already a **single sign on** (SSO) can be supported as offered for the trial. User name as shown for the community and login name are independently. Pseudonyms are supported.

Self-organisation needs silent management commitment and smooth **trigger events** like newsletters, workshops, competitions or bar camps.

5.2 Further SPREE Developments

The following items are a selection of identified tasks for further development to support future SPREE-based communities.

Usage & appearance:

- Increased participation of the community to develop the knowledge domain
- Configurable user experience levels to handle the web application
- Development of a mobile app.

Functions & features:
- Support of multi-lingual matching core
- Enrichment by further context awareness and groupware components
- Reinvention of visualisation by social graphs
- Topic graph: knowledge domain maintenance
- User graph: community relationship discovery

Core technology & environment:

- Flexible interfaces to export and import user profiles
- Standardized storage of taxonomy
- Classificatory library for optimized matching
- Introduction of feedback control loops
- Expertise level adaptation
- Rating-based expert recommendation
  - Decline and join rate re-profiling

Platform:

- Java transformation from JSF towards OpenJDK
- Increased browser independency.

References

[1] Tapscott, D., Williams, A.D.: Wikinomics. How Mass Collaboration Changes Every-thing, Portfolio, New York (2006)
[2] Kurzweil, R.: The singularity is near: when humans transcend biology. Penguin Books, New York (2005)
[3] BSI TR-03110, Advanced Security Mechanisms for Machine Readable Travel Documents – EAC, PACE, and RI. Bundesamt für Sicherheit in der Informationstechnik, Bonn (2010)
[4] Eichler, G., Lommatzsch, A., Strecker, T., Ploch, D., Strecker, C., Wetzker, R.: From Community towards Enterprise – a taxonomy-based search for experts. In: Erfurth, C., Eichler, G., Schau, V. (eds.) 9th International Conference on Innovative Internet Community Systems (I²CS 2009) GI LNI-148, Köllen Druck, Bonn (2009)
[5] Eichler, G., Kantner, V.: Spree Pilot Trial @Products & Innovation: Social Pilot for Recommendation Experts' Exchange. Deutsche Telekom Laboratories, Darmstadt (2010)
[6] DAI-Labor TU Berlin. Spree Project Portal: spree – the knowledge exchange network (2008), http://project.askspree.de/; (accessed July 8, 2011)
[7] Rederer, A., Eichler, G., Kury, T., Schwaiger, R.: Spree – Trial-based Improvements: Transferring an Enterprise 2.0 into a Web 2.0 Expert Community Application. In: Eichler, G., Küpper, A., Schau, V., Fouchal, H., Unger, H. (eds.) 11th International Conference on Innovative Internet Community Systems (I²CS 2011) GI LNI-186, Köllen Druck, Bonn (2011)

A Generic Sublayer for Structured Peer-to-Peer-Networks

Daniel Berg

Abstract. Structured peer-to-peer overlay networks offer great advantages for commercial, industrial, and scientific networked applications. However, those networks have to deal with the problem of churn; Peers can join and leave the network at any time and they often leave the network without any prior notification and therefore with no chance for the network to preserve the structure in advance. This article presents the concept for a generic sublayer for arbitrary structured peer-to-peer networks to overcome the churn problem.

1 Motivation

Structured peer-to-peer overlay networks establish a certain topology on top of the transport network. Resources are placed at specific, identifiable locations in the overlay network, and resource discovery can be performed in a systematic, and often even resource-related directed manner [9], guaranteeing that resources can surely be found when they are available anywhere in the network. However, keeping these overlays interconnected is more complicated especially in case of ungraceful node failures.

In general it cannot be expected that a peer always logs off gracefully according to the rules of the peer-to-peer protocol. Node failures also can occur because of network failures or computer crashes. In these cases there is no explicit notification that lets the neighbor-nodes know that a node-failure occurred. In case of multiple peers failing at the same time, it is even possible that the structure is entirely destroyed. Though most overlays, such as Chord, provide mechanisms to try to react on such situations, they lead to additional traffic overhead, longer routing times, or to a temporary or permanent unavailability of resources that actually should still be accessible.

Daniel Berg
Faculty of Mathematics and Computer Science, Fernuniversität Hagen
e-mail: `daniel.berg@fernuni-hagen.de`

On the other hand, in a network with thousands of peers it is almost sure that at least a certain fraction of the peers is online at any time. It should be possible to utilize these peers to preserve the overlay's topology and thus to avoid or reduce the need for complex repair-mechanisms.

There are a lot of well-investigated topologies and new algorithms for structured peer-to-peer networks, and there are still coming more. Depending on the application they are designed for, they have their advantages and disadvantages. The place concept introduced in this paper does not aim to introduce yet another peer-to-peer overlay architecture, but to increase the reliability and the availability of existing and new architectures.

2 Related Work

Many existing peer-to-peer protocols provide topology specific mechanisms to provide a certain degree of reliability and availability. These mechanisms may be parts of their routing algorithms and have to be fine-tuned to achieve a trade-off between availability and additional resource requirements. Tapestry [14] uses backup-neighbors, if neighbors are unavailable. Chord [12], which establishes a ring topology, uses finger-tables to improve lookup time and to provide alternative routes. Kelips [4] works with node-groups sharing the same id to build a DHT, and thus providing alternative routes through the peers of a certain group. Other approaches like Hybrid Chord modify or extend existing specific protocols to achieve more reliability [3, 2]. Yet another group of approaches use existing topologies and apply additional algorithms on top of them to increase resource availability e.g. by replicating resources, or by preferring peers with high availability to store resources [5, 11]. In contrast to these approaches, this article introduces a sublayer to provide a generic base for existing protocols. Beside this, the presented concept offers new possibilities for new protocols built on top of this sublayer.

3 Architecture

We introduce the concept of a generic peer-to-peer sublayer: the place management layer. It aims to avoid failures in any peer-to-peer network built on top of it from the outset.

Figure 1 shows the place management layer residing on top of the network layer. The overlay's algorithm is implemented on top of the place management layer at the peer-to-peer topology layer; the actual application is implemented at the application layer on top of the topology layer.

Based on the place management, layer nodes building the overlay's structure are not represented by single peers that provide their resources at the locations they reside on in the network, but by places. Within the sublayer a place in fact is maintained by a small number of n arbitrary peers that form a place cluster. Any peer of this cluster is able to overtake routing-, management- and maintenance tasks related to the corresponding place position. This happens transparently to the algorithms

Application Layer The P2P-Application (File-Sharing, Resource-Sharing, Voice-Chat, Video-On-Demand, etc)
P2P-Topology-Layer The layer that builds and maintains the network topology; CAN, Chord, Tapestry, Koorde, etc.
Place-Management-Layer Place-Abstraction; multiple peers form dynamically a topology-element for the structured network, a place. It appears as a single peer to the above layer.
Transport Layer Transport Network; TCP/IP

Fig. 1 The Place-Management-Layer is located between the Transport Layer and the P2P-Topology-Layer that defines the actual P2P-topology.

built on top of the place management layer; they operate on places, on virtual peers, as if they were real ones. All details about the place management are hidden in the place management layer. Even if $n-1$ of the n peers fail, the place is still available and no repair mechanisms of the upper layer's algorithms have to be initiated. Thus, reliability and availability for any peer-to-peer network working on top of this layer can be improved.

The place layer concept allows to separate overlay maintenance tasks from tasks like providing and consuming services. A peer can offer its resources at a certain place and contribute to the overlays's stability at a completely different place, and - if the peer has sufficient network resources - even at multiple places at the same time. The place management can change and adapt these associations at any time to improve availability dynamically. Again, the application on top of the place management layer will not even notice it.

3.1 Places and Place Records

A place is defined by its place record, a dataset P containing all information that characterize the place's role within the overlay. It is a 5-tuple $P = (D, A, l, R, N)$ containing the following data:

- D: A unique place id that identifies the place's position within the structure. On top of the place management layer, place ids are used to address neighbor places.
- A: The place's place address, which is a set containing the addresses of all peers that currently contribute to the place cluster. The elements of A are called node addresses. They are tuples (a, w), where a is a unique peer id and w is the network address of the peer associated with it.

- $l \in A$: A leader peer's node address. A dedicated peer that is needed for synchronization among the cluster's peers.
- Resource set R, containing tuples (R, w), where R is the id of a resource virtually associated with this place, and w the node address of a computer that actually provides the resource.
- A set N that contains the place addresses of this place's neighbor places.

P is shared among the peers building the place cluster. The cluster leader l is responsible for synchronizing the instances of P if changes occur.
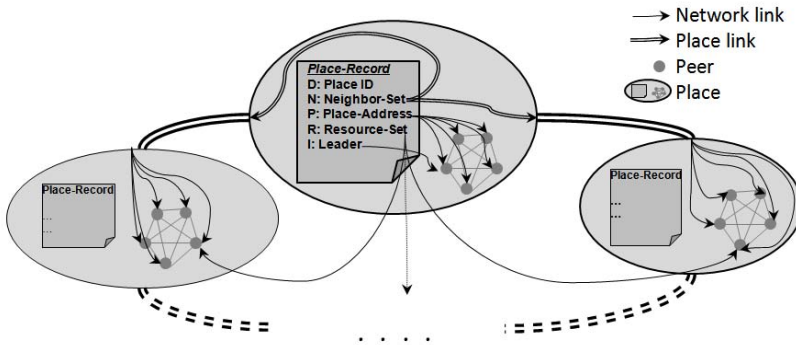
Fig. 2 The place-concept, simplified

3.2 Place Clusters

A place cluster consists of a small number of n peers. They keep their place record P online and in sync. Furthermore each of the peers is able to perform routing tasks, such as forwarding of resource discovery requests.

A place cluster is not static. The place management is able to remove peers and to add new peers dynamically at any time. For example, when the number of online peers in a place cluster falls under a certain threshold, it could start to aggregate other peers to ensure the place's availability. High bandwidth peers can be associated with clusters of multiple places regardless of their role as resource-provider or -consumer. Figure 3 illustrates how peers and places forming a ring are related to each other.

If peers provide additional information about their online behavior, the place management can use this information to make reasonable decisions on when to remove or aggregate peers. Algorithms can be applied that distribute the peers among the places in that way that a (sub-) optimal overall-availability is achieved.

3.3 Cluster Leader

It was already mentioned that routing tasks related to a place can be performed by any peer of its place cluster. Operations that change the place record's content, as
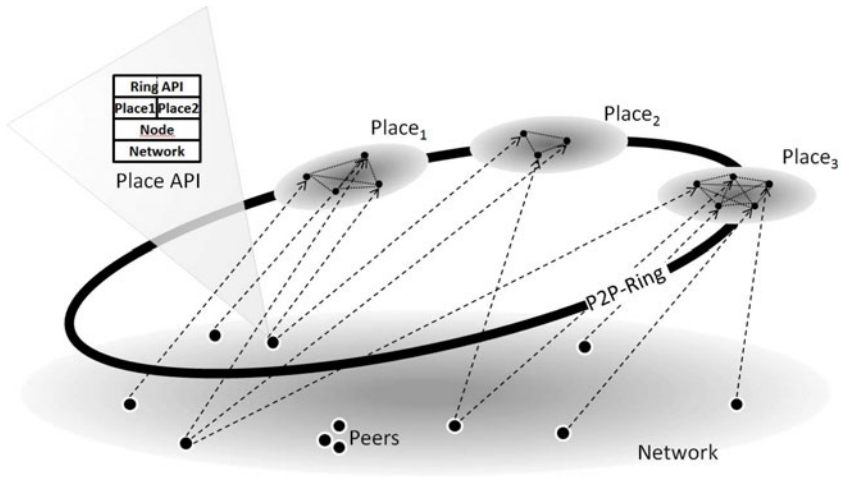
Fig. 3 Peers and places forming a ring network

for example adding or removing of peers or changing neighborhood relationships, need to be synchronized among a place-cluster's peers.

These operations are done by a dedicated peer of the place cluster, the cluster leader. The leader is responsible for observing the cluster. It decides whether and when to add or remove peers and it also manages the neighbor link set N that the overlying layer establishes to build the network structure. This is necessary since it is possible that place addresses change in the place management layer without the knowledge of the overlying layer. If a cluster leader goes offline, the remaining peers of the place agree on a new cluster leader.

3.4 Place Addresses

When a place sends a message to another place, the sending place management layer chooses a node address from the target place's place address and sends it to the corresponding peer. If this peer is unavailable or does not belong to the target place's cluster anymore, the sending place management layer will choose another node address. When the message could be received by a peer of the correct target place, the place management layer will give this message to the overlying peer-to-peer topology layer, where it can be processed as an ordinary message coming from another place.

As peers can join or leave a place cluster at any time, a place's place address permanently changes and needs to be updated everywhere, where it is used to link to the place. These updates should not be performed every time the place address changes in order to keep network overhead low. Even if some nodes' addresses of a place address become invalid, the remaining addresses can still be used to communicate with the referred place. Place address updates can come piggybacked with the ordinary place-to-place communication.

All these things happen within the place management layer. The application implemented on top of it will not get in contact with place addresses at all just with their unique place ids, which never change, regardless to what node addresses they are mapped. Since the place id is unique and independent of individual nodes and their network addresses, the peer-to-peer application has no need to deal with leaving peers, as long as the place management layer finds peers in time that keep the place alive.

This way of place communication' distributes the communication between places to multiple links between the places' peers and can reduce individual peers' communication load. It also makes it harder to observe the communication between places, which could be useful in applications that require anonymous communication.

4   A First, Simple Place Availability Model

A place is called available, if at least one peer of the place cluster is online. To give a first answer to the question, how many peers are necessary for a cluster to guarantee a certain minimum cluster availability, we apply a simple churn model for a cluster's peers, based on $MTTF$- and $MTTR$-parameters. (Mean Time To Failure, and Mean Time To Repair resp.). $MTTF$ specifies the mean time a peer is online. $MTTR$ specifies the mean time a peer is offline. A peer's online behavior is modeled by a state machine switching with a probability of $\lambda = 1/MTTF$ from its online- to offline-state and with a probability of $\mu = 1/MTTR$ from offline to online. In this context we call λ a peer's offline-rate and μ a peer's online-rate.

The availability B_i of a peer p_i is the probability with which it is online:

$$B_i = \frac{MTTF_i}{MTTF_i + MTTR_i} = \frac{\mu_i}{\mu_i + \lambda_i} \tag{1}$$

Figure 4 visualizes the influence of μ and λ on the availability B. The higher the offline-rate λ is, the worse the availability and the bigger the influence of the online-rate μ. For $\lambda = \mu$ B is 0.5.

A cluster's availability that contains the peers p_i with the availability B_i is calculated with:

$$B_{cluster} = 1 - \prod_{i=1}^{n} (1 - B_i) \tag{2}$$
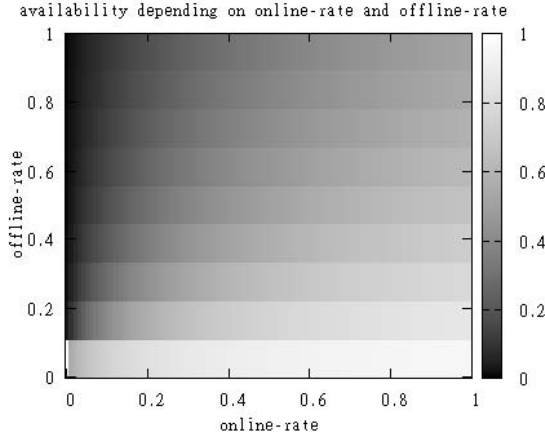
Fig. 4 The function $B(\lambda,\mu) = \mu/(\lambda+\mu)$. Bright areas show availability near 100%, dark areas show availability near 0%.

For 10 peers, all having an availability of $B_i = 0.5$ the cluster's availability would result in $B_{cluster} = 1 - 0.5^{10} = 0,9990 = 99,9\%$. But a cluster's online behavior can strongly differ depending on what online- and offline-rates lead to the availability of 50%. For big rates there occur a lot of offline and online-changes. In contrast, for small rates, we have very few changes in the cluster. The bigger the offline-rates of the participating peers, the faster the place management has to react on possible peer-failures that finally might lead to a cluster failure.

To illustrate this effect, consider two groups of peers, A and B. Peers in group A have mean MTTF- and MTTR-values of 7 hours. For peers in group B is $MTTF = MTTR = 7days$. The availability of all peers in both groups is 50%. With different cluster sizes from 2 to 20 and peers coming with different fractions from group A and group B the number of offline-changes (i.e. the number of situations all peers of the cluster went offline) were counted. Figure 5 shows the result. The x-axis shows the cluster size, the y-axis shows the fraction of peers coming from group B. For example, 0.2 means that 20% of the peers in the cluster come from group B and 80% from group A. The gray squares show the number of offline-changes that occurred with the corresponding combination and cluster size. The brighter a square is, the more offline changes occurred. Black shadings indicate that the cluster never went offline during the whole simulation taking about 1,05 million time steps corresponding to about 24 months.

The figure indicates that the different $MTTR$- and $MTTF$-values strongly influence the cluster's offline behavior even though they all result into the same availability of 50% (Note, that the gray-scales indicating the offline-changes are logarithmic). If a peer provides information such as $MTTF$ and $MTTR$, the place manager can use them as a criterion for effectively collecting peers for its place cluster. If a cluster already contains a certain amount of high-availability peers, it might
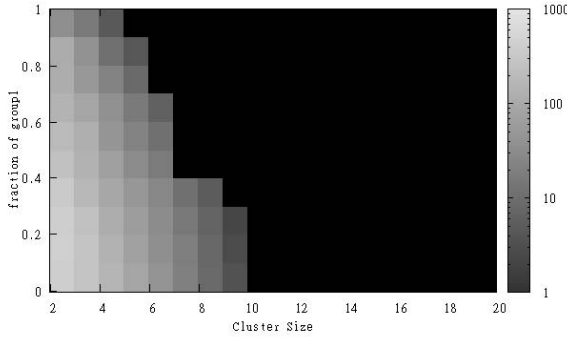
Fig. 5 Number of offline-changes for different cluster sizes and combinations of peers with different off- and online-rates.

be enough just to aggregate some 'office-peers' with low availability and exchange them from time to time to ensure the cluster's availability. This avoids that place managers just look for high-availability peers, while nobody would benefit from the contributions smaller' peers could make to distribute the network load.

Note that the presented model is a first simple approach that turns out to be imprecise when comparing with real-life situations related to churn. But it already shows which criteria should be taken into account when choosing peers to build a place cluster. Central part of the future work will be to investigate and apply other, more precise churn models like those discussed for example in [1], [13], [6], or [8].

5 Advanced Operations on Place-to-Place Networks

The transparent separation of a peer's role as a resource provider/consumer from its role as overlay manager provides some new ways on organizing place-to-place networks.

5.1 New Ways of Letting a Network Grow

The place management layer offers new possibilities in ways of letting a peer-to-peer network grow. Places can generate new places and destroy them largely independently from existing, joining and leaving peers. Multiple places can perform these operations synchronously. For instance, a grid for a torus could be build. To let it grow, all places residing on a ring R of that torus (bold circle in figure 6) would perform a synchronized split operation, resulting in a new ring of places inserted next to R. The peers maintaining the new places could be the same that belong to the places of R. But they could also be completely different peers, selected in a preprocessing step of the split operation. There is no necessary direct relationship between the number of peers and the size of the network (i.e. number of places). A
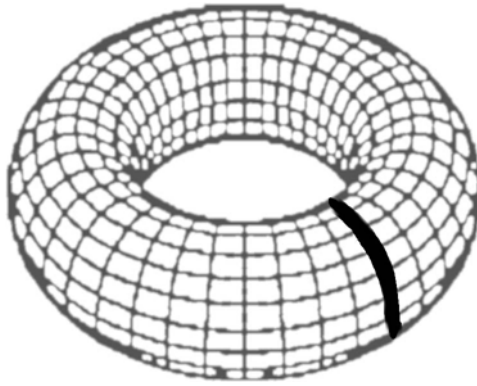
Fig. 6 A p2p torus that can be expanded by synchronously inserting places at the bold ring

'classical' torus grid overlay without places would require a certain number of peers to join at the same time to extend the torus, or it would have to accept intermediate inconsistent states for the torus.

5.2 Implicit Shortcuts

Since a peer can participate in multiple place clusters, it has knowledge about different locations of the peer-to-peer network. These peers could be utilized for shortcut links like proposed in [7]. These links are established dynamically to improve routing performance by considering various criteria, like user behavior and network resources. By participating at multiple place clusters a peer could act as a natural, implicit shortcut, and the rules for establishing those shortcuts could be used for a criterion on how to distribute peers to places.

6 Future Work

A central task of future work will be to look for more precise online- and churn-models for peers. They should take different groups of individual user behavior, bandwidth resources, and different time zones into account allowing the place management layer to make more precise predictions on when a cluster failure is likely to occur. This also leads to more precise criteria for peer aggregation; i.e. answering the question, how and when to aggregate what peers to prevent a cluster failure.

Furthermore it has to be investigated if the proposed architecture is really suitable for any peer-to-peer protocol. Protocols that maintain many neighbor links per peer may result into a lot of network overhead in order to keep the links up-to-date. It has to be observed when this overhead exceeds the overhead coming from the build-in repair mechanisms of the peer-to-peer protocol.

An important property of a peer-to-peer overlays is their scalability. A well-designed overlay scales well with its number of participants. It needs to be investigated how the place concept may influence the scalability of peer-to-peer overlays. We expect, that the place concept can improve existing protocols' scalability, since it provides the possibility to decouple the overlay's size from the number of participants at a certain degree.

7 Conclusion

Working on top of the proposed place management layer turns a peer-to-peer network into a place-to-place network. On top of the place management layer, the structure building entities now are places rather than peers. A place is an abstraction of a concrete peer which is transparently maintained by a cluster of arbitrary chosen peers. The peer composition that build such a place cluster can permanently change in order to ensure that the cluster remains online. This leads to very reliable topologies providing high availability for any topology built on top of the place management layer.

The place management layer makes it possible to distinguish between a peer's role as a resource consumer/provider and its role as an entity contributing to the overlay's structure. A peer can even perform the latter role at multiple locations in the network, transparently to the above lying application, leading to the possibility to design advanced peer-to-peer architectures beyond the classical ones.

Based on the concept introduced in this article further research has to be done to exactly quantize the benefits for existing peer-to-peer protocols in terms of network overhead, availability, and reliability.

References

1. Bhagwan, S., Voelker: Understanding Availability. In: Kaashoek, M.F., Stoica, I. (eds.) IPTPS 2003. LNCS, vol. 2735, pp. 256–267. Springer, Heidelberg (2003)
2. Cichoń, J., Jasiński, A., Kapelko, R., Zawada, M.: How to Improve the Reliability of Chord? In: Chung, S., Herrero, P. (eds.) OTM-WS 2008. LNCS, vol. 5333, pp. 904–913. Springer, Heidelberg (2008)
3. Flocchini, P., Nayak, A., Xie, M.: Hybrid-Chord: A Peer-to-Peer System Based on Chord. In: Ghosh, R.K., Mohanty, H. (eds.) ICDCIT 2004. LNCS, vol. 3347, pp. 194–203. Springer, Heidelberg (2004)
4. Gupta, B., Linga, D., van Renesse: Kelips: Building an Efficient and Stable P2P DHT Through Increased Memory and Background Overhead. In: Kaashoek, M.F., Stoica, I. (eds.) IPTPS 2003. LNCS, vol. 2735, pp. 160–169. Springer, Heidelberg (2003)
5. Hildrum, K.: Asymptotically Efficient Approaches to Fault-Tolerance in Peer-to-Peer Networks. In: Proceedings of the 17th International Symposium on Distributed Computing, pp. 321–336 (2003)
6. Leonard, Y., Rai, L.: On Lifetime-Based Node Failure and Stochastic Resilience of Decentralized Peer-to-Peer Networks. In: Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, vol. 33, pp. 26–37. ACM, New York (2005)

7. Lertsuwanakul, U., Mututanon, K.: Bridging Overloaded Nodes in an Overlay Network by Long Distance Links. In: Proceedings of 30th IASTED Conference on Modeling, Identification, and Control, Phuket, Thailand (2010)
8. Pandurangan, Raghavan, Upfal: Building Low-Diameter P2P Networks. In: Proceedings of the 42nd Annual IEEE Symposium on the Foundations of Computer Science, pp. 492–499 (2001)
9. Ratnasamy, F., Handley, K., Shenker: A Scalable Content Addressable Network. Journal ACM SIGCOMM Computer Communication Review 31, 161–172 (2001)
10. Rhea, G., Roscoe, K.: Handling churn in a DHT. In: Proceedings of the Annual Conference on USENIX Annual Technical Conference ATEC 2004, p. 10. USENIX Association Berkeley (2004)
11. Shi, S., Yang, G.-W., Yu, J., Wu, Y., Wang, D.: Improving availability of P2P storage systems. In: Zhou, X., Xu, M., Jähnichen, S., Cao, J. (eds.) APPT 2003. LNCS, vol. 2834, pp. 446–456. Springer, Heidelberg (2003)
12. Stoica, M., Karger, K., Balakrishnan: Chord - A Scalable Peer-to-peer Lookup Service for Internet Applications. In: Proceedings of SIGCOMM 2001, vol. 31, pp. 149–160. ACM, New York (2001)
13. Yao, L., Wang, L.: Modeling Heterogeneous User Churn and Local Resilence of Unstructured P2P Networks. Journal of IEEE Internaltional Conference on Network Protocols, 32–41 (2006)
14. Zhao, H., Stribling, R., Joseph, K.: Tapestry - A Resilient Global-Scale Overlay for Service Deployment. IEEE Journal on Selected Areas in Communications 22, 41–53 (2004)

Navigation in the P2Life Networked Virtual Marketplace Environment

Hauke Coltzau and Bastian Ulke

Abstract. As a current development of Peer-to-Peer applications, Networked Virtual Marketplace Environments (*NVME*) aim to provide large-scale virtual worlds, in which trading goods and services for real money is possible. Due to the distributed nature of the underlying system, special algorithms are needed to provide a consistent (over-)view of the whole virtual world. In this article, we therefore present an approach to build, maintain and query scalable dynamic maps as part of the P2Life NVME.

1 Networked Virtual *Marketplace* Environments

Seamless virtual 3-d environments, which are built on a distributed dynamic set of autonomous servers in a peer-to-peer manner, are known as *networked virtual environments* (NVE). Such systems usually are designed for a large number of participants, and thus can not only be used for massive-multiplayer-online-gaming (MMOG), but also to provide and enhance the functionality of today's web servers and web applications in an intuitive manner. Therefore, NVEs have the potential to transform the current World-Wide-Web into a World-Wide-Second Reality.

An important subset of NVEs are networked virtual *marketplace* environments (NVMEs), in which the interaction between user and virtual objects may be subject to a charge and result in a transfer of real money (in contrast to virtual currencies as e.g. in SecondLife [2]). While NVEs are often built on an implicit *agreement of trust* (see [1]), i.e. there usually is no way to verify the integrity of information given by any peer about the structure of the virtual world, the core design principle of NVMEs must be an *agreement of suspicion*. That is, in such systems, there always

Hauke Coltzau · Bastian Ulke
Fernuniversität in Hagen
e-mail: `{hauke.coltzau,bastian.ulke}@fernuni-hagen.de`

have to exist alternative and independent message routing paths, allowing the user to crosscheck information.

To gain acceptance both from providers and users, NVMEs additionally need to provide easily understandable and intuitive ways of navigation and orientation. Using maps of the virtual world for this purpose seems to be an appropriate mechanisms, since most people are accustomed to using maps in the real world. While building and maintaining maps in *centralized* systems can easily be done, fault tolerant, efficient and scalable *distributed* algorithms are still needed. They should be able to give an overview over the whole virtual world and let the user zoom and scroll through it. Additionally, to reduce both incentives as well as opportunities for single participants to deliberately give false structural information with e.g. causing damage to a competitor in mind, the data structures and algorithms must allow to easily identify malicious behavior of single peers.

In this article, a distributed map building and management concept is described. It is catered to the P2Life project [1], which aims to provide an infrastructure for scalable unlimited NVMEs. The structure of this article thus is as follows: After giving an overview about the P2Life project and related work in section 2, a distributed data structure is developed in section 3, containing all information necessary to build and explore the maps. Section 4 shows, how the data structure is distributed over the participating peers in a P2Life network. Finally, section 5 rounds up this article with a conclusion and an outlook on future work.

2   Related and Previous Work

Currently, to the best knowledge of the authors, no other project than P2Life exists that specifically focuses on networked virtual *marketplace* environments. Still, for the sake of completeness, an overview is given on concepts and approaches for networked virtual environments (NVEs).

2.1   *Networked Virtual Environments*

The most important aspect of developing an approach to provide networked virtual environments, is how to partition the virtual world and distribute the resulting segments over the dynamically changing number of participating peers. Since most existing approaches have in common that they aim to provide scalable virtual *gaming* environments, they consequently do not necessarily fulfill the requirements for virtual *marketplace* environments, as Table 1 shows.

As usual in gaming environments, the size of the virtual world is often limited (as in MinMud [3], Colyseus [5], MOPAR [4], Hyperverse [6], and Sollipsis [7]). In NVMEs, however, such an artificial shortage of available virtual space will limit the number of providers unnecessarily, which particularly influences the structure of applications and contents provided. Only scalable and therefore unlimited virtual

worlds are truly open to arbitrary and dynamically changing providers, contents, and users. However, although some approaches allow for such an unlimited size of the virtual world, none of them provides any map-alike mechanisms for orientation and navigation. In [9] and [10], the necessity of globally known maps is mentioned, but neither proposes ways to build and maintain them.

Table 1 properties of existing NVE projects

Project	Connectivity & Lookup	Authorative Owners	Unlimited Size
MinMud	DHT	no	no
ZFGS[a]	DHT & neighborhood	yes	not specified
Colyseus	DHT	no	no
MOPAR	DHT & neighborhood	no	no
pSense	neighborhood	yes	yes
KMA(2002)[b]	neighborhood	yes	yes
LOADER	neighborhood	yes	not specified
HyperVerse	DHT & neighborhood	no	no
Sollipsis	neighborhood	no	no
VAST / VON	neighborhood	yes	yes
Donnybrook	neighborhood	yes	not specified

[a] Zoned Federation of Game Servers [8]; [b] Work of Kawahara, Morikawa and Aoyama [10]

In NVMEs, where interaction between user and objects can imply an act of trade, mechanisms for authorative object ownership are needed, such that a provider always is the only authority controlling access to and state of his objects. Additionally, a user must always be able to identify the owner of an object, i.e. the contracting party for the trade. As shown in table 1, half of the existing approaches do not allow for any kind of authorative object ownership. Although no direct relation seems to exist between the two problems, exactly the same approaches that allow for an unlimited size of the virtual world also provide mechanisms for authorative ownership.

Whenever the lookup structure for finding objects in the virtual world relies on the structure of the virtual world (as in the Zoned Federation of Game Servers [8], pSense [9], the work of Kawahara, Morikawa and Aoyama [10], Sollipsis, VAST [11], and Donnybrook [12]), the potential for several neighborhood related attack scenarios as described in [1] arises. Even when distributed hashtables (DHTs) are used, they do not provide an independent lookup structure, since none of the concepts uses DHTs to deliberately detach neighborhood in the NVE from neighborhood in the underlying lookup structure.

2.2   The P2Life Project

2.2.1   Key Features and Concepts of P2Life

The P2Life NVME consists of square parcels of equal size. Each parcel is uniquely identified by an integer 2-dimensional coordinate. The size of the virtual world is not limited. Each parcel is either assigned to exactly one peer (the *provider*) or empty. The provider of a parcel, who takes over a similar role as today's WWW servers, manages all virtual objects and events on the parcel on its own account.

The NVME is kept in a consistent state by maintaining a separate lookup structure (see fig. 1) in form of an adapted *Content-Addressable-Network* (CAN) [13]. Each peer, who owns at least one parcel in the NVME, automatically takes part in maintaining the CAN. The region of the key space, each peer manages, does usually not contain the own parcel's key.

Users can explore the virtual world by means of a browser plugin, which acts as a client for the CAN. Interactions between users are currently not provided by P2Life, but will be part of future work. The interaction between users and the objects of the virtual world is managed by the provider of the parcel, the user currently is located on. Keeping in mind that P2Life is a *marketplace* environment, in which the interaction between users and objects may be subject to a charge (as in the current WWW), this approach is not only required, but also fair. Each provider can always control, which users are allowed to interact with the objects. Additionally, providers, who implicitly generate load by attracting users and possibly earn money this way, also should carry the resulting load themselves.

Fig. 1 P2Life NVME and lookup structure. The upper layer represents the unlimited virtual world and shows the assignment of parcels to providers. The lower layer displays the CAN managing the lookup. Each NVME parcel is assigned to a 2-dimensional key in the CAN. Neighbored parcels usually do not have neighbored keys.
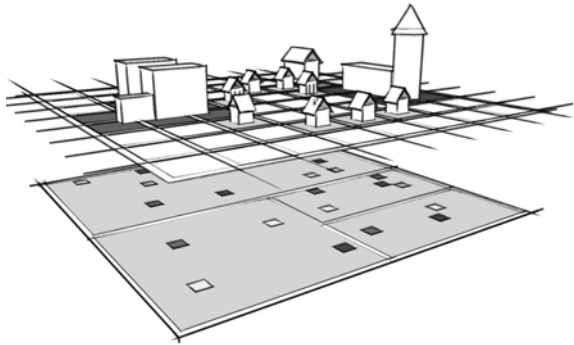
2.2.2   Lookup and Coordinate-to-Key Mapping

For looking up the provider managing any given parcel, the requested coordinate is mapped to a 2-dimensional key in the CAN by using a coordinate-to-key mapping with a dynamically adjustable number of digits. If the key generated during a lookup request does not have enough digits to uniquely identify the managing peer

in the CAN, further digits can subsequently be added without changing the already existing digits. The lookup peer managing the requested key in the CAN can now return the (IP-) address of the provider assigned to the coordinate.

Because the NVME uses unlimited coordinates while the CAN is based on a normalized coordinate space of $[0..1] \times [0..1]$, the coordinate-to-key mapping currently used in P2Life maps infinite integer coordinates to fractional numbers of arbitrary granularity. In a number system on base b, where $x = \sum_{(j=0)}^{n} a_j \cdot b^j$ with $a_j \in \mathbb{N} \wedge 0 \le a_j < b$, the value for k_x with d digits is generated by

$$k(x) = h_0(a_0) \cdot b^{-1} + h_1(a_1) \cdot b^{-2} + ... + h_{d-1}(a_{d-1}) \cdot b^{-d}$$

The key (k_x, k_y) for a given coordinate (x, y) is thus created by inverting the order of digits of each the x and y value and using them as fractional digits for k_x and k_y. In addition, each fractional digit a_i is mapped to another value by using a function $h_i(a_i) \to [0..b-1]$. The actual mappings described by h_i vary with different places i, such that the keys are distributed uniformly over the CAN. For a detailed description of h_i, please refer to [1].

2.2.3 Related and Unrelated Lookups

In P2Life, two different kinds of lookups are distinguished: A *related lookup* is used to find out the provider for one of the four parcels directly neighbored to the user's current position. In contrast, an *unrelated lookup* targets for any parcel in the NVME, which is not neighbored to the user's current position.
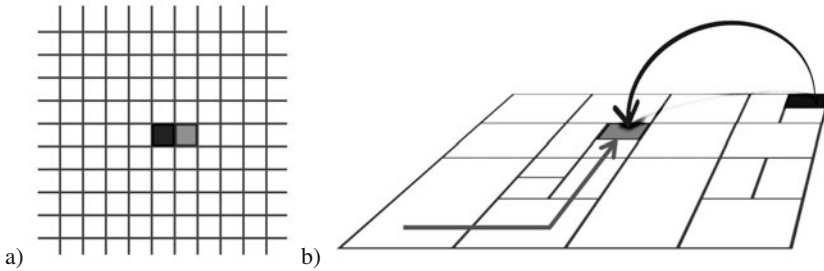
Fig. 2 a) Neighbored parcels in the NVME and b) disjoint paths for the according related (curved arrow) and unrelated lookup (angulated arrow) in the CAN

In [13], the average path length for routing a message in a 2-dimensional CAN is given with $1/2 \cdot \sqrt{N}$ (with N = number of providers), i.e. the complexity is in $O(\sqrt{N})$. Since neighbored NVME parcels usually do not have neighbored keys in the CAN (see fig. 2), both related and unrelated lookups in general are also processed in $O(\sqrt{N})$ time and message complexity. To reduce complexity for related lookups, a peer in the P2Life CAN does not only store the coordinate and (IP-) address of all providers registered for a given key, but also the addresses of the four

peers in the CAN that manage the key for the according *neighbored parcels*. This way, related lookups can be executed in a single hop independently from the size of the network and thus in $O(1)$ complexity.

Considering that each key represents an unlimited number of NVME coordinates, the coordinate-to-key mapping must be *collision preserving*. That means, if two values x_1 and x_2 map to the same key k_1, their successors $x_1 + 1$ and $x_2 + 1$ also map to the same key k_2. This way, each CAN peer only needs to store four addresses of CAN peers managing the keys of the neighboring coordinates. Hence, the memory complexity on each peer for maintaining the neighborhood references is in $O(1)$.

By using related lookups, browsing the P2Life NVME from any given coordinate is now possible in an efficient manner. Still, whenever a user wants to crosscheck the result of a related lookup, an unrelated lookup can be performed starting from any other position in the CAN. A malicious peer executing a *man-in-the-middle* attack will most likely not be in the routing path for the unrelated lookup, and thus, it's malicious behavior will be revealed.

Table 2 Symbols and terms used to describe map building and management approach

Symbol	Term	Description
(x, y)	parcel	smallest possible region in the P2Life NVME. Each parcel is either assigned to exactly one service provider, or empty. A parcel is uniquely identified by its coordinate (x, y).
b and e	base and level	b^{2e} defines the number of NVME-parcels visible when performing e consecutive zoom out operations starting from a single parcel.
$M_{(x,y),e}$	map of level e	the square fraction of the root map (see below) visible when parcel (x, y) is centered and e zoom-out operations have been performed. The coordinate of the lower left corner of $M_{(x,y),e}$ must always be of the form $(c_x \cdot b^e, c_y \cdot b^e)$ with $c_x, c_y \in \mathbb{Z}$, so that overlapping map fractions of the same level can never occur.
$B_{(x,y),e}$	top view	graphical representation (bitmap) of $M_{(x,y),e}$
$M_{(x,y),0}$	parcel map	map containing a single parcel located at (x, y)
$M_{(x,y),e_{max}}$	root map	smallest possible map containing all occupied NVME parcels
$C_{(x,y),e}$	set of carriers of level e	set of all peers in the CAN, which maintain a copy of $M_{(x,y),e}$.

3   Distributed Scalable Maps

The approach for building scalable maps in P2Life consists of two parts. First, a data structure needs to be found, which allows to build and maintain a map of the NVME in a distributed manner. Second, an algorithm is needed that performs the redundant distribution of that data structure over the existing peers in the P2Life CAN. In this section, the distributed data structure is discussed. Section 4 shows, how the map fractions are assigned to carriers in an existing P2Life lookup-network. For the description of the approach, the symbols and terms as seen in table 2 are used.

3.1   Requirements and Operations

The approach described in this article needs to fulfill the scalability requirement and provide zooming and scrolling operations:

Scalability: Both the map data structure as well as the carrier assignment algorithm must be scalable over (i) the total number of occupied parcels, (ii) the density of occupied parcels in any given area, and (iii) the available number of carriers.

Zooming: When performing a *zoom out* operation (see fig. 3), the size of the currently visible map is increased in such a way that the current number of visible parcels is multiplied by b^2. In return, the level of detail is reduced by $1/b^2$, such that the map resolution remains constant while zooming. The *zoom in* operation analogously decreases the number of visible parcels by $1/b^2$ and increases the level of detail by b^2.
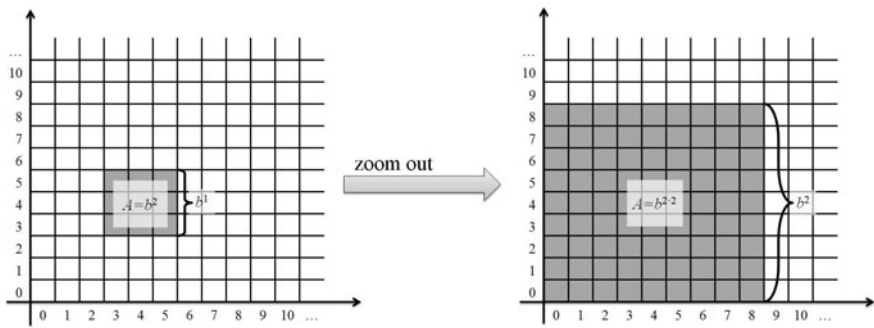
Fig. 3 Zoom out from a) $M_{(4,4),1}$ to b) $M_{(4,4),2}$. Starting level is $e = 1$ and base $b = 3$. For means of simplification, only parcels (x, y) with $x, y > 0$ are shown.

Scrolling: Scroll operations result in placing another NVME-parcel into the center of the user's current view. They can only be performed on neighbored maps of the same level e and will always result in displaying the map subsequent to the currently displayed map. E.g., a *scroll west* operation started while displaying the map $M_{(x,y),e}$ will result in displaying the map $M_{(x-b^e,y),e}$ left of it.
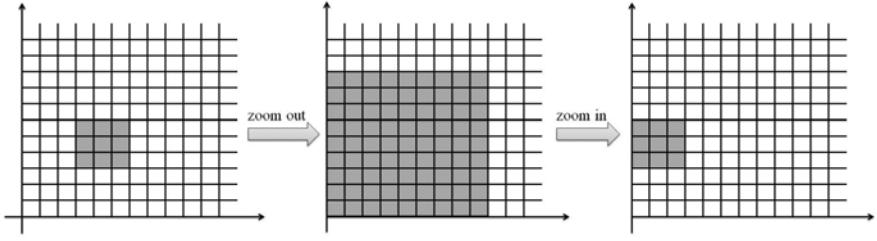
Fig. 4 Scrolling from $M_{(4,4),1}$ to $M_{(1,4),1}$ by zooming out, re-centering and zooming back in

Each scroll operation can be replaced by performing a single *zoom out* and a consecutive *zoom in* while centering the parcel $(x - b^e, y)$ (see fig. 4).

3.2 Distributed Data Structure

As the zoom operation already indicates, the maps of different levels are best organized in a rooted tree, of which the root map $M_{(x,y),e_{max}}$ defines the root and the parcel maps define the leafs. Each node represents a square map $M_{(x,y)e}$ with $b^e \cdot b^e$ parcels and contains of

1. a graphical representation (top view, see table 2) of the map. The resolution of this bitmap is fixed and independent from it's level e.
2. either 0 or at least 2 and up to b^2 children (*child maps*). A child map must be located in and wholly fit into the same area of the NVME as the current map, and therefore always has a lower level than the current map. If the number of child maps is 0, the node necessarily has level $e = 0$ and is a parcel map.
3. 0 or 1 references to the *parent map*, i.e. the map that the current map is a child map of. If there is no parent map, the current node is the root of the tree and therefore contains a (bit-)map of the whole NVME.

Nodes with only a single child-map are dispensable, since they represent exactly the same map as their child map. Therefore, a node must have either 0 or at least 2 child maps.

Additionally, it is important to point out that a map of level e is not restricted to only have child maps of level $e - 1$. On the contrary, child maps are always selected to have the *lowest possible level* in such a way that still a maximum of b^2 child maps can be maintained and the parcel maps of all occupied parcels within the current node's map can be reached by descending the tree from the current position (see fig. 5).

This way, the number of nodes representing any area of the NVME roughly scales with the number of *occupied* parcels (density) in that area and thus, each node can be reached in $O(\log N)$ time complexity. If only few occupied parcels exist, only
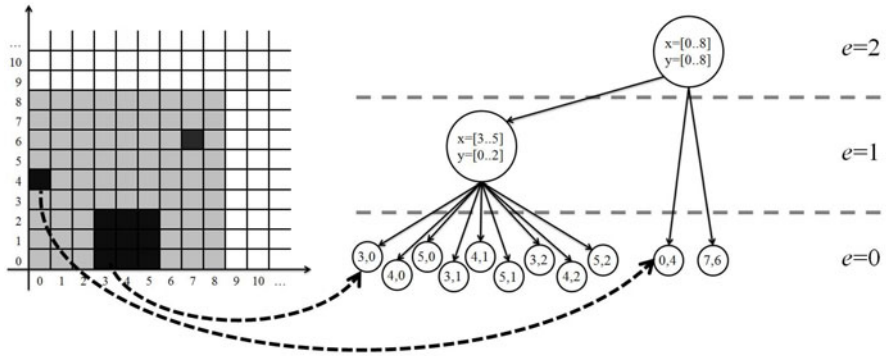
Fig. 5 NVME with few occupied parcels (dark-gray) and the resulting tree structure with $b = 3$. The light-gray area shows the size of the root map. The root node has children of levels 1 and 0.

few nodes are necessary to represent a complete map for the area. With increasing number of occupied parcels, more nodes are needed until the area is fully occupied and, in this case, represented by a B-tree, in which all inner nodes (including root) exactly have b^2 children.

3.3 Map Events

The three events resulting in a change of the current map representation are *adding*, *changing* and *removing* a top view. Thus, events are always triggered by carriers of maps on level $e = 0$, i.e. by the providers that occupy a parcel and change its appearance.

3.3.1 Updating the Tree Structure

The structure of the tree only changes in case of adding or removing a parcel map. When a new parcel map is to be added, an appropriate parent node needs to be found. This will be the one existing node in the tree that represents the smallest possible map (i.e. it has the minimum possible level), which still covers the new parcel map's coordinate (parcel). If such a node does not exist, the new parcel map is out of the current root map's focus, so that the level of the root map needs to be increased. In this case, the root map also is the parent for the new node. Now, the new node is added as a child to the determined parent node. Possible overflows of the maximum number of children per node (b^2) can only occur, when the parent node's level is at least 2, and thus can be handled by inserting intermediate nodes.

The removal of a parcel is handled in a similar manner. The parent node of the parcel map in question is informed about the removal. In case the parent node only has one child node left, the parent node is removed and the child is directly linked to the former parent's parent node.

3.3.2 Updating the Top Views

Every event generally needs to be reflected in the top views of all nodes that cover the parcel from which the event originated. Thus, each event would be forwarded up the whole tree and be processed in $O(\log N)$ time complexity (N being the number of occupied parcels), if each parent map can be reached in constant time independently from its level. Since multiple redundant copies of each node have to be maintained to ensure both scalability as well as availability (see section 4), the message complexity will necessarily be larger and might not even be in $O(N)$.

Hence, to reduce the number of top view updates forwarded upwards the tree, each node maintains a *change threshold*, up to which changes in the map are not signaled to its parent node. Only events, which significantly change the own graphical (bitmap) representation, are forwarded to the parent node. An inherently given threshold would be to require at least a single bit of the top view of the parent map to change. A more natural value could be to expect a change at least to be recognizable by an average user. In this article, however, no further details shall be given on how to find a proper threshold function.

Theorem 1. *By using any constant relative threshold Δ, the average time necessary to handle possible events is limited by a constant.*

Sketch of proof: Since each level represents b^2 times as much parcels as the previous level, a single parcel event influences only $1/b^2$ of the top view on the next-higher level. In other words, to be recognized at the next-higher level, b^2 as much average events are necessary than to be recognized in the current level. Therefore, the probability that any event is recognisable at level e is $1/(\Delta \cdot b^{2e})$. Thus, in average,

$$\frac{1}{\Delta} \cdot \left(\frac{1}{b^2} + \frac{1}{b^4} + \frac{1}{b^6} + ... + \frac{1}{b^{2e}} \right) < \frac{1}{\Delta} \cdot \frac{b^2}{b^2 - 1} = const.$$

forwards are necessary.

4 Carrier Assignment

The carrier assignment mechanism makes use of the existing lookup structure, especially of the coordinate-to-key mapping, of P2Life (see section 2.2.2). For the following description, the base of the underlying CAN (and therefore also of the coordinate-to-key mapping) is assumed to be b. Nevertheless, this serves only as simplification, since the carrier assignment will work on any CAN with only few changes.

4.1 Carrier Selection Search Pattern

A CAN peer, which maintains a copy of node data for any map $M_{(x,y),e}$, is called a *carrier* for that map (see table 2). The carriers are selected by their position in

the CAN in relation to the position of the peer that maintains the key (k_x, k_y) of the *center* of the map. A map of level $e = 0$, i.e. the parcel map of the provider at (x, y), is therefore assigned to the CAN peer managing the lookup for that coordinate. To identify carriers for all higher level maps (i.e. $e > 0$), the leftmost e digits of the key for the center position are masked out. The resulting mask is used as search pattern to identify the map's carriers. All CAN peers maintaining the keys that only differ in the first e places each in k_x and k_y, match the pattern and thus are selected as carriers for $M_{(x,y),e}$.

The number of carriers per map increases with the level of it. Maps of a lower level, which represent a small fraction of the virtual world, are replicated only on a few number of peers. Maps of a higher level are in contrast maintained by a lot of peers, since they also represent a larger fraction of the virtual world. The root map representing the whole virtual world is stored on every peer in the CAN. This does not only allow high responsiveness of the whole system, but especially gives users the possibility to validate information by crosschecking it with another carrier, since all maps except the top-views, which are given directly by the provider of a parcel, are redundantly stored on at least b^2 different and independent carriers.

Each carrier maintains a single bidirectional link to the next available carrier of its parent map. If that carrier is not available for any reason, any other can be identified using the algorithm described above.
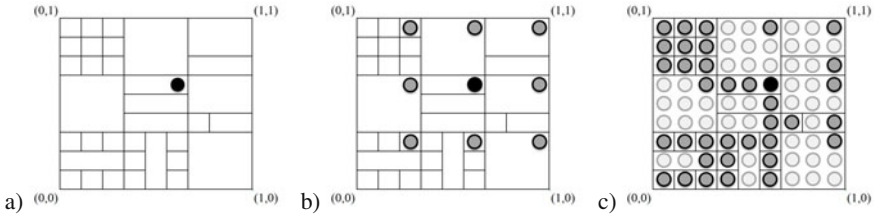
Fig. 6 Carrier assignment for an example map of a) level $e = 0$, b) level 1, and c) level 2 in an example CAN with base $b = 3$. Each key has 2 digits for both k_x and k_y. Light-gray circles show carriers that have not been assigned due to limited CAN granularity. In these cases, a single digit for k_x and/or k_y would already be sufficient to uniquely identify the CAN peer.

4.2 Granularity and Event Forwarding

Regardless of the size of the CAN, the number of digits necessary to uniquely identify a peer in it can be below or equal the number of digits identifying a carrier for a specific map. If so, a ratio of $1/b^2$ of all keys identifying carriers for the *next* level will be mapped into the same region of the CAN (since they have the same leading digits in k_x and k_y) and thus on the same peer. It is neither necessary nor advantageous to let the peer maintain multiple map copies. Instead, it will just keep a single

copy of the map. In case of adding another peer to the CAN such that the current peer's zone is splitted, the map data can be replicated on the newly added peer, if necessary.

If an update event as described in section 3.3 has to be forwarded from any level e to level $e+1$, each carrier for level e will decide, if the limited granularity of the CAN automatically also lets the current carrier be the carrier for level $e+1$. Only if this is not the case, the current carrier generates a cumulated update message to all those b^2 carriers of level $e+1$, which are within the *same region* of the CAN, i.e. which have the same e leading digits for k_x and k_y.

Fig. 6 shows, how carriers in an example CAN with base $b=3$ and $N=24$ peers are identified. The black circle in a) represents the key for the center of the requested map. Gray circles in b) and c) show the positions of the keys for the carriers of the map in level 1 and 2. For level 2 (see Fig. 6c), not all possible carrier positions in the CAN are actually used, since they identify the same CAN peer due to the limited granularity.

4.3 Complexity

When discussing the scalability of the proposed algorithms, two different aspects are of importance:

1. How long does it take to maintain consistency of the distribute map data structure in case of an event, and how many physical messages are necessary to do so?
2. How long does it take to query the map, i.e. how many hops does it take to reach the next available carrier for a map?

In this section it is shown that the approach is scalable in both aspects even without using any kind of caches. However, when each peer maintains a single cache of constant size with b^2 entries, the time and message complexity can be remarkably reduced.

Theorem 2. *Forwarding an event from level* 0 *to level* 1 *can be done within a maximum of* $2(b-1)$ *hops, i.e. in* $O(1)$ *time complexity.*

Sketch of proof: Due to the specific properties of the coordinate-to-key mapping, each CAN peer maintains direct links to all CAN peers that manage the neighboring NVME coordinates (see section 2.2.3). Finding all carriers of level 1, i.e. all CAN peers, which only differ in the first digit of k_x and k_y for any given coordinate (x,y), is thus equal to finding at most the $b-$next NVME neighbors in x- and y-direction. In the worst case, this forwarding takes $2(b-1)$ hops.

Theorem 3. *Forwarding any event originated at level 0 upwards the tree and thereby informing all affected carriers can be done in* $O(\sqrt{N})$ *time complexity, when no caches are used. By using caches with* b^2 *entries on each carrier, the complexity can be reduced to be in* $O(1)$.

Sketch of proof: Each carrier of level e will only notify the carriers of level $e+1$ within the same region. Since the number of CAN peers sharing the same e digits in k_x and k_y is below or equal to N/b^{2e}, the average path length is $\frac{1}{2}\sqrt{N/b^{2e}}$. Since all carriers can execute the forwarding process in parallel, the time complexity for forwarding an event up one level is in $O(\sqrt{N/b^{2e}})$. As discussed in section 3.3.2, the probability that an event reaches level $e+1$ at all, is $1/(\Delta \cdot b^{2(e+1)})$ due to the change threshold Δ. The average number of sequential hops necessary to handle an event is now calculated by summing up the number of hops for forwarding an event up a single level weighted by the probability that such a forward is necessary at all. Thus, the average hop count is

$$\sum_{e=1}^{e_{max}} \frac{1}{\Delta \cdot b^{2(e+1)}} \cdot \frac{1}{2}\sqrt{\frac{N}{b^{2e}}} < \frac{\sqrt{N}}{2\Delta} \cdot \frac{1}{b^5 - b^2} \in O(\sqrt{N})$$

without using caches. When caches are used, in which each carrier stores the addresses of all b^2 carriers of level $e+1$ within the same region, the average path length for forwarding the event a single level is reduced to 1. Again, using the probability of $1/(\Delta \cdot b^{2(e+1)})$ (see above) for any event to reach level $e+1$ at all, the average number of sequential hops is

$$\sum_{e=1}^{e_{max}} \frac{1}{\Delta \cdot b^{2(e+1)}} < \frac{1}{\Delta} \cdot \frac{1}{b^4 - b^2} \in O(1)$$

and therefore independent from the number of providers.

Theorem 4. *The average number of physical messages necessary to forward any event originated at level 0 upwards the tree and thereby informing all affected carriers, is in $O(\sqrt{N})$, if no caches are used, and in $O(\log N)$ otherwise.*

Sketch of proof: The maximum number of messages needed to forward an event from level e to level $e+1$ is $b^{2(e+1)}$. Each of these messages needs $\frac{1}{2}\sqrt{N/b^{2e}}$ hops in average (see above). Thus, the total number of physical messages generated to forward an event up one level is

$$\frac{1}{2}b^{2(e+1)} \cdot \sqrt{N/b^{2e}} = \frac{1}{2}b^{e+2} \cdot \sqrt{N}$$

Again, since the probability that an event has to be forwarded up to level $e+1$ is $1/(\Delta \cdot b^{2(e+1)})$, the average number of physical messages generated to handle any event is

$$\frac{1}{2\Delta}\sum_{e=0}^{e_{max}-1} \sqrt{N}\frac{b^{e+2}}{b^{2(e+1)}} = \frac{\sqrt{N}}{2\Delta}\sum_{e=0}^{e_{max}-1} \frac{1}{b^e} < \frac{\sqrt{N}}{2\Delta}\frac{1}{b-1} \in O(\sqrt{N})$$

without using caches. If caches are used, in which each carrier stores the b^2 addresses of the available carriers for level $e+1$ (as above), only $b^{2(e+1)}$ physical messages are necessary to propagate an event from level e to level $e+1$ (one per

carrier). The average number of physical messages for handling any event is therefore reduced to

$$\frac{1}{\Delta} \sum_{e=0}^{e_{max}-1} \frac{b^{2(e+1)}}{b^{2(e+1)}} = \frac{e_{max}-1}{\Delta} \in O(\log N)$$

The maximum level e_{max} is in $O(\log N)$, because it correlates linearly to the height of the distributed tree structure described in section 3.2.

Theorem 5. *The average path length for handling a zoom-out request is in $O(\sqrt{N})$ without using caches and in $O(1)$ otherwise. A zoom-in request can usually be handled with only a single hop and thus in $O(1)$ complexity.*

Sketch of proof: Each carrier of level e can reach a carrier of level $e+1$ in an average of $\frac{1}{2}\sqrt{N/b^{2e}} \in O(\sqrt{N})$ hops (see above). If caches are used, only a single hop is necessary. Since every carrier stores exactly one *bidirectional* link to a carrier of its parent map (see section 3.3), for each child map at least one carrier is known directly and can be reached in a single hop. Only, if this link is not valid, $\frac{1}{2}\sqrt{N/b^{2e}}$ hops are necessary to reach the next available carrier.

For all proof sketches it is important to point out that they all refer to the *same* cache with b^2 entries storing the addresses of the carriers of level $e+1$, which are in the same region as the current carrier. Thus, with a constant amount of only b^2 dynamic entries on each peer, the reduced time and message complexities as described above can be reached.

5   Conclusion and Future Work

This article presents an algorithm for building scalable maps of NVMEs in a distributed and reliable manner. It shows that in the P2Life NVME, the management of the maps can be done in $O(\sqrt{N})$ complexity without caches, and in $O(1)$ time- and $O(\log N)$ message complexity by using a single cache with b^2 entries on each peer.

The main advantage of this approach is its massive redundancy. To make a map of level e fully unavailable, b^{2e} peers have to be compromised or go unexpectedly offline. Malicious behaviour can still be detected using byzantine tolerant protocols, if not more than $b^{2e}/3$ peers are compromised.

A prototype implementation for the map management currently is in development and will be integrated into the P2Life prototype. To reduce the load, a single peer has to carry for answering user requests, a distributed dynamic cache utilizing the bandwidth of users, who already loaded map fractions, will also be part of future work.

References

1. Coltzau, H.: P2Life: An infrastructure for networked virtual marketplace environments. In: International Journal of Intelligent Informaiton Processing (IJIIP) 1(2),1–13 (2010)
2. SecondLife, http://secondlife.com/ (July 5, 2011)

3. Knutsson, B., Lu, H., Xu, W., Hopkins, B.: Peer-to-peer support for massively multiplayer games, in INFOCOM 2004. In: Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1, p. 107 (2004), http://dx.doi.org/10.1109/INFCOM.2004.1354485

4. Yu, A.P., Vuong, S.T.: MOPAR: a mobile peer-to-peer overlay architecture for interest management of massively multiplayer online games, in NOSSDAV 2005. In: Proceedings of the International Workshop on Network and Operating Systems Support for Digital Audio and Video, pp. 99–104. ACM, New York (2005)

5. Bharambe, A., Pang, J., Seshan, S.: Colyseus: a distributed architecture for online multiplayer games, in NSDI 2006. In: Proceedings of the 3rd Conference on Networked Systems Design & Implementation, p. 12. USENIX Association, Berkeley (2006)

6. Botev, J., Hohfeld, A., Schloss, H., Scholtes, I., Sturm, P., Esch, M.: The Hyperverse – concepts for a federated and torrent-based web. Int. J. Adv. Media Commun. 2(4), 331–350 (2008)

7. Frey, D., Royan, J., Piegay, R., Kermarrec, A., Anceaume, E., Le Fessant, F.: Solipsis: A decentralized architecture for virtual environments. In: International Workshop on Massively Multiuser Virtual Environments (MMVE 2008). IEEE (2008), http://fabrice.lefessant.net/papers/mmve2008.pdf

8. Iimura, T., Hazeyama, H., Kadobayashi, Y.: Zoned federation of game servers: a peer-to-peer approach to scalable multi-player online games. In: Proceedings of 3rd ACM SIGCOMM Workshop on Network and System Support for Games, NetGames 2004, pp. 116–120. ACM, New York (2004)

9. Schmieg, A., Stieler, M., Jeckel, S., Kabus, P., Kemme, B., Buchmann, A.: Psense - maintaining a dynamic localized peer-to-peer structure for position based multicast in games. In: Proceedings of the 2008 Eighth International Conference on Peer-to-Peer Computing, P2P 2008, pp. 247–256. IEEE Computer Society, Washington, DC (2008)

10. Kawahara, Y., Morikawa, H., Aoyama, T.: A peer-to-peer message exchange scheme for large scale networked virtual environments. In: Proceedings of the The 8th International Conference on Communication Systems, ICCS 2002, pp. 957–961. IEEE Computer Society, Washington, DC (2002)

11. Hu, S.-Y., Chang, S.-C., Jiang, J.-R.: Voronoi state management for peer-to-peer massively multiplayer online games. In: 5th IEEE in Consumer Communications and Networking Conference (CCNC 2008), pp. 1134–1138 (2008), http://dx.doi.org/10.1109/ccnc08.2007.255

12. Bharambe, A., Douceur, J.R., Lorch, J.R., Moscibroda, T., Pang, J., Seshan, S., Zhuang, X.: Donnybrook: enabling large-scale, high-speed, peer-to-peer games. In: Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication, SIGCOMM 2008, pp. 389–400. ACM, New York (2008)

13. Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A scalable content-addressable network. In: Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM 2001, pp. 161–172. ACM, New York (2001)

Node Behaviour Driven Network Topology Adaption

Coskun Akinalp and Herwig Unger

Abstract. The performance of networks is usually linked to network-related aspects such as bandwidth and protocols. However, behind every network element there are users who may organize their relations to others. But which organizational structures are successful for individuals and groups and how can the data traffic be optimized for network infrastructure? By means of a network creation game, this paper examines the results of different network structures and compares individual decisions to grouping behaviours. The experiments were simulated in the form of a ring network which enables individual nodes to create long links (direct connections between nodes) to other nodes based on their account values. For one group within the network, an extended approach was tested, i.e. there was the option of having the behaviours of decentralized"democracy" or centralized-hierarchical"kingdom" organizations? manage the long links within the group. The results of the simulations for different power law distributions of messages show that cooperation within the structures, compared to the individual node, differ in terms of routing steps, account values and the stable management of long links. We find that the hierarchical approach represents successful network transactions for different grouping sizes.

1 Introduction

In the face of current research in computer network science focused on game theory [1, 7, 8, 9, 10, 11, 12] and the future of network interaction, [13, 14] the following question arises: How can efficient network managing be achieved, or, in how far does the behaviour of a single or a cluster of users with a certain profile react

Coskun Akinalp · Herwig Unger
FernUniversitt in Hagen, Faculty of Mathematics and Computer Science
e-mail: coskun@akinalp.com, herwig.unger@fernuni-hagen.de

to and influence network traffic? Research with limbic characters applied to minority games forms the starting point for this work [2, 3]. In this paper we present a different approach of analysing the impacts of network routing within a network. The conducted experiment discussed in this paper can quickly answer the question of whether an individual or a group strategy - box (de-central or central) is more successful.

To answer this question, we consider 1) the routing steps within the ring structure, 2) the management of long links and 3) the account value of the nodes. The set-up of the simulation can be briefly described as follows:

- within a ring network, we transfer messages with a source and targetnode
- the ring network will never be changed
- for every transfer of messages, each node earns a certain amount on his account
- with these account values, nodes can create long links (shortcuts) to other nodes, which, however, creates costs
- each node needs to pay for the maintenance costs to manage the long link
- within the network one group can be defined which could behave decentrally-uniformly or centrally-hierarchically
- the size of the group can be changed from 5% to 50% of the total number of nodes
- the messages will be distributed according to the power law distribution $p(x) = e^{-\lambda x} - e^{-\lambda(x+1)}$ for $\lambda(0,....0,3)$

The simulations were conducted by trainees using the simulations tool P2PNetsim [5] [4].
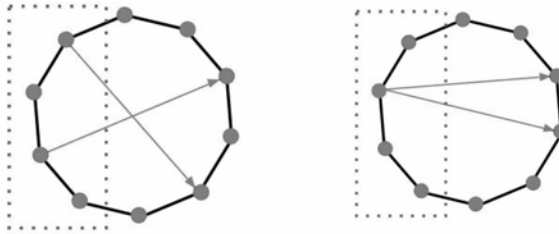
Fig. 1 Decentral and central long link creation

2 Models

2.1 Individual

In Appendix A all symbols and abbreviations used are listed. This section describes the different models and their behaviours.

The account of the nodes: Every node manages an account enabling it to create and maintain long links. By forwarding a message to another node (ring or long

link, the node also earns a certain amount of money). The actual earning represents the account value for the next simulation step $B_N(t) = B_N(t-1) + C_{SL} + C_{LL}$. The payment for the accounts is made by the messages.

Creating long links: Each node can create one long link in every simulation step. First, it determines to which nodes it can create long links (this decision is based on the budget and on already created links), then it chooses the node to which it has forwarded the most messages from this set. In order to o create a long link, the following condition must be fulfilled : $B_{Ni}(t) = B_{Ni}(t-1) - d(u,v)$, in addition, the node also has to pay maintenance costs for the long link in every simulation step $B_{Ni} = B_{Ni} - CM_{LL}$.

Deleting long links: There are two conditions under which a long link can be deleted: First, if a node has not enough money to pay for its maintenance $B_N < CM_{LL}$ and second, if a link has not been used for a long time $\tau \geq \tau_{max}$, where the time past since the last usage is measured. If a node has enough money to maintain only a few of its links (implying that some links must be destroyed, and some not), the nodes most recently used are preserved.

2.2   De-central

In the decentral approach, all nodes share the same account, and all the nodes pay their income into this account. If a long link needs to be created, the money is taken from this account. The average account balance per node is the balance of this common account divided by the total number of nodes in the box. All costs and earnings are shared. In this approach, the box as a whole determines the creation or deletion of long links, i.e. all nodes in the box decide jointly which long links are the most beneficial ones to create, and these links are created using money from the shared account. The same principle applies to the deletion of long links: all nodes jointly decide which links should be deleted. More than one node from a decentral box can create a long link to a target node, but in each simulation step only one long link from a box can be created. The decision of which long link should be created is based on the message transfer probabilities: the one with highest probability is created.

2.3   Central

In the central approach, there is one main node (the so-called king or central node) which collects all the money earned by the nodes in the box. In other words, if a node other than the central node earns money, it pays it into the central node account. This results in the fact that the account balances of all nodes (except the central node) are always zero and, consequently, these nodes cannot create any long links, since they cannot access the central node account. Only the central node itself can use the money to create and maintain long links.

2.4 Node Rules

Node earning: Each node earns money every time a message uses its link. If a short link is used, then the node will get C_{SL} money and its balance will be increased by this amount. The price for using a long link is C_{LL}, and this amount of money is transferred to the respective node's account. Regearding nodes in a box, instead of flowing into the individual nodes accounts, it is transferred to the common account (democracy) or to central account (kingdom). If a message is transferred within a box, the earned money equals C_{BL}. Obviously, for messages sent from a box to the outside, the earning is the same as described for the individual node. For both models the account values will be updated as described $B_N(t) = B_N(t-1) + C_{SL} + C_{LL} + C_{BL}$.

Nodes costs: Nodes have to pay for each created long link once they create it ad well as for their maintenance in every simulation step. The costs of creating a link are equal to the distance between the nodes, and the balance account of a node is reduced by: $B_N(t) = B_N(t-1) - d(u,v)$. The maintenance costs equal CM_{LL}, thus the balance is reduced by this amount in every simulation step: $B_N(t) = B_N(t-1) - CM_{LL}$.

2.5 Messages

Message distributions: Senders and receivers are subject to an exponential distribution. The first set of probabilities is calculated according to the formula $p(x) = e^{-\lambda x} - e^{-\lambda(x+1)}$ (where lambda is the slope coefficient and x is the number of a node), then it is normalized (so that the sum of all probabilities is equal to one), and randomly distributed to the nodes. For lambda being equal to 0 the distribution is uniform.

Creating messages: Each node can create a new message targeted to any other node. In one simulation step it can create N_M new messages at maximum. The probability for node i to create a new message is equal to p_{Mi}. The receiver of the message is chosen according to the probability distribution of those nodes being a target node p_{Mi}. Both probabilities can be set individually for every node, thus any kind of message distribution can be achieved.

Message costs: A message needs to pay each node for using its link. For using a short link, it has to pay C_{SL} units, for using a long link it has to pay C_{LL}. However, if a message travels within a box, it will have to pay C_{BL} for any transition within the node (independently of the path length). The money a message pays is deducted from its account. Each message has an initial account balance equal to B_{Minit}. The message costs are calculated by $B_M(t) = B_M(t-1) - C_{SL} - CM_{LL} - C_{BL}$.

Message routing: The basic message routing in a ring is described by a source i to a destination j getting the minimum $d = \min(d_1, d_2)$ from $d_1 = n_i - n_j mod N$ and $d_2 = n_j - n_i mod N$. It is important to mention nodes only act locally and do not have any information about long links. The long link will only be used if a long link is on the route of the ring route calculations, otherwise it will not be used. However, this rule is only valid for individuals. Groups are aware of the fact that long links exist and use them to calculate the minimum path to the destination.

3   Simulation and Results

Simulations were performed for a number of cases (a total of 252) with different parameter settings. The parameters were as follows:

1. ring size: $N = 1000$ nodes
2. total number of messages: 50000
3. messages sent in one simulation step: 100
4. long links deleted after not having been used: 100 steps
5. boxes size: 0; 50; 100; 150; 200; 250; 300; 350; 400; 450; 500 nodes
6. boxes type: central / decentral
7. λ coefficient for probability distributions: 0.0; 0.005; 0.01; 0.02; 0.05; 0.075; 0.1; 0.125; 0.15; 0.2; 0.25; 0.3

As mentioned before, the results will be discussed with regard to the measurements of routing steps, long links management and account values.

Calculation rules for nodes: Each node performs the following computations in every simulation step:

1. deleting long links which are not used ($\tau \geq \tau_{max}$)
2. paying for long link maintenance (if impossible, deleting some of the long links)
3. creating new long links (if possible)
4. forwarding / receiving messages
5. updating statistics

Routing steps: The results of the simulations indicate a clear impact of λ on the routing steps. Depending on the distribution, they also show an advantage of group behaviour compared to individual behaviour. A high λ states that single nodes receive a lot of traffic, which converts a single node to central behaviour of one node.
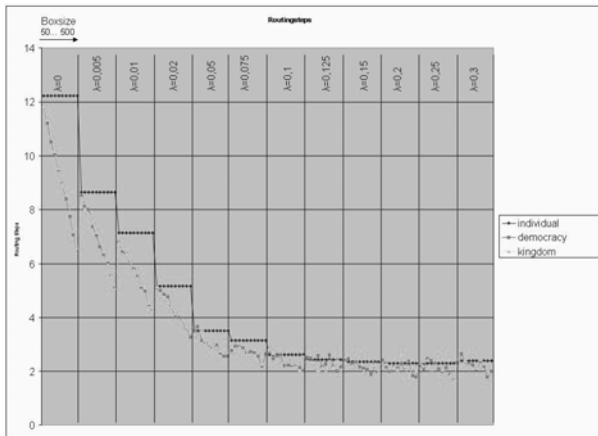
Fig. 2 Routing step by lambda and box size for individual, decentral and central approaches

Long link management: The results of the analysis of long links differ from the analysis of the routing steps. The following figure shows the created and deleted long link and the relative view of the surviving long link to the newly created one.

As with the routing steps, the impact of lambda is significant both for the creation and the deletion of long links. However, the diagrams also reveal that while the central approach indicates stability in the management of long links, the decentral approach
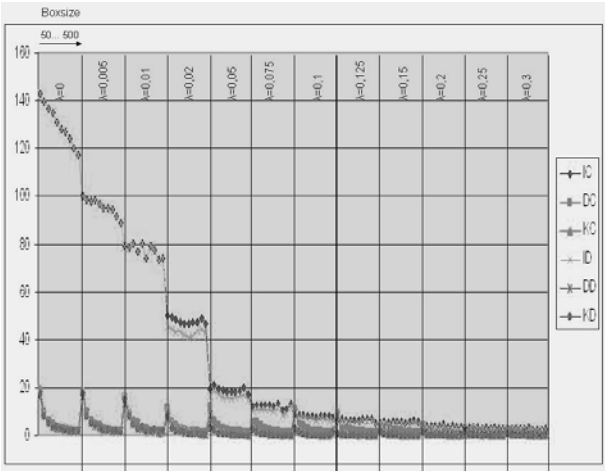
Fig. 3 Created and deleted long links
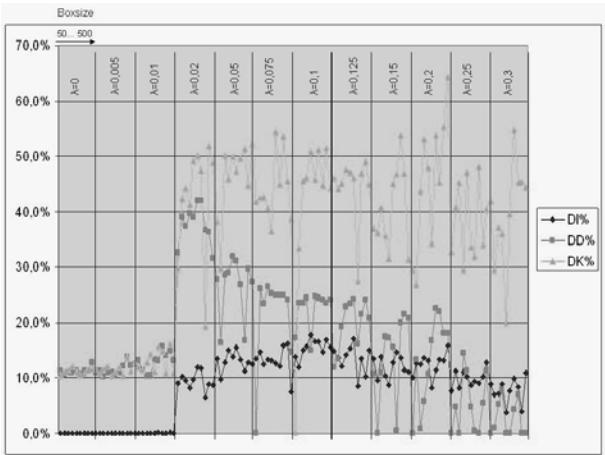
Fig. 4 Percentage of surviving long links

shows a decrease in the number of surviving long links. The increase in long links for small group sizes also illustrate that these groups behave like individuals.

Account values: The achievement of account values represents the summary of the different approaches compared with regard to the different distributions. As discussed before, lambda and the box size influence the outcome of each strategy. The next figures represent the average final account regarding lambda (the average box size) and the box size (the average lambdas). The long link maintenance discussion has already revealed a uniform impact of the distribution on the number of creations and deletions of long links by individual nodes, which can be explained by the fact that there are no earnings for small lambdas. With higher lambdas, the behaviour of the individual nodes moves towards a single, centralized approach which increases the account values. The box size presents a different result. The central approach
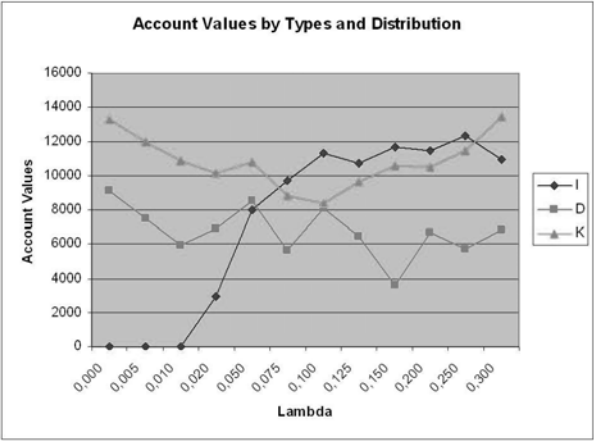
Fig. 5 Account value by lambda
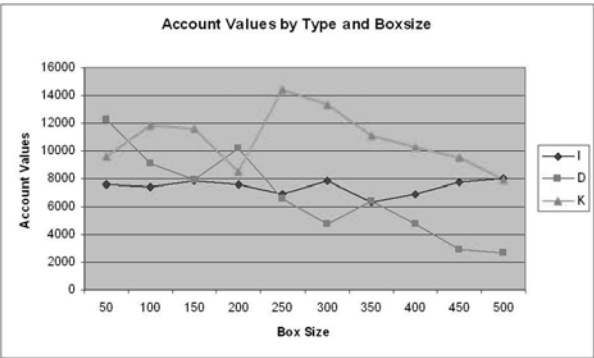
Fig. 6 Account value by box sizes

indicates successful modelling, which decreases with the box size regarding the individual behaviour. It is interesting that the decentral approach is extremely sensitive to the setting of lambda and the box size of the groups.

The abbreviations in the figures are I: individual, D: decentral, K: central, C: creation, D: deletion, %-percentage.

4   Conclusion

This paper has introduced a network creation game model to compare individual and grouping (decentral and central) behaviours. Technical and economical aspects have been discussed and analysed. The analysis aimed at finding out which strategy is more successful in terms of routing, long link maintenance efforts and account values. The results of this simulation can be applied to network structure developments with economical demands and indicate that the cooperation management of networks reflects a successful strategy for the network, revealing a decrease with growing box sizes. Another interesting outcome is that the decentral approach has proven to be worse than the central approach, which seems plausible due to the fact that a higher number of long links is managed compared to the central approach. Consequently, in the central approach all nodes benefit more than in the decentral approach. With an increasing lambda, these results may slowly shift towards an individual behaviour. The success of a network is strongly related to the distribution of the message and the box size of the groups. These frame conditions need to be analysed in order to optimise e.g. routing aspects. Fault risk, reliability and security aspects, however, were not taken into consideration by this analysis.

5   Future Work

Further fields of research might include 1) the analysis of multiple groups with different sizes and strategies within a network and their relation to each other; 2) giving nodes the ability to be part of two boxes; 3) introducing the limbic characters into the nodes and analyse their behaviour in different groups with different lead characters and distributions; 4) changing the values for different costs including dynamic (i.e.supply and demand) changes; 5) sending messages in bursts. This leads to a new perspective of analysing network structure in order to develop it under economical and game theoretical aspects.

Appendix

Table 1 Table of Variables

Symbol	Description
N	number of nodes on a ring (from 10 to 100,000)
$d(u,v)$	distance between nodes u and v (calculated on a basic ring)
C_{SL}	cost of using a short link (equal to 1)
C_{LL}	cost of using a long link (equal to d(u,v))
C_{BL}	cost of travelling within a box (equal to 1)
CR_{LL}	creation cost of a long link (equal to d(u,v))
CM_{LL}	maintenance cost of a long link (equal to 1)
B_{Ni}	account balance of node i
B_D	account balance of de-central box (shared account)
B_K	account balance of the central node (in kingdom)
B_M	account balance of message
B_{Mini}	initial account balance of a message $d_{max}(u_{source}, v_{destination})$
τ	time for which a long link was not used (continuously)
τ_{max}	maximum time, for which a long link not used before it is deleted links
N_M	maximal number of messages that a node can create in one simulation step
p_{Mi}	probability of creating a new message by node i $(0 < p_{Mi} \leq 1)$
p_{Ti}	probability of a node being a target node $1 = \sum_{i=1}^{N} p_{Ti}$

References

1. Alpcan, T., Buttyán, L., Baras, J.S.: GameSec 2010. LNCS, vol. 6442, pp. 87–106. Springer, Heidelberg (2010)
2. Akinalp, C.: The Limbic Characteristic and El-Farol Games. In: 10th International Conference on Innovative Internet Community Systems (I2CS). GI-Edition Lecture Notes in Informatics. Jubilee Edition, Bangkok (2010)
3. Häusel, H.G.: Think Limbic. Rudof Haufe Verlag Gmbh, Muenchen (2007)
4. Jooybar, H., Wachowski, J.: Working individually or in cooperation- What is more successful. In: Project Report Supervised by Coskun Akinalp. FernUniversität-Hagen (August 25, 2010)
5. Coltzau, H.: Specification and Implementation of Parallel P2P Network Simulation Environment. University of Rostock (2006)
6. Koed-im, S., Hamad, Y.: Limbic Character in Network Creation Game Result of Simulation Parameter dependencies. In: Project Report. FernUniversität-Hagen (September 1, 2010)
7. Fabrikant, A., Luthra, A., Maneva, E., Papadimitriou, C.H., Shenker, S.: On a network creation game. In: Proceedings of the Twenty-Second Annual Symposium on Principles of Distributed Computing (PODC 2003), Boston, Massachusetts, pp. 347–351 (2003)
8. Albers, S., Eilts, S., Even-Dar, E., Mansour, Y., Roditty, L.: On Nash equilibria for a network creation game. In: Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm (SODA 2006), Miami, Florida, pp. 89–98 (2006)

9. Demaine, E.D., Hajiaghayi, M.T., Mahini, H., Zadimoghaddam, M.: The price of anarchy in network creation games. In: Proceedings of the Twenty-Sixth Annual ACM Symposium on Principles of Distributed Computing (PODC 2007), Portland, Oregon, pp. 292–298 (2007)
10. Corbo, J., Parkes, D.: The price of selfish behavior in bilateral network formation. In: Proceedings of the Twenty-Fourth Annual ACM Symposium on Principles of Distributed Computing (PODC 2005), Las Vegas, NV, pp. 99–107 (2005)
11. Anshelevich, E., Dasgupta, A., Kleinberg, J., Tardos, E., Wexler, T.: The price of stability for network design with fair cost allocation. In: Proc. of 45th Annual IEEE Symposium on Foundation of Computer Science, Rome, Italy, pp. 295–304 (2004)
12. Fiat, A., Kaplan, H., Levy, M., Olonetsky, S.: On the price of stability for designing undirected networks with fair cost allocations. In: Proc. of 33rd Annual International Colloquium on Automata, Languages, and Programming. Springer, Heidelberg (2006)
13. Cisco-White-Paper: Cisco Visual Networking Index Forecast and Methodology 2009-2014 (June 2010),
    http://www.cisco.com/en/US/solutions/collateral/ns341/
    ns525/ns537/ns705/ns827/White_paper_c11-481360_
    ns827_Networking_Solutions_White_Paper.html (cited August 16, 2011)
14. Cisco-White-Paper: Hyperconnectivity and the Approaching Zettabyte Era (June 2010),
    http://www.cisco.com/en/US/solutions/collateral/ns341/
    ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.html
    (cited August 16, 2011)

Cellular Neural Networks Based Time-Series Approximation for Real Time Systems' Modeling-and-Identification and Behavior Forecast in Transportation: Motivation, Problem Formulation, and Some Research Avenues

Jean Chamberlain Chedjou and Kyandoghere Kyamakya

Abstract. This paper discusses the potential of using "time series approximation" for mathematical modeling, online system identification and forecasting of the dynamical behavior of scenarios in the field of traffic and transportation. The tremendous attention devoted to both modeling and forecasting (in transportation) is justified whereby some challenges and unsolved research issues are discussed. Due to the time-varying dynamics experienced by transportation related systems/scenarios, an appropriate identification process is necessary and should be applied to determine the parameter settings of the corresponding mathematical models in real time. The concept of a simulation and computing platform design based on the cellular neural network (CNN) paradigm will be presented. Then the capability to study the spatio-temporal and time-varying dynamics exhibited by time-varying transportation systems/scenarios will be demonstrated. In the essence, we develop a concept that uses the CNN model as a universal mathematical system-model and/or system-model approximator.

1 Introduction

The last decades have witnessed a tremendous attention on applying analytical methods to process sensors data obtained from the dynamics of real-time traffic scenarios/events [1-11] and Refs. therein. The interest devoted to the analytical methods is explained by the potentials of modeling traffic dynamics mathematically at macroscopic, microscopic and mesoscopic levels [12, 13] and, by the

Jean Chamberlain Chedjou · Kyandoghere Kyamakya
Alpen-Adria-University Klagenfurt, Institute of Smart System Technologies,
Transportation Informatics Group, 9020 Klagenfurt, Universitaetsstrasse 65-67, Austria
e-mail: {jean.chedjou,kyandoghere.kyamakya}@uni-klu.ac.at

flexibility of controlling this dynamics by monitoring or tuning the parameter- settings of the analytical models. Further, mathematical models which are of necessary importance to develop flexible and robust simulation platforms of the traffic dynamics could lead to an efficient analysis, control and forecasting of the real traffic dynamics [2, 3, 8, 13]. Such a dynamics which experiences spatio-temporal variations is generally modeled by complex differential equation (i.e. nonlinear and/or stiff equations with time varying coefficients) and/or stochastic differential equations [11, 7, 8, 6]. The coefficients of these equations which are generally time- varying are evaluated (given a specific traffic scenario) through an identification process. The challenging contribution in this work is to develop a systematic and fundamental analysis platform to derive the mathematical models describing the traffic dynamics from the experimental traffic sensor data generally expressed in the form of time- series.

The time series' mathematical modeling is of necessary importance due to its interesting applications in forecasting and control [2, 3, 8, 13]. Several classical and well-known linear and nonlinear methods are being currently used for time series modeling.

The first class is based on *Univariate time-series models* [2, 8]. This class includes linear principles such as the autoregressive (AR), the moving-average (MA), the combination of both (ARMA) , the *integrated models* which exploit a stationary principle based on ARMA, *Fractional models*, *nonlinear time series models* (e.g. threshold regression, smooth transition, random coefficients, ARCH models, etc..), *Neural networks* [14, 15], and *the state-space modeling* [7, 8]. For each of these methods the type of the mathematical model is pre-defined and the identification process is used to derive the coefficients of the model. This is achieved by exploiting experimental data (time series) of the scenario under investigation. This is a serious limitation as the form of the appropriate mathematical model cannot be depicted accurately from observations on experimental data (time series). Therefore, these methods appear not efficient enough for the modeling of complex scenarios experiencing stiffness, time varying and stochastic dynamics just to name a few.

The second class of classical methods for modeling time series is based on *Ordinary differential equations (e.g. deterministic-ODEs and/or stochastic- SODEs)* [6, 7, 8, 10, 16]. The advantages provided by this class of methods are the possibility of using ODEs for real-time simulations and the flexibility of controlling the scenarios (be them deterministic or stochastic) by monitoring the parameter- settings of ODEs. The seminal mathematical model (ODEs)- the so-called *"perfect model"* was proposed by Lorenz [17] for modeling experimental time series data. Another leading formalism based on ODEs- the so-called "S-system formalism" is being currently used for identifying the parameter- settings of ODEs to model experimental time series data [6] and Refs. Therein. For these modeling concepts based on ODEs, the general principle is focused on identifying both *structure a*nd *parameters* of differential equations (i.e. *ODEs and/or SODEs)* from experimental data (i.e. time series data). In addition to the time series data and, in order to ease

the identification process, a model space must be well-defined to specify the possible forms of the ODEs and/or SODEs and their appropriate parameters as well. *The perfect model scenario* [10] is a well-known prototype of coupled ODEs which is classically used for time series' modeling. For this technique, the mathematical model is pre-defined and the identification process is used to derive the coefficients of the model. Due to the fact that the appropriate mathematical model cannot be depicted accurately from observations on experimental data (i.e. time series), this method appears not efficient enough for the modeling of complex scenarios experiencing stiffness, time varying and stochastic dynamics just to name a few.

With regard to the above limitations, the key challenge which is the main objective of this paper is to enrich the literature by developing a universal-, systematic- and fundamental- analysis framework for the mathematical modeling of time series data. The key requirements to be fulfilled are threefold. (a) There must be no prior knowledge of the system or scenario under investigation; - only experimental data (time series) are known. (b) There must be no prior knowledge of the type of appropriate ODEs (i.e. coupled and/or uncoupled ODEs, linear and/or nonlinear ODEs, the degree and/or the order of the ODEs, etc…) to model the experimental time series data. (c) There must be no prior knowledge of the type of nonlinearity exhibited by the scenario under investigation (time series data).

The rest of this paper is organized as follows. Section 2 presents the CNN paradigm and some motivations of modeling and analyzing time series data using this paradigm. Section 3 presents the CNN as *a universal system model*. A full description of the optimization platform (NAOP) leading to the CNN- templates optimization is provide. Section 4 deals with the proof of concept. An application example is considered and it is demonstrated how CNN- templates can be optimized to transform time series data unto simple forms of nonlinear ODEs representing the CNN model. The van der Pol equation is considered as a good prototype of ordinary differential equation experiencing stiffness. For this equation, corresponding time series data are obtained from the state space representation of components of the vector flow. These data are exploited to optimize the corresponding CNN- templates. The last part (section 5) deals with concluding remarks. We provide a brief summary of the results obtained. We discuss some advantages and limitations of the approach developed in this paper. We end this section by proposing some open research avenues as outlooks.

2   The Cellular Neural Networks (CNN) Paradigm For Time- Series Modeling and Analysis: Motivations

The concept of Cellular Neural Network (CNNs) is defined as *"2D or 3D arrays of mainly locally connected nonlinear dynamical systems called cells, whose dynamics are functionally determined by a small set of parameters which control the cell interconnection strength"* [18]. These parameters determine the connection pattern, and are collected into the so-called cloning templates (i.e. space

invariance templates), which, once determined, define the processing of the whole structure [18, 19, 20]. The original elementary (or isolated) CNN cell was proposed in terms of an electronic structure made up of static voltage source, static current source, linear capacitor, nonlinear voltage-controlled current sources and few resistors [18, 19,20] as shown in figure1a. The isolated CNN- cell is characterized by three main blocks: the input u_i, the state x_i and the output y_i. The cells of the complete CNN- processor can be arranged in a k-dimensional square grid which is the most commonly used CNN architecture. These cells are locally connected via programmable weights called templates. These templates can be monitored to make it programmable the CNN cell array. Hence, the essential of the technology based on the CNN paradigm is located in templates. The CNN computing platform used in this paper exploits the structure of the state-control CNN (SC-CNN) which is modeled by Equation (1).

$$\frac{dx_i}{dt} = -x_i + \sum_{j=1}^{M}\left[\hat{A}_{ij}x_j + A_{ij}y_j + B_{ij}u_j\right] + I_i \tag{1}$$
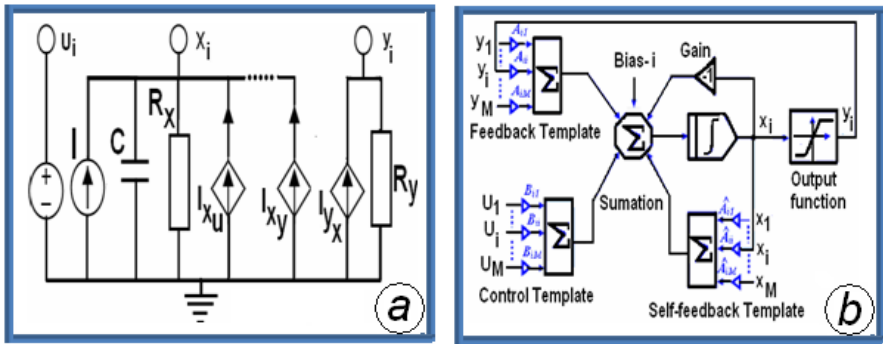
Fig. 1 (a) Basic architecture of an isolated CNN- cell. (b) SIMULINK graphical representation of a SC-CNN cell coupled to (M-1) neighbors

The coefficients $\hat{A}_{ij}$, A_{ij} and B_{ij} are the self-feedback template, feedback template and control template, respectively. The schematic representation of a state-control CNN cell coupled to (M-1) neighboring cells is shown in figure 1b. I_i is the bias value and y_j is the nonlinear output sigmoid function of the cell j. u_j denotes the input value of the cell j and x_j represents the state of the cell j [20].

A challenging innovation nowadays performed in Intelligent Transportation Systems (ITS) is the successful combination of advanced sensors, computer, electronics and communications technologies for designing and implementing devices for Advance Driver Assistance Systems (ADAS) [2, 21, 22]. Due to the complex

structures of ADAS, the tremendous research focused on ADAS aims at improving their performance, efficiency and robustness with regards to the dynamic environment and the dynamic nature of traffic scenarios as well [21]. The most common approach to design ADAS exploits distinct kind of real- traffic data which are continuously captured by sensors. These data are used for the analysis, management, planning and forecasting of the traffic dynamics. To achieve this, many classical methods are being used, ranging from the modeling and forecasting based probabilistic analysis (case of stochastic traffic) to the modeling based ODEs (case of deterministic and/or stochastic traffic). As it is well-known that real-traffic scenarios experience spatio-temporal dynamics, the key challenge is related to deriving an appropriate model to capture, with high accuracy, the complex dynamical behavior exhibited by traffic scenarios. Statistical methods are possible analytically when the distributions are Gausian as a function of standard deviation [10, 23]. With these methods the analysis of the traffic dynamics is based on samples and therefore could provide results with corresponding levels of confidence. Nevertheless, it is very tough to analyze and forecast the traffic dynamics as accurately as possible using statistical analysis. In the other hand, some classical methods based on ODEs (see section 1) are being used for traffic modeling and forecasting. Due to the fact that these methods are based on pre-define mathematical models it is challenging to model the spatio-temporal traffic dynamics as accurately as possible using the underlined approaches.

In essence, mathematical models to describe the spatiotemporal dynamics exhibited by real-traffic scenarios are generally materialized by very complex and stiff nonlinear ODEs, PDEs, and SODEs with time varying coefficients. These models are untrackable analytical. Numerical simulation using Von Neumann computing architecture is prone to accumulation of round-off errors (due to stiffness). Further, numerical simulation is very time consuming as well. With regard to these limitations, the key challenge can be formulated around the following questions. (1)- *"Is it possible to exploit experimental time series data to derive a corresponding mathematical model to accurately capture/describe the spatiotemporal dynamics exhibited by real-traffic scenarios?". (2)- "Is it possible to perform accurate and Ultra-fast simulation of the complex mathematical models of the traffic dynamics?"*

This paper attempts to provide answers to the key research questions formulated above by developing a systematic analysis platform to model time- series data with the CNN- paradigm. This paradigm which is formulated around nonlinear first- order ODEs is advantageous as its flexibility could be exploited to easily derive the corresponding CNN- templates in order to perform analog computing which is well-known as being an appropriate computing platform to analyze/simulate real-time scenarios [18, 19, 20]. The challenging contribution of this work is to demonstrate the possibility of efficiently (i.e. accurate and fast computing) solving optimization problems with electronic circuits (analog computing). The strong points of this approach based on the CNN paradigm is that CNN- templates are obtained through an identification process which is

mathematically defined as an optimization problem. This problem is solved by a training setup performing a *Nonlinear Adaptive Optimization (NAOP).* CNN-templates are generated upon convergence of the training process.

3   Design of the Complete Architecture Based on the "NAOP" Concept for the Modeling of Experimental Time Series with Ordinary Differential Equations

This section describes the approach based on the Nonlinear Adaptive Optimization (NAOP) for the modeling of experimental traffic time series data. The overall flow diagram of this approach is schematically displayed by the synoptic representation in figure 2a. The NAOP is performed by a complex 'computing' "module/entity/procedure" which does work on two inputs. The first input contains wave-solutions of the models describing the dynamics of a state-control CNN-network (i.e. CNN processor) modeled by Eq. (1). The second input contains experimental time series data. These data are represented in a state space (e.g. phase portrait) from which the corresponding component of the vector flow are depicted. The output of the NAOP system will generate, after extensive iterative computations or 'training' steps, appropriate CNN-templates upon convergence of the training process. Using these CNN-templates the appropriate parameter settings of the ODEs to model the experimental time series data are obtained. It should be worth a mentioning that figure 2a represents an elementary block (i.e. CPU-NAOP) of the complete architecture design to model experimental time series data with ODEs and/or solving ODEs with the CNN- paradigm. The complete architecture (see figure 2b) includes some key steps- say for instance, the derivation of the objective function with related constraints, the application of the concept of Neuron dynamics, and the design of an analog computing platform.

The overall flow diagram of the complete architecture for time series data modeling by ODEs is displayed by the synoptic representation in figure 2b. The global process to model experimental time series data can be summarized as follows. Time series data are represented in the *state space "phase portrait"* in terms of the components of the vector flow $\vec{\Phi}$ defined as follows:

$$\vec{\Phi}\left(s, \frac{ds}{dt}, \frac{d^2 s}{dt^2}, \cdot\frac{d^3 s}{dt^3}, \ldots\ldots\ldots\ldots \frac{d^i s}{dt^i} \right) \tag{2}$$

Where i is an integer, s are coordinates representing the state of the experimental time series, and $\dfrac{d^i s}{dt^i}$ represents the ith – derivative of s. Similarly the vector flow $\vec{\Theta}$ of the ordinary differential equation to be derived as corresponding mathematical model of experimental time series can be defined as follows:

$$\vec{\Theta}\left(x, \frac{dx}{dt}, \frac{d^2x}{dt^2}, . \frac{d^3x}{dt^3}, ...\frac{d^ix}{dt^i} \right) \tag{3}$$

Where x represents the temporal evolution of the state variable of the mathematical ODE under derivation as model of the experimental time series, and $\frac{d^ix}{dt^i}$ represents the $i^{th}-$ derivative of x.
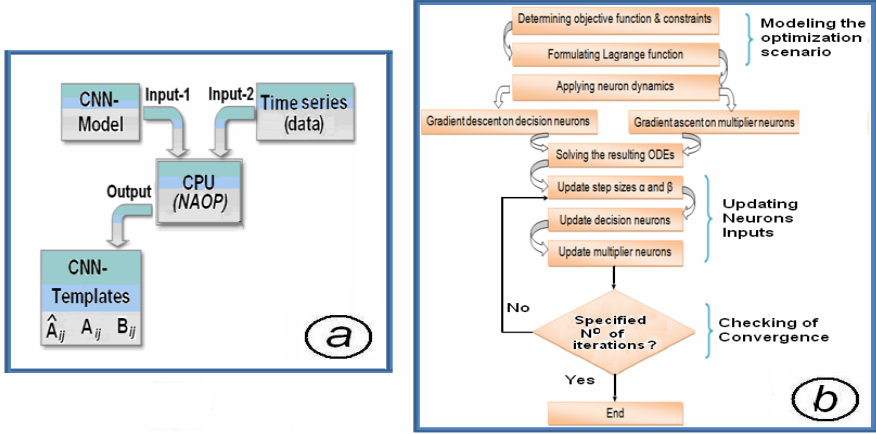
Fig. 2 (a) Synoptic representation of the architecture for CNN- templates calculations and time series data modeling using the NAOP approach. (b) Synoptic representation of the key steps involved in the complete training/learning process leading to time series modeling and forecasting.

The objective function is derived based on the statement/condition that a good mapping must be achieved between the mathematical ODE model and the experimental time series data. Equivalently to this statement, the two vectors flow $\vec{\Phi}$ and $\vec{\Theta}$ do evolve, at long term, on a common trajectory in the n-D phase space representation. Therefore, the objective function can be formulated/expressed in the following mathematical form:

$$Min\left[\sum_{i=0}^{n}\left(\frac{d^ix}{dt^i} - \frac{d^is}{dt^i} \right)^2 \right] \tag{4}$$

n is an unknown integer corresponding to the order of the "*appropriate*" ODE derived for the modeling of experimental time series data. The next step of the modeling process is concerned with the formulation of the related constraints.

Constraints are defined to make sure the statements above are fulfilled at long term. These constraints can be formulated mathematically as follows:

$$\frac{d^i x}{dt^i} = \frac{d^i s}{dt^i} \quad \text{(for all } i\text{)} \tag{5}$$

The Lagrange function $L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right)$ is obtained by combining the objective function with the related constraints. This function is formulated mathematically as follows:

$$L = \sum_{i=0}^{n} \left(\frac{d^i x}{dt^i} - \frac{d^i s}{dt^i}\right)^2 + \sum_{i=0}^{n} \lambda_i \left(\frac{d^i x}{dt^i} - \frac{d^i s}{dt^i}\right) \tag{6}$$

$x^{(i)}$ represents the ith – derivative of x, and γ_i are coefficients of the "appropriate ODE" to model experimental time series. These coefficients are determined through the identification process based on the training/learning method. Therefore, the problem related to the modeling of experimental time series can be transformed unto a new problem, that is an optimization problem, the objectives being the minimization of the Lagrange $L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right)$. To address this new problem, the concept of *"Neuron dynamics"* is exploited to demonstrate the possibility of solving this optimization problem using electronic circuits (i.e. analog computing). It is well-known that performing optimization with the aid of analog computing leads to more accurate and ultra-fast computing while compared the results obtained with those of the classical approaches based on Von Neumann computing architecture [18, 19, 20]. As it appears in Eq. (6), the Lagrange function $L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right)$ is expressed in terms of some key variables called *"metrics"*. These *"metrics"* represent *"Decision-neurons"* $\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}\right)$ and *"Multiplier- neurons"* $\left(\lambda_i, \gamma_i\right)$. The concept of "Neuron Dynamics" exploits two gradient principles [24]. The first is gradient- descent and the second is gradient- accent. Gradient descent is applied to decision neurons while gradient accent is applied to multiplier- neurons. Applying the gradient principles to the Lagrange function $L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i\right)$ leads to the following set of first- order coupled mathematical equations:

$$\frac{dx^{(i)}}{dt} = -\alpha \frac{\partial L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right)}{\partial x^{(i)}} \tag{7a}$$

$$\frac{ds^{(i)}}{dt} = -\alpha \frac{\partial L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right)}{\partial s^{(i)}} \tag{7b}$$

$$\frac{d\lambda_i}{dt} = +\beta \frac{\partial L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right)}{\partial \lambda_i} \tag{7c}$$

$$\frac{d\gamma_i}{dt} = +\beta \frac{\partial L\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right)}{\partial \gamma_i} \tag{7d}$$

The knowledge of the Lagrange function is very important as it helps to transform Eqs. (7) unto a set of coupled first order ordinary differential equations as expressed in Eqs. (8).

$$\frac{dx^{(i)}}{dt} = F\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right) \tag{8a}$$

$$\frac{ds^{(i)}}{dt} = G\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right) \tag{8b}$$

$$\frac{d\lambda_i}{dt} = H\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right) \tag{8c}$$

$$\frac{d\gamma_i}{dt} = K\left(x, x^{(1)}, ...x^{(i)}, s, s^{(1)}, ...s^{(i)}, \lambda_i, \gamma_i\right) \tag{8d}$$

Equations (8) are sets of coupled linear/nonlinear ODE of the first order. Applying the circuit theory in electrical engineering, it is possible to design an implement an electronic setup to solve Eqs. (8).

4 Modeling Time Series Data of the Van der Pol Oscillator with Cellular Neural Networks (CNN)

Section 2 has revealed that the CNN- processor can be modeled by the coupled nonlinear ODEs described in Eq. (1). This model is used in the optimization process and, its corresponding coefficients are depicted after convergence. These coefficients which are called CNN- templates correspond to the appropriate coefficients to model time series data obtained in the phase space representation of the van der Pol oscillator. Figure 3a shows an illustration of experimental chaotic traffic time series data predicted from a real- traffic scenario/dynamics in Beijing Xizhimen. Figures 3b and 3c show the time series data obtained in both the state space representation (Fig. 3b) and the phase space representation (Fig. 3c) of the van der Pol oscillator. The state space reveals the temporal dynamics of the oscillator while the phase space is a 2D representation of the coordinates x versus the coordinates $\dot{x}$ (Fig. 3c). As it appears in figure 3c, the transient phase can be

depicted along the open loop- trajectory located inside the closed loop- trajectory which is the well-known limit cycle which reveals the permanent state of the oscillator.

We now want to model the time series data in figures 3 by a set of coupled non-linear ordinary differential equations. As it has been extensively discussed in the previous sections, the mathematical model expected is that describing the dynamics of the CNN- processor). Due to the fact that the phase space representation of the time series data is performed in a 2D space, the index i can take only 2 values (i.e. $i \in \{0, 1\}$), leading to the following coordinates of the time series data $\left(s^{(0)}, s^{(1)}\right)$. These coordinates can be conveniently represented into the equivalent forms $\left(s, s^{(1)}\right)$ or $\left(s, \dfrac{ds}{dt}\right)$ without loose of generality.
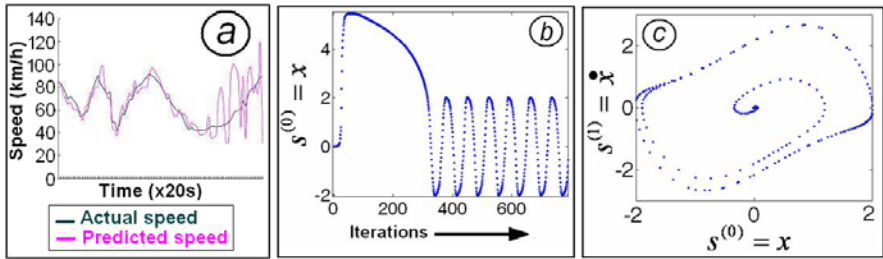
Fig. 3 (a) General illustration of a chaotic traffic dynamics for the ***Beijing Xizhimen*** data. The chaotic evolution of the average vehicle-speed time series data is depicted. This plot is obtained from Ref. [1]. (b) Time series data of the state- space representation of the van der Pol oscillator and (c) Time series data of the phase - space representation of the van der Pol oscillator. Both transient & permanent states are depicted.

The learning/training process is based on a mapping between the two inputs of the NAOP procedure. A convergence to global minima is the key purpose governing this template calculation process, the so-called *NAOP*. A large number of randomly generated attractors are being chosen in the time series data. An attempt to map these attractors to those generated by the model under investigation (i.e. the CNN- model) is performed in a sequential process leading to the convergence to a global minimum when the mapping is achieved successfully. This convergence must be achieved in both the 'CNN-templates' and the 'attractors' $\left(s, \dfrac{ds}{dt}\right)$ depicted from the time series data. $\left(s, \dfrac{ds}{dt}\right)$ are all considered to be dynamic

variables during the learning/training process. It is further worth a mentioning that the quintessence of the concept *NAOP* is in the core an adaptive training process that is very comparable to the concept developed for the training of Hopfield neural networks towards an efficient tracking of global and/or local minima [8]-[9].

Using the time series data in the phase space representation of the van der Pol oscillator, the *NAOP* concept has been exploited to optimize the corresponding CNN- templates after convergence of the training process. This convergence is clearly illustrated by the plots presented in Figs. (4a) and (4b) showing the temporal evolution of both the state-control templates $\hat{A}_{ij}$ (see Fig. (4a)) and the feedback templates A_{ij} (see Fig. (4b)). As it appears in figures 4, the convergence is achieved after a transient phase displayed by the global training network. It is worth a mentioning that the convergence of the process is achieved for suitable basins of attractions. From figures 4, one can easily read the following corresponding CNN templates that are then used to solve the van der Pol equation:

$\hat{A}_{11} = 1.0770$, $\hat{A}_{12} = -0.6300$, $\hat{A}_{21} = 1.3450$, $\hat{A}_{22} = 0.5850$, $A_{11} = 0.4473$,

$A_{12} = -0.2586$, $A_{21} = 0.4846$, $A_{22} = 0.1310$.
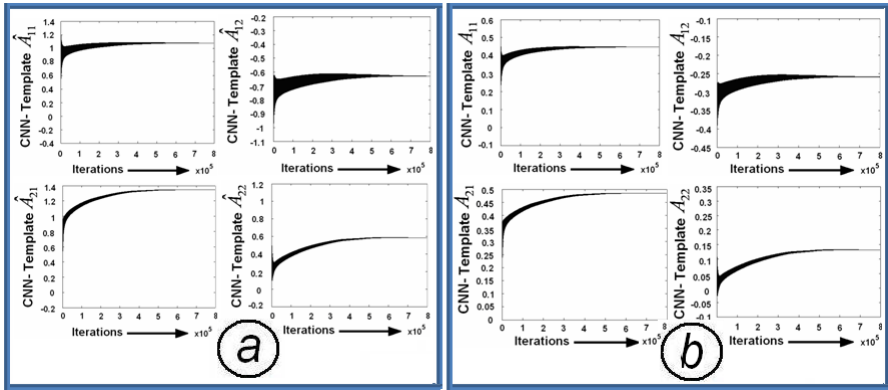
Fig. 4 (a) Convergence of state-control CNN templates as achieved by the NOAP process using the time series data in figure 5. (b) Convergence of Feedback- templates achieved by the NOAP process using the time series data in figure 5

The set of template values above has been inserted in Fig. 5a to obtain the phase space representation in Fig. 5b using the CNN paradigm. It clearly appears that this representation (Fig. 5b) is similar to the representation of the experimental time series data in figure 3c. Interestingly, the transient phase has been removed or suppressed during the modeling process based on the NAOP concept.
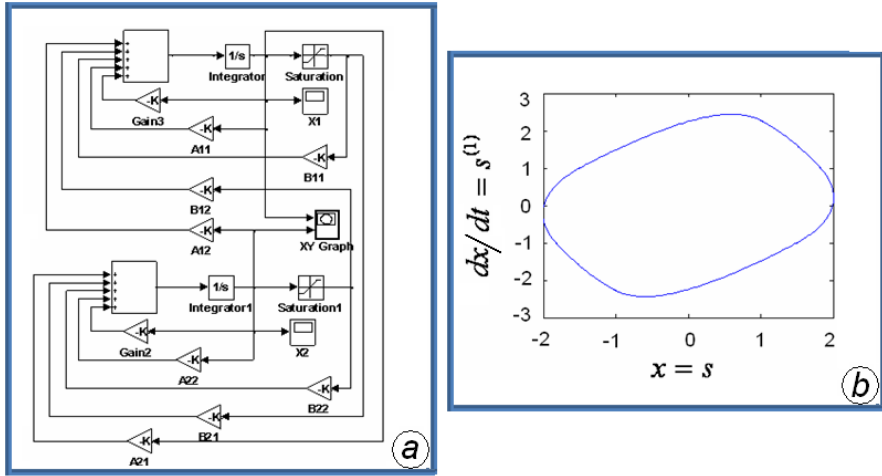
Fig. 5 SIMULINK graphical representation of the CNN- computing platform to solve (3). (b) Result of equation (3): Phase space representation plotted using the equations of the CNN- processor set of coupled ODEs which have been derived to model the time series data in figure 5.

5 Concluding Remarks

We have proposed and validated a concept based on the CNN paradigm for the modeling of time series data. The method proposed in this paper is challenging as it demonstrates a systematic and straightforward way to model time series data by the CNN- paradigm. The key challenge has been the development of appropriate algorithms for mapping nonlinearity time series data unto the nonlinearity displayed by the elementary CNN- cell.

In essence, it has been demonstrated that the CNN- paradigm is an appropriate concept for the modeling of time series data. The issue of modeling time series data can be transformed into an optimization problem, the objectives being the identification of the corresponding CNN- templates.

Another strong point of the CNN-paradigm is the resulting ultra-fast processing depending on the CNN implementation: DSP, FPGA, GPU, or CNN-Chip. One key objective of this work has been to advance the relevant state-of-the-art by proposing a novel framework to modeling time series data with the CNN- paradigm. To achieve this goal, we have proposed and demonstrated that the Nonlinear Adaptive Optimization (NAOP) technique is a best and efficient scheme to cope with the modeling of various types of time series data. The NAOP is a learning/training method for mapping the wave solutions of the models describing the dynamics of a CNN-network to the phase space representation of time series data.

In order to overcome the problem related to the speed of computation, an implementation either on FPGA or DSP or GPU of the concept developed in this work is possible and straight-forward.

An interesting question to be addressed is the extension of the approach developed in this work to the modeling of noisy time series data. This is a research question of necessary importance as it is well-known that experimental data obtained from sensors are generally imperfect data due to some uncontrollable phenomena. Therefore a pre-processing is necessary for the sensor data filtering process.

References

[1] Shang, P., Li, X., Kamae, S.: Chaotic Analysis of Traffic Time Series. J. Chaos, Solitons and Fractals 25, 121–128 (2005)

[2] Medeiros, C.B., Joliveau, M., Jomier, G., De Vuyst, F.: Managing sensor traffic data and forecasting unusual behaviour propagation. Geoinformatica (2010)

[3] Peeta, S., Anastassopoulos, I.: Automatic Real-Time Detection and Correction of Erroneous Detector Data with Fourier Transforms for Online Traffic Control Architectures. Transportation Research Record, N 02-2244 1811, 1–11 (2002)

[4] Casdagli, M., des Jardins, D., Eubank, S., Farmer, J.D., Gibson, J.: Nonlinear Modeling of Chaotic Time Series: Theory and Applications. J. Applied Chaos, 335–380 (1992)

[5] Abarbanel, D.I.H., Brown, R., Kadtke, J.B.: Prediction in Chaotic Nonlinear Systems: Methods for Time Series with Broadband Fourier Spectra. Phys. Rev. A 41, 1783–1807 (1990)

[6] Gennemark, P., Wedelin, D.: Benchmarks for Identification of Ordinary Differential Equations from Time Series Data. J. Bioinformatics 25, 780–786 (2009)

[7] Eisenhammer, T., Hübler, A., Packard, N., Kelso, J.A.S.: Modeling Experimental Time Series with Ordinary Differential Equations. Biological Cybernetics 65, 107–112 (1991)

[8] McSharry, P.E., Smith, L.A.: Just di it? Reductionism, Modelling and Black-box Forecasting. In: International Workshop on Advanced Black-Box Techniques for Nonlinear Modeling, pp. 106–111. Kluwer Academic Publishers, Belgium (1999)

[9] Daw, C.S., Kennel, M.B., Finney, C.E.A., Connolly, F.T.: Observing and Modeling Nonlinear Dynamics in an internal Engine. Phys. Rev. E 57, 2811–2819 (1998)

[10] Smith, L.A.: Disentangling uncertainty and error: on the predictability of nonlinear systems. J. Nonlinear Dynamics and Statistics, 31–64 (2001)

[11] Beek, P.J., Schmidt, R.C., Morris, A.W., Sim, M.-Y., Turvey, M.T.: Linear and Nonlinear Stiffness and Friction in Biological Rhythmic Movements. Biological Cybernetics 73, 499–507 (1995)

[12] Liu, Z.: A Survey of Intelligence Methods in Urban Traffic Control. International Journal of Computer Science and Network Security (IJCSNS) 7, 105–112 (2007)

[13] Shvetsov, V.I.: Mathematical Modeling of Traffic Flows. J. Automation and Remote Control 64, 1651–1689 (2003)

[14] Gültekin Cetiner, B., Sari, M., Borat, O.: A Neural Network Based Traffic-Flow Prediction Model. Mathematical and Computational Applications 15, 269–278 (2010)

[15] Amin, S.M., Rodin, E.Y., Liu, A.-P., Rink, K., Garcia-Ortiz, A.: Traffic Prediction and Management via RBF Neural Nets and Semantic Control. Computer- Aided Civil and Infrastructure Engineering 13, 315–327 (1998)

[16] van Mourik, A.M., Daffertshofer, A., Beek, P.J.: Deterministic and stochastic features of Rhythmic Human Movement. Biological Cybernetics 94, 233–244 (2006)

[17] Lorenz, E.N.: Predictability- A Problem Partly Solved. In: Predictability EXMWF, Seminar Proceedings, Shinfield Park, Reading (1995)

[18] Chua, L.O., Yang, L.: Cellular Neural Networks: Theory. IEEE Transactions on Circuits and Systems 35(10) (October 1988)

[19] Chedjou, J.C., Kyamakya, K., Latif, M.L., Khan, U.A.: Solving Stiff Ordinary Differential Equations and Partial Differential Equations Using Anolog Computing Based on Cellular Neural Networks. ISAST Transactions on Computers and Intelligent Systems 2, 8–14 (2010)

[20] Manganaro, G., Arena, P., Fortuna, L.: Cellular Neural Networks: Chaos, Complexity, and VLSI Processing. Advanced Microelectronics, 269 (1999)

[21] Joliveau, M.: Reduction of Urban Traffic Time Series from Georeferenced sensors, and extractionof spatio-temporal series-in French. PhD thesis, Ecole Centrale Des Arts Et Manufactures, Ecole Centrale de Paris, France (2008)

[22] Stough, R., Yang, G.: Intelligent Transportation Systems. In: Encyclopedia of life support systems (EOLSS). Developed under the Auspices of the UNESCO, Oxford, UK (2003)

[23] Judd, K., Smith, L.A.: Towards Forecasting Bounding Bloxes: Applications to both Waether and Climate. J. Atmos. Sci. 32 (2000)

[24] Hopfield, J.J., Tank, D.W.: Neural computation of decisions in optimization problems. Biological Cybernetics (52), 141–152 (1985)

CNN Based High Performance Computing for Real Time Image Processing on GPU

Sasanka Potluri, Alireza Fasih, Laxminand Kishore Vutukuru, Fadi Al Machot, and Kyandoghere Kyamakya

Abstract. Many of the basic image processing tasks suffer from processing overhead to operate over the whole image. In real time applications the processing time is considered as a big obstacle for its implementations. A High Performance Computing (HPC) platform is necessary in order to solve this problem. The usage of hardware accelerator make the processing time low. In recent developments, the Graphics Processing Unit (GPU) is being used in many applications. Along with the hardware accelerator a proper choice of the computing algorithm makes it an added advantage for fast processing of images. The Cellular Neural Network (CNN) is a large-scale nonlinear analog circuit able to process signals in real time [1]. In this paper, we develop a new design in evaluation of image processing algorithms on the massively parallel GPUs with CNN implementation using Open Computing Language (OpenCL) programming model. This implementation uses the Discrete Time CNN (DT-CNN) model which is derived from originally proposed CNN model. The inherent massive parallelism of CNN along with GPUs makes it an advantage for high performance computing platform [2]. The advantage of OpenCL makes the design to be portable on all the available graphics processing devices and multi core processors. Performance evaluation is done in terms of execution time with both device (i.e. GPU) and host (i.e. CPU).

1 Introduction

Image processing is an ever expanding and dynamic area with applications reaching out into everyday life such as in medicine, space exploration, surveillance, authentication, automated industry inspection and in many more areas [3]. Real time image

Sasanka Potluri · Alireza Fasih · Laxminand Kishore Vutukuru · Fadi Al Machot · Kyandoghere Kyamakya
Alpen-Adria-University Klagenfurt, Institute of Smart System Technologies,
Transportation Informatics Group, 9020 Klagenfurt, Universitätsstrasse 65-67, Austria
e-mail: {spotluri,lvutukur}@edu.uni-klu.ac.at, {Alireza.Fasih,
       Fadi.Almachot,Kyandoghere.Kyamakya}@uni-klu.ac.at

processing using modern processors is limited [4]. Problems in computer vision are computationally intensive [5]. The tremendous amount of data required for image processing and computer vision applications present a significant problem for conventional microprocessors [4]. Consider a sequence of images at medium resolution (512 x 512 pixels) and standard frame rate (30 frames per second) in color (3 bytes per pixel). This represents a rate of almost 24 million bytes of data per second. A simple feature extraction algorithm may require thousands of basic operations per pixels, and a typical vision system requires significantly more complex computations. As we can see, parallel computing is essential to solve such problems [5]. In fact, the need to speed up image processing computations brought parallel processing into computer vision domain. Most image processing algorithms are inherently parallel because they involve similar computations for all pixels in an image except in some special cases [5]. Conventional general-purpose machines cannot manage the distinctive I/O requirements of most image processing tasks; neither do they take advantage of the opportunity for parallel computation present in many vision related applications [6]. Many research efforts have shifted to Commercial-Off-The-Shelf (COTS) -based platforms in recent years, such as Symmetric Multiprocessors (SMP) or clusters of PCs. However, these approaches do not often deliver the highest level of performance due to many inherent disadvantages of the underlying sequential platforms and the divergence problem. The recent advent of multi-million gate on the Field Programmable Gate Array (FPGAs) having richer embedded feature sets, such as plenty on chip memory, DSP blocks and embedded hardware microprocessor IP cores, facilitates high performance, low power consumption and high density [7]. But, the development of dedicated processor is usually expensive and their limited availability restricts their widespread use and its complexity of design and implementation also makes the FPGA not preferable. However, in the last few years, the graphic cards with impressive performance are being introduced into the market for lower cost and flexibility of design makes it a better choice. Even though they have been initially released for the purpose of gaming, they also find the scientific applications where there is a great requirement of parallel processing. Along with the support of hardware platforms there are some software platforms available like CUDA (Compute Unified Device Architecture) and OpenCL for designing and developing parallel programs on GPU [8]. Out of these available software platforms OpenCL framework recently developed for writing programs can be executed across multicore heterogeneous platforms. For instance, it can be executed on multicore CPUs and GPUs and their combination. Usage of this framework also provides an advantage of the portability that is; the developed kernel is compatible with other devices. Along with the available hardware and software platforms we used the CNN parallel computing paradigm for some image processing applications. The idea of CNN was taken from the architecture of artificial neural networks and cellular automata. In contrast to ordinary neural networks, CNN has the property of local connectivity. The weights of the cells are established by the parameters called the template. The functionality of the CNN is dependent on the template. So with a single common computing model, by calculating the templates we can achieve the desired functionality. The CNN has been successfully used for various high-speed

parallel signal processing applications such as image processing, visual computing and pattern recognition as well as computer vision [9]. So we thought of implementing it on the hardware for the need of HPC in real time image processing. Also, the parallel processing capability of the CNN makes us to implement the CNN architecture on the hardware platform for its efficient visualization. In this paper, the effort is done to develop a DT-CNN model on the graphics processing units with the OpenCL framework. An effort is done to make the development of DTCNN entirely on the kernel which make it executable on every platform. But, it should be noticed that the GPU is a coprocessor which supports the processor in our system. Hence, the CPU still executes several tasks, like the transmission of the data to the local memory of the graphics card and retrieving back. Finally, GPU-based Universal Machine - CNN (UM-CNN) was implemented using the OpenCL framework on NVIDIA GPU. A benchmark is provided with the usage of GPU based CNN model for the image processing in comparison with CPU. The paper is structured as follows: Section 2 gives a clear description about the theory involved in parallel computing. Section 3 introduces the concepts of CNN, the system diagram and its functionality and systems designed methodology which is done using OpenCL. Section 4 concludes the paper and says about the work going to be done in full paper.

2   Theory of Parallel Computing

Traditionally, computer software has been written for the serial computation and time sharing computation. Therefore to solve a problem, an algorithm is constructed which produces a serial stream of instructions. These produced instructions are executed sequentially one after the other on the CPU of the computer. Parallel computing on the other hand uses multiple processing elements simultaneously on a problem. The problem is broken into parts which are independent so that each processing element can execute its part of the algorithm simultaneously with others. The processing elements can be diverse and include resources such as a single computer with multiple processors, a number of networked computers, specialized hardware or any combination of any of the above.

The software speedup was achieved by using a CPU with higher clock rates, which significantly increased each passing year. However, when clock speed reached 4GHz, the increase in power consumption and heat dissipation formed what is known as the Power Wall which effectively caused the CPU clock rate to level off [10]. Along with this, many applications today require more computing power than a traditional sequential computer can offer. All these things made the vendors search for an alternative to make increase of available cores with in a processor instead of increasing the clock rates. The increase of cores on the processor made the CPU clock rates to remain same or even reduced to economize the power usage. The old software design used for the sequential process will not get increased directly with the increase in the processor cores. To get the benefit form the current developed multi core processor, new software has to be designed to take the advantages

of the new architecture. This makes the use of all available multi cores and performs process in parallel. Parallel computing is defined as a form of computation in which many calculations are carried out simultaneously, operating on the principle that large problems can often be divided into smaller ones, which are then solved concurrently (in parallel)". Many different hardware architectures exist today to perform a single task using multiple processors. Some examples are: Grid computing a combination of computer resources from multiple administrative domains applied to a common task. Massively Parallel Processor (MPP) systems known as the supercomputer architecture Cluster server system a network of general-purpose computers. Symmetric Multiprocessing (SMP) system identical processors (in powers of 2) connected together to act as one unit. Multi-core processor a single chip with numerous computing cores [10]. Heterogeneous computing systems also provide an opportunity to dramatically increase the performance of parallel and HPC applications on clusters with CPU and GPU architecture [11]. This concept can be achieved by combining the GPU and multicore CPUs.

3 System Design and Architecture

This section explains in detail about the CNN, its architecture and advantages. It is followed by the description of the system we have developed and about the OpenCL framework we have used and its advantages for programming on GPU.

3.1 Cellular Neural Networks

Analog circuit have played a very important role in the development of modern electronic technology. Even in our digital computer era, analog circuits still dominate such fields as communications, power, automatic control, audio and video electronics because of their real-time signal processing capabilities [12]. CNN technology is both a revolutionary concept and an experimentally proven new computing paradigm. Analogic cellular computers based on CNNs are set to change the way analog signals are processed and are paving the way to an entire new analog computing industry [13]. CNN was proposed by Chua and Yang in 1988 [12]. The CNN is defined as a n-dimensional array of cells that satisfies two properties: (i) most interactions are local within a finite radius r, and (ii) all state variables are continuous valued signals [14].The CNN has M by N processing unit circuit called cells C (i, j) located at site $(i,j), i = 1,2,...,M, j = 1,2,...,N$[15]. The array of CNN cell structure is as shown in Fig. 1.

Each cell of the CNN is made of a linear capacitor, a nonlinear voltage controlled current source and a few resistive linear circuit elements. The dynamic equation of a cell C (i, j) in an M x N CNN, given by CHUA and Yang [12] is shown below:

$$C\frac{dX_{ij}}{dt} = \frac{1}{R}X_{ij} + \sum_{c(k,l)\in N_r(i,j)} (A_{ij,kl}Y_{kl} + B_{ij,kl}U_{kl}) + I \qquad (1)$$

Where the output equation Y_{ij} can be written as:

$$Y_{ij} = f(X(i,j)) = f\frac{1}{2}(|X+1| - |X-1|) \qquad (2)$$

The mathematical equation mentioned in equation (1) is representing the model of the Continuous Time CNN (CT-CNN). In the equation, C is a linear capacitor and R is a resistor. Ykl is the output state of each cell. Ukl is the input of each cell. Aij and Bij are the template elements. Xij represents the initial state and I represents the threshold or bias for each cell. The equation (2) is the output equation of each iteration. This equation gives the functional model for the calculation of each pixel element to the output. This model in not very fast in the real time image processing. In order to overcome the drawbacks of CT-CNN, the concept of Discrete Time CNN (DT-CNN) is developed. The DT-CNN is defined by the difference equations instead of differential equations used in the CNN [16]. The model of DT-CNN is derived from the model of CT-CNN using the Eulers method. The DT-CNN can be described with the following equation [16].

$$X_{i,j}(t+1) = \sum_{c(k,l) \in N_r(i,j)} (A_{ij,k,l} f(X_{k,l(t)}) + \sum_{c(k,l) \in N_r(i,j)} B(i,j;k,l)U_{k,l} + I \qquad (3)$$

From equation (3), we can see that $X_{i,j}$ is the state of the cell $C(i,j)$ and $f(X_k,l)$ is the output of cell $C(k,l)$ within the neighborhood Nr(i, j) of C(i, j). Uk,l is the input of each cell $C(k,l)$ within $Nr(i,j)$, and I is the bias of cell. A and B are called the feed-back and feed-forward templates of the CNN respectively. CNNs are widely used for real time image processing applications. Though the CNN, as a concept is characterized by a strict locality operation, the large scale digital implementation has been far from trivial [17].
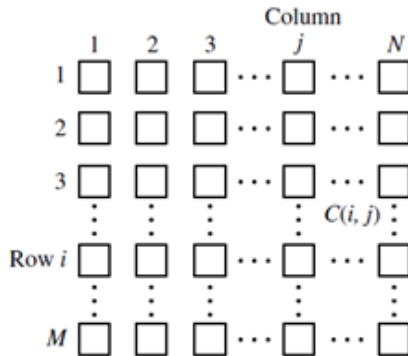
Fig. 1 A simple CNN array architecture [13]

3.2   System Diagram

The system diagram gives the clear understanding of the work we have done along with the flow of data and processing steps. The system diagram of the designed model of our research was show in Fig. 2. From the system diagram shown above we can understand that there are several available platforms which can help in the processing of image processing tasks. They are FPGAs, DSPs, CPUs and GPUs. Out of which the OpenCL framework is capable of developing these image processing algorithms on the multi core CPUs and on GPU or on cluster of GPUs. So we have chosen this OpenCL framework as a programming language for our task.

The Image processing unit has access to the global memory as a temporary buffer for the processing of images. There is a high level script interpreter for task management and accessing to I/Os and digital channels such as files, Camera, etc. The data which is to be processed is kept in the shared memory and is accessible by the OpenCL framework and the Image processing unit. After loading the instructions from the image processing unit, the API of the OpenCL will try to take the required kernel from the kernel bank, for the process of the instructions. After getting the required kernels from the kernel bank the OpenCL make these kernels to be executed on the necessary processor and then the results are again stored in the shared memory which is collected by the image processing unit. The OpenCL API also checks the processing commands and the synchronization commands from the image processing unit in order to perform a correct operation on the proper image.
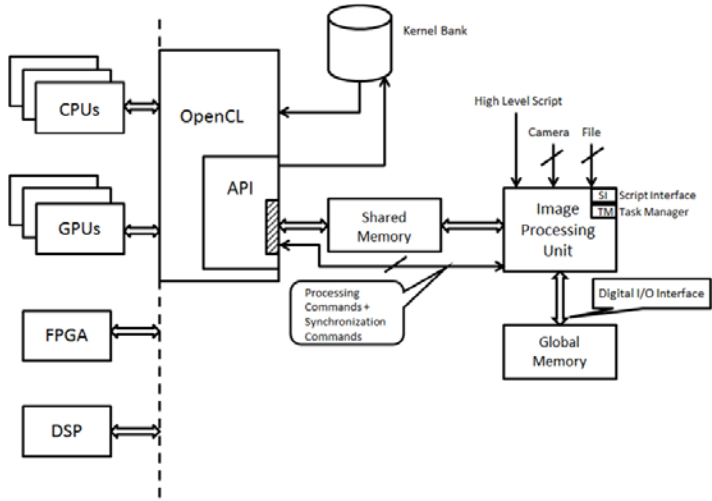
Fig. 2 System Design Architecture

3.3   Methodology of System Design Using OpenCL

OpenCL is a framework suited for parallel programming of heterogeneous systems. The framework includes the OpenCL C language as well as the compiler and the runtime environment required to run the code written in OpenCL C. OpenCL is standardized by the Khronos Group, who is known for their management of the OpenGL specification [10]. OpenCL is a framework which has been developed in order to program on any heterogeneous environment. Therefore we can say that the OpenCL code is capable of being executed on any computer. OpenCL provides standardized APIs that perform tasks such as vectorized SIMD (Single Instruction Multiple Data) operations, data parallel processing, task parallel processing and memory transfer. A simple functional execution block written in OpenCL is called as a kernel. The OpenCL kernel is written in its native language using its own API. The benefits of OpenCL are: it allows acceleration in parallel processing, it manages the computational resources, it views multi-core CPUs, GPUs etc. as computational units, allocates different levels of memory, it also supports cross-vendor software portability.
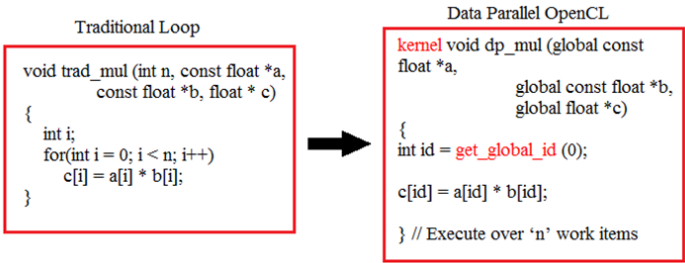
Fig. 3 A comparison traditional loop with the OpenCL data parallel kernel

A simple kernel written in OpenCL in comparison with the traditional loop coding is seen the following Fig. 3. The Fig. 3 gives the advantages of writing the code in OpenCL as well as the parallel computing capability of the OpenCL kernel. From Fig. 4 we can clearly understand the steps involved in the OpenCL programming model. The steps show the structural design and execution of a kernel. Whenever the kernel is designed, these steps are followed in order to execute the kernel on the appropriate device. Here in our work, we have used the OpenCL in order to develop the UM - CNN which is to be executed on the GPU. The required kernel which is necessary for the execution of the CNN is written using the OpenCL programming model. The interface of the data is done from the CPU which is to be loaded onto the device. For proper interface of the communication of data we have used the OpenCV (Open Computer Vision Library) for the reading of images, loading
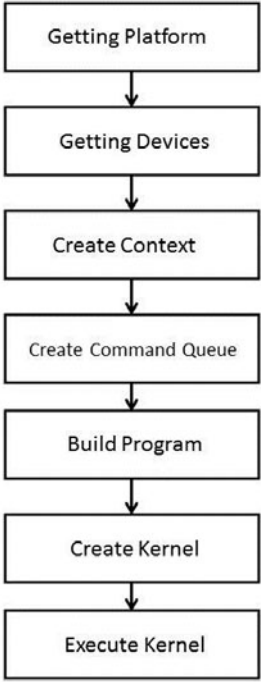
Fig. 4 OpenCL programming Flow

of the image data on to the memory elements and to display the retrieved image back from the GPU after its execution. In our work we have used the combination of OpenCV for the appropriate user interface and for data acquisition, and OpenCL for the development of the desired UM CNN on the heterogeneous platform.

3.4 Universal Model of CNN

As we have seen in section 3 about the mathematical description of the DT-CNN. The equation (3) can be also be written as:

$$X_{ij}(t+1) = T_A * Y_{kl}(n) + T_B * u_{kl} + I \tag{4}$$

The equation (4) gives an equivalent model of the DT-CNN which is given in equation (3). Form this equivalent model we can clearly understand the functionality of the DT-CNN. The model or the architecture developed is taken from equation (4). The architecture of designed CNN-UM model can be seen from Fig. 5. The CNN-UM model developed on the GPU with multiple kernels is as shown in Fig 5.

From figure we can see that there are in total three kernels involved in the development of the CNN-UM on GPU. The kernels that are necessary in order to implement the CNN-UM are the convolution kernel, the accumulation kernel and the sigmoid kernel. The execution of these kernels in a particular pattern will result in the functionality of the CNN-UM on GPU. The first convolution operation is performed in order to compute the TBU image that is the TB * Ukl part of the CNN equation. The value of TBU is constant throughout the process for a single input image.
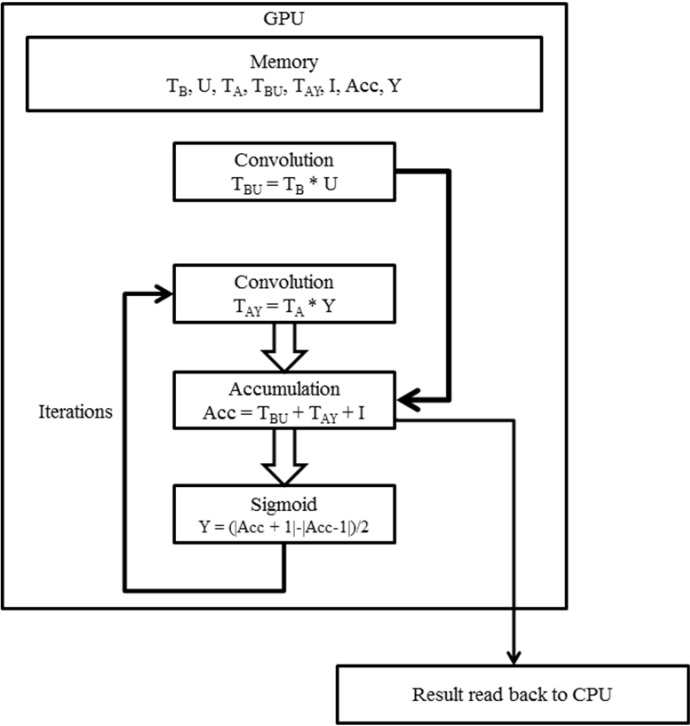
Fig. 5 Architecture of CNN-UM Model on GPU

Next, the convolution operations are performed in each iteration to calculate the TAY image that is $T_A * y_{kl}(n)$ part of the CNN equation. After computing the TBU and TAY the accumulation is performed. Along with TBU and TAY the bias I is also added together in the accumulation kernel. After the accumulation operation the sigmoid operation is performed which is the nonlinear function used. After the sigmoid operation is performed on the accumulation it is considered as the state of the single cell and again the entire process is repeated. Again the new TAY image is calculated. This process repeats until the n iterations.

4 Experimental Results

previous section we have seen the design procedure of the CNN-UM model on the GPU. In this section we see the results obtained and the benchmarks to evaluate the performance of the CNN-UM on GPU. The resources available to evaluate the results are Intel (R) Xeon (TM) CPU 3.60 GHz with 2GB RAM and 32-bit Windows operating system. The GPU used for this process is GeForce 9500 GT with 256MB memory. It is having 4 multiprocessors with 8 cores on each multiprocessor. In total it is having 32 CUDA cores which can run in parallel. The development environment is Visual Studio 2008.

4.1 Benchmarking

In Fig 6 we can see the comparision between the performance of the CPU and GPU with respect to the number of pixels and execution time. From Fig we can observe that when the number of pixels is low the level of performance offered by the GPU is nearly equalt to the performance level of the CPU. When the number of pixels to be executed increased the amount of parallelism obtained is high and hence the performace of the GPU increases.
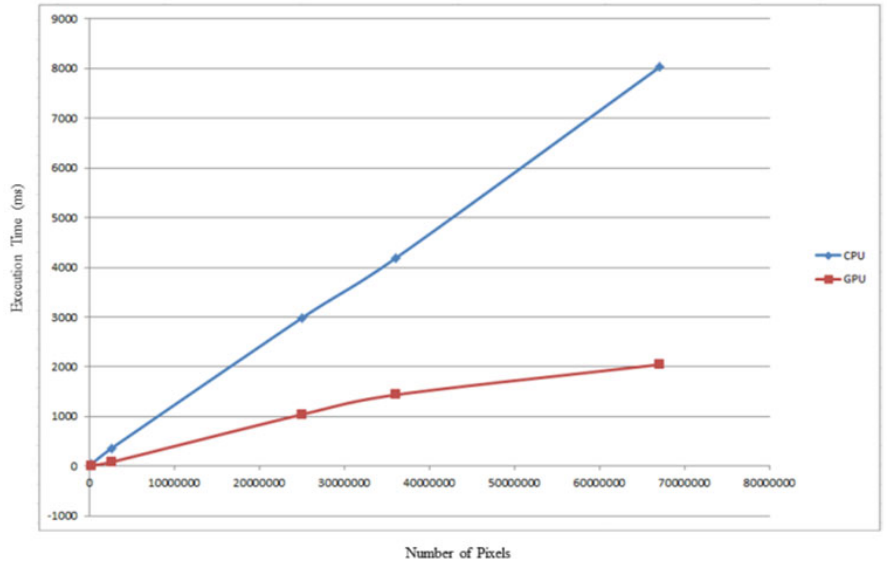
Fig. 6 A benchmark for CPU and GPU

4.2   Conclusion

GPUS are being widely used in many real time image processing applications where there is a need for huge amount of computation. These are playing a key role in developing the new platform of the high performance computing. The parallel computing capability of the CNN and the interaction between the cells makes it an advantage for the image processing applications. The OpenCL framework used for the development of this model is a good platform to develop the applications on heterogeneous platforms. Hence, we can say that the developed CNN-UM model is capable to execute on the CPU as well as GPU and is vendor independent that is the developed model is able to perform its operations on AMD, NVIDIA and other vendors with any modifications. From the performance evaluation we can see that the usage of GPU for image processing applications really accelerates the processing speed and reduces the execution time. In the near future we plan to expand this architecture as an universal parallel computing platform for several areas. This architecture can show way to the several time consuming applications in many fields of applications. The areas include signal processing framework, to the field of mathematics to find solutions for complex problems and also to the field of finite element analysis. Secondly, this model at present uses OpenCV for handling the I/O of the image and OpenCL for the execution on the GPU. We also want to add the OpenGL for high level GUI and also the features of OpenMP to make model more user friendly. We also plan to make this architecture an uniform model by modifying the architecture using Unified Modelling Language (UML). With this we can use this architecture as a tool for several applications and data abstraction is possible.

References

1. Montufar-Chaveznava, R., Guinea, D.: A Cellular Neural Network System for Real-Time Image Processing. In: MED 2003 - The 11th Mediterranean Conference on Control and Automation, 3rd edn. (2003)
2. Fernandez, A., et al.: Cellular neural networks simulation on a parallel graphics processing unit, 3rd edn., pp. 208–212 (2008)
3. Kanwar, R.: Real-Time Edge Detection using Sundance Video and Image Processing System, 3rd edn. (2009)
4. Greco, J.: Parallel Image Processing and Computer Vision Architecture, 3rd edn., Citeseer (2005)
5. Choudhary, A., Ranka, S.: Guest editor's introduction: parallel processing for computer vision and image understanding. In: Computer, vol. 25(2), pp. 7–10 (1992)
6. Athanas, P.M., Abbott, A.L.: Real-time image processing on a custom computing platform. In: EComputer, 3rd edn., vol. 28(2), pp. 16–25 (2002)
7. Wang, X., Ziavras, S.G.: KHERA: A reconfigurable and mixed-mode parallel computing engine on platform FPGAs. Citeseer (2004)
8. Tonde, C.: Hardware and software platforms for computer vision, 3rd edn.
9. Yang, Z., Nishio, Y., Ushida, A.: A Two Layer CNN in Image Processing Applications
10. Tsuchiyama, R., et al.: The OpenCL Programming Book. Fixstars Corporation
11. Barak, A., et al.: A Package for OpenCL Based Heterogeneous Computing on Clusters with Many GPU Devices

12. Nossek, J.A., et al.: Cellular neural networks: Theory and circuit design. International Journal of Circuit Theory and Applications 20(5), 533–553 (1992)
13. Chua, L.O., Roska, T.: Cellular neural networks and visual computing: foundation and applications. Cambridge Univ. Pr. (2002)
14. Chua, L.O., Roska, T.: The CNN paradigm. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 40(3), 147–156 (2002)
15. Kawahara, M., Inoue, T., Nishio, Y.: Image processing application using CNN with dynamic template. IEEE
16. Takenouchi, H., Watanabe, T., Hideki, A.: Development of DT-CNN Emulator Based on GPGPU
17. Malki, S., Spaanenburg, L., Ray, N.: Image stream processing on a packet-switched discrete-time CNN (2004)

A Novel Real-Time Emotion Detection System for Advanced Driver Assistance Systems

Fadi Al Machot, Ahmad Haj Mosa, Alireza Fasih, Christopher Schwarzlmüller, Mouhanndad Ali, and Kyandoghere Kyamakya

Abstract. This paper presents a real-time emotion recognition concept of voice streams. A comprehensive solution based on Bayesian Quadratic Discriminate Classifier(QDC) is developed. The developed system supports Advanced Driver Assistance Systems (ADAS) to detect the mood of the driver based on the fact that aggressive behavior on road leads to traffic accidents. We use only 12 features to classify between 5 different classes of emotions. We illustrate that the extracted emotion features are highly overlapped and how each emotion class is effecting the recognition ratio. Finally, we show that the Bayesian Quadratic Discriminate Classifier is an appropriate solution for emotion detection systems, where a real-time detection is deeply needed with a low number of features.

1 Introduction

Every minute, on average, at least one person dies in vehicle crash [9]. For this reason, different approaches to increase the safety on road have been developed. These methods are generally called Advanced Driver Assistance Systems (ADAS) and supports the driver in its driving process. Due to the fact, that every reaction on road is time crucial, ADAS must propose real-time processing. This is the major requirement for such systems.

Typical representatives of ADAS are e.g. Adaptive Cruise Control (ACC) and Lane Departure Warning Systems (LDWS) which are systems to increase safety. Both systems can intervene in the driving process, to avoid hazardous situations for the driver. Other ADAS systems monitor drivers fatigue by analyzing its facial features (eyes), the inclination of its head or the characteristic of its voice [8].

Fadi Al Machot · Ahmad Haj Mosa · Alireza Fasih · Christopher Schwarzlmüller · Mouhanndad Ali · Kyandoghere Kyamakya
Alpen-Adria-University Klagenfurt, Institute of Smart System Technologies,
Transportation Informatics Group, 9020 Klagenfurt, Universitätsstrasse 65-67, Austria
e-mail: Fadi.AlMachot@uni-klu.ac.at

The system is able to determine the reaction time of the driver and can warn him. In general, an audio signal is used for warning.

We developed a "Driver Fatigue Warning System" which is based on voice analysis with the purpose of preventing a driver from crashing. The idea is to recognize sadness, angriness and normal mood of the driver. The motivation for mood detection is based on the fact, that aggressive behavior on road leads also to traffic accidents. In that aggressive mood, driver's pitch and volume of his/her voice increase. This change is observed and the system can respond adequately. This system supports other ADAS for monitoring and hence, increases the total safety of on road.

In the recent results of speech emotion recognition systems, researchers use a lot of features. Yacoub *et al.* use 37 features of the voice streams, Wu *et al.* use 52 features and Jian *et al.* use 32 features.

In this paper we use only 12 features to classify between 5 types of emotions based on Bayesian Quadratic Discriminate Classifier and we obtain a very high performance compared to the other existing works.

2 Related Works

Most of the previous works on emotion detection by analyzing audio streams are based on supervised learning by using different emotion classes and apply the standard pattern recognition procedure to train a classification model.

The state-of-the-art of emotion detection is divided into two branches; the first branch is the detection of emotion in music and the second one is the detection of emotions in voice streams.

In the branch of emotion detection in music, several emotion detection methods have been published, for example, Thayer maps the emotion classes on four quadrants in Thayer's arousal-valance emotion plane. He suggests a two dimensional emotion model that is simple but powerful. (In organizing different emotions response: stress and energy). The dimension of stress is called valance, while the dimension of energy is called arousal [4].

Yang *et al.* (2006), apply fuzzy classifiers, which assign a fuzzy vector for a song to indicate the relative strength of each class. The proposed system can be divided into two parts: the Model Generator (MG) and the Emotion Classifier (EC). The MG generates a model according to the features of the training samples, while the EC applies the resulting model to classify the input samples [1].

Synak *et al.* (2005), develops a multi-label classifiers which is able to detect more than one emotion (class) to the same song. Principally, the ideas of [1] and [5] are to provide emotion intensity measurement for each emotion class. They focus on automatic detection of emotion by analyzing audio streams, using features on spectral contents. The data set consists of a few hundred music pieces. The emotion are grouped into 6 or 13 classes [5].

Yang *et al.* (2007), formulates Music Emotion Recognition (MER) as a regression problem and Support Vector Regression (SVR) is applied to predict the arousal and valence (AV) values. With this regression approach, the problems inherent to

categorical approaches, are avoided. For instance, besides the quadrant to which the song belongs, one can further calculate, the emotion intensity the song expresses, by examining its arousal and valence (AV) values [3].

Han *et al.* (2009), their recognition system consists of three steps, the first step is the extraction of seven main features from music pieces, the second step is the mapping into eleven emotion categories on Thayers's two-dimensional emotion model. Finally, two regression functions are trained by using Support Vector Regression (SVR), followed by the predicting of arousal and valance values.

In the branch of emotion recognition of a human voice, a few research based on rough Set theory is done. Zhou *et al.*, use an approach based on rough Set theory and SVM for speech emotion recognition. The experiment results show that this approach can reduce the calculation cost while keeping high recognition rate [6].

Yacoub *et al.*, they distinguish between different classes of emotions e.g. sadness, boredom, happiness, and cold anger. They compare results from using neural networks, Support Vector Machines (SVM), K-Nearest Neighbors, and decision trees [7].

The approach of Nwe *et al.* (2003) has several classes of emotions namely, the archetypal emotions of anger, disgust, fear, joy, sadness and surprise. They create a data base of 60 emotional utterances, which are used to train and the proposed system by using Hidden Markov Models. They compare the Low-Frequency-Coefficient (LFPC) features with feature of the Linear Prediction Cepstral Coefficients (LPCC) and Mel-Frequency Cepstral Coefficients (MFCC) features [10].

In this paper, we show that the usage of Bayesian Quadratic Discriminate Classifier (QDC) enables a real-time processing with a low amount of features (12 features). We are able to reduce the calculation cost while keeping high recognition rate.

3   Overall Architecture of the Emotion Detection System

In this section, we present the overall architecture of the emotion detection system (see Figure 1). As training data, the Berlin emotional speech database is used to classify discrete emotions. This publicly available database is one of the most popular databases used for emotion recognition, thus facilitating comparisons with other works. Ten actors $5m/5f$) each uttered 10 everyday sentences (five short and five long, typically between 1.5 and 4 s) in German; sentences that can be interpreted in all of seven emotions acted. The raw database (prior to screening) has approximately sentences and is further evaluated by a subjective perception test with 20 listeners. Utterances scoring higher than 80% emotion recognition rate and considered natural by more than 60% listeners are included in the final database.

In total, we extract 12 features from each sample (see Table 1):

- The minimum, the maximum, the mean and the median of the energy.
- The minimum, the maximum, the mean and the median of the pitch of the signal.
- The minimum, the maximum, the mean and the median of the Mel-frequency cepstrum (MFCC) of the signal.

To extract these features, we use the statistical moments (minimum, maximum, mean and median) of 3 features (MFCC, Pitch and energy) and then for the classification, we use a quadratic discriminate classifier (QDC).

Table 1 The extracted Features

Features
ENERGY − MAX
ENERGY − MIN
ENERGY − MEAN
ENERGY − MEDIAN
PITCH − MAX
PITCH − MIN
PITCH − MEAN
PITCH − MEDIAN
MFCC − MAX
MFCC − MIN
MFCC − MEAN
MFCC − MEDIAN
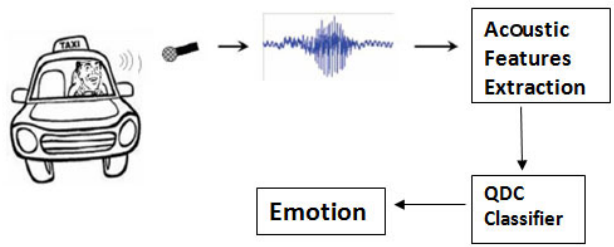
Fig. 1 The overall architecture of the emotion detection system

3.1 The Mel-Frequency Cepstral Coefficients (MFCCs)

MFCC features: the mel-frequency cepstral coefficients (MFCCs), firstintroduced in (Davis and Mermelstein, 1980) and successfully applied to automatic speech recognition, are popular short-term spectral features used for emotion recognition[13].

3.2 The Pitch of a Signal

Fundamentally, this algorithm exploits the fact that a periodic signal, even if it is not a pure sine wave, will be similar from one period to the next. This is true even

if the amplitude of the signal is changing in time, provided those changes do not occur too quickly. A pitch detector is basically an algorithm which determines the fundamental period of an input speech signal. Pitch detection algorithms can be divided into two groups: time-domain pitch detectors and frequency domain pitch detectors [14].

3.3 The Energy of a Signal

The size of a signal is very important for different applications. We define the signal energy as the is the area under the squared signal [12]:

$$E_f = \int_{-\infty}^{\infty} |f(t)^2| dt \tag{1}$$

Table 2 The classes of the decision table

Class	Class symbol
Afraid	1
Normal	2
Angry	3
Sad	4
Happy	5

4 Support Vector Machines SVM

In this section, most of results of the state-of-the-art are presented using Support Vector Machine (SVM). Support vector machines (SVMs) (Vapnik, 1995) are used for recognition of both discrete and continuous emotions. While support vector classification finds the separation hyperplane that maximizes the margin between two classes, support vector regression determines the regression hyperplane that approximates most data points with precision. The SVM implementation in (Chang and Lin, 2009) is adopted with the radial basis function (RBF) kernel employed. The design parameters of SVM are selected using training data via a grid search on a base logarithmic scale. In general, the RBF kernel can be a good choice as justified in because:

- It can model the non-linear relation between attributes and target values well.
- The linear kernel is a special case of RBF kernel.
- It has less hyperparameters than the polynomial kernel.
- It has less numerical difficulties compared to polynomial and sigmoid kernels [11].

5   Bayesian Quadratic Discriminate Classifier(QDC)

Bayesian Quadratic Discriminate classifier or QDC is based on the probability distribution of features vector in each class. To estimate the discriminate function between classes we have to first estimate the PDF of each class . For example, let us see the case of our three classes ω_1, ω_2 and ω_3 the first required statistical quantity is the prior probability of each class denoted by $P(\omega_1)$, $P(\omega_2)$ and $P(\omega_3)$ which can be obtained by the training data as [15]:

$$P(\omega_1) = \frac{N_1}{N} \tag{2}$$

$$P(\omega_2) = \frac{N_2}{N} \tag{3}$$

$$P(\omega_3) = \frac{N_3}{N} \tag{4}$$

Where N is the total number of the available training data , and N_1, N_2 and N_3 are the number of the features which belong to ω_1, ω_2 and ω_3. Another required statistical quantity is the class conditional PDF $P(x|\omega_1)$, $P(x|\omega_2)$ and $P(x|\omega_3)$ or the likelihood function, commonly Gaussian PDF is used as [15]:

$$p(x|\omega_i) = 1\frac{1}{(2\pi^{n/2}|\Sigma_i|^{1/2}}exp[-\frac{1}{2}(x-\mu_i)^T)\Sigma_i^{-1}(x-\mu_i)] \tag{5}$$
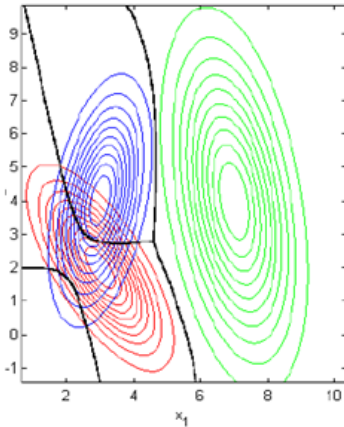
(a) The decision boundaries 2D   (b) The decision boundaries 3D
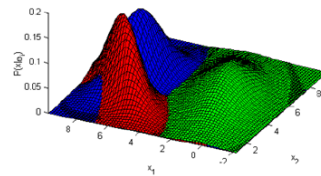
Fig. 2 The decision boundaries are hyper ellipses or hyper-paraboloids (quadratic) in 2 dimensions and three dimensions

6 Maximum Likelihood Parameter Estimation

ML method is used to estimate the unknown probability distribution function, for instance suppose $P(x|\omega_1;\theta)$ is the likelihood function with unknown parameter (θ), the ML method estimates the unknown parameter so that the *ML* function become maximum [15]. Suppose the following function is the log-likelihood function:

$$L(\theta) = lnP(x|\omega_1;\theta) \tag{6}$$

So we take the first derivative with respect to The maximum *ML* of θ is related to zero value of the first derivative:

$$\frac{d(L(\theta))}{d(\theta)} = 0 \geq MaxL \tag{7}$$

In our case the mean μ_i and the covariance matrix $\sum i$ are the unknown parameters for the class conditional PDF, by using *ML* we estimate μ_i and $\sum i$ for each class as:

$$\mu_{iML} = \frac{1}{N} \sum_{k=1}^{N} x_{ik} \tag{8}$$

Quadratic discernment function:

$$\Sigma_{ijML} = \frac{1}{N} \sum_{k=1}^{N} (x_{ik} - \mu_i)(x_{jk} - \mu_j) \tag{9}$$

Let $g_1(x), g_2(x)$ be the cost function of classes ω_1, ω_2 so x is classified to ω_1 if :

$$g_1(x) > g_2(x) \tag{10}$$

The decision surface which separates the two regions is:

$$g_{12} \equiv g_1(x) - g_2(x) = 0 \tag{11}$$

In our case the cost function:

$$g(x) = -\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} - \frac{1}{2}log(|\Sigma_i| + log(P(\omega_i)) \tag{12}$$

The decision boundaries are hyper-ellipses or hyper-paraboloids (quadratic) as shown in 2(a) and 2(b).

7 Results

In the recent results of speech emotion recognition systems, researchers in [6] use 37 features of the voice streams. They classify 6 types of emotions, their total accuracy

was 74% based on a combination of support vector machine (SVM) and Rough Set theory and 77,91% based on only SVM. The recognition rates of normal emotion is 90,50%, anger emotion is 86% and sadness is 66%.

In [11], the number of features is between (30-52), using Berlin database features of the voice streams. They classify 7 types of emotions, their total accuracy was 91.6% based on a Speaker Normalization (SN) and Linear Discriminant Analysis (LDA). The recognition rates of normal emotion is 77%, anger emotion is 82% and sadness is 92%.

In [7], authors use 4 statistical moments (mean, maximum, minimum and standard deviation) of 13 features. They classify 4 types of emotions(hot anger, cold anger, neutral and sadness). Their total accuracy was 87% based on Support Vector Machine (SVM).

In our case study, we use only 12 features to classify between 5 types of emotions based on Bayesian Quadratic Discriminate Classifier. For ADAS system, we focus on the 3 classes (sad, normal and angry), because they are strongly related to the representation of the behavior of drivers.

Table 3 The obtained results using three emotions (sad, angry and normal using QDC

	Normal	Sad	Angry	Total
Training set	69	52	117	238
Test set	10	10	10	30
False Positive	1	1	2	4
Detection ratio	90%	90%	80%	86,67%

Here, we present the experimental results after using different combinations of emotion classes. We use a quadratic discriminate classifier (QDC) and Berlin data base for emotional voices . For feature extraction, the statistical moments (minimum, maximum, mean and median) of 3 features (MFCC, Pitch and energy) are extracted. Table 3 shows that we used for training (69 voice files for normal, 52 for sad and 117 for angry) and for testing (10 for normal, 10 for Sad, 10 for angry). A 90% of normal and sad signals are correct classified while 80% of angry, So a 86.67% where the total result of our classifier.

Table 4 The obtained results after adding the fear emotion using QDC

	Happy	Normal	Sad	Angry	Total
Training set	60	69	52	117	298
Test set	10	10	10	10	30
False Positive	7	1	1	3	12
Detection ratio	30	90%	90%	70%	70%

Table 5 shows that the fear emotion is added to the training set to see the influence of this class on the classification in general. We retrain the QDC classifier and then we use the following test data (9 of fear, 10 of normal, 10 of sad, 10 of angry), we obtained the following results, 6 of 9 Afraid voices where false positives which forms 33.33% of success. This result is low because of the lack of training data (60 only) but the positive side of this experiment is that the recognition ratio of (normal, sad and angry) emotions is increased to (93.33%, 91.66% and 84.32%).

Table 5 The obtained results after adding the fear emotion using QDC

	fear	Normal	Sad	Angry	Total
Training set	60	69	52	117	276
Test set	9	10	10	10	61
False Positive	6	1	3	12	
Detection ratio	33,33	93,33%	91,66%	84%	80,32%

Table 4 shows that the happiness emotion is added to the training set to see the influence of this class on the classification in general. We retrain the QDC classifier and then we use the following test data (10 of happy, 10 of normal, 10 of sad, 10 of angry). We obtained the following results, 7 of 10 happy voices are false positives which form 30% of success, this low detection ration is because of the low number of the training sets. We also realize that the recognition ration is decreased (70%) for angry class, while no change is occurred to sadness and normal classes.

8 Conclusion

Speech emotion recognition system will be useful to understand the state and emotion of a driver. In this work, we came to know that the acoustic information could help to increase the performance of ADAS systems.

However, we show that the usage of Bayesian Quadratic Discriminate Classifier (QDC) enables a real-time processing with a low amount of features (12 features). We are able to reduce the calculation cost while keeping high recognition rate.

In our future work, we will perform the evaluation over different databases to check the robustness of the algorithms and to see the scalability of the algorithms. Further this work can be extended in the direction of reducing the acoustic noise generated by the vehicle and the vehicle entertainment systems to improve the quality of the ADAS system.

References

1. Yang, Y.H., Liu, C.C., Chen, H.H.: Music emotion classification: a fuzzy approach. In: Proceedings of the 14th Annual ACM International Conference on Multimedia, 3rd edn., pp. 81–84. ACM (2006)

2. Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I.: Multilabel classification of music into emotions. In: Proc. 9th International Conference on Music Information Retrieval (ISMIR 2008), 3rd edn., Philadelphia, PA, USA (2008)
3. Yang, Y.H., Lin, Y.C., Su, Y.F., Chen, H.H.: Music emotion classification: A regression approach. In: 2007 IEEE International Conference Multimedia and Expo., 3rd edn., pp. 208–211 (2007)
4. Thayer, R.E.: The biopsychology of mood and arousal, New York (1989); ISBN: 0195051629
5. Synak, P., Wieczorkowska, A.: Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, 3rd edn., pp. 314–322. Springer, Heidelberg (2005)
6. Zhou, J.: Speech Emotion Recognition Based on Rough Set and SVM. In: 5th IEEE International Conference on Cognitive Informatics, 3rd edn., Sch. of Inf. Sci.& Technol., Southwest Jiaotong Univ., Chengdu, Beijing (2006)
7. Yacoub, S., Simske, S., Lin, X., Burns, J.: Recognition of emotions in interactive voice response systems. In: Eighth European Conference on Speech Communication and Technology, 3rd edn. (2003)
8. May, J.F., Baldwin, C.L.: Driver fatigue: The importance of identifying causal factors of fatigue when considering detection and countermeasure technologies. Transportation Research Part F: Traffic Psychology and Behaviour 12(3), 218 – 224 (2009), http://www.sciencedirect.com/science/article/ B6VN8-VFBYG1-1/2/07087f8c3b6f88f9e9ed6996388d01ed
9. Jones, W.: Keeping cars from crashing. IEEE Spectrum 38(9), 40–45 (2001), http://ieeexplore.ieee.org/xpls/abs_all.jsp? arnumber=946636&tag=1
10. Nwe, T.L., Foo, S.W., De Silva, L.C.S.: Speech emotion recognition using hidden Markov models. Journal of Speech Communication 41, 603–623 (2003)
11. Wu, S., Falk, T.H., Chan, W.Y.: Automatic speech emotion recognition using modulation spectral features. Speech Communication, 167–6393 (2010); ISSN: 0167-6393
12. Maragos, P., Kaiser, J.F., Quatieri, T.F.: Energy separation in signal modulations with application to speech analysis. IEEE Transactions on Signal Processing 41, 3024–3051 (1993)
13. Molau, S., Pitz, M., Schluter, R., Ney, H.: Computing Mel-frequency cepstral coefficients on the power spectrum. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001), vol. 1, pp. 73–76 (2001)
14. Leslie Jr, G.F., MacKay, K.W.: Method and apparatus for pitch controlled voice signal processing. Google Patents, US Patent 4, 700, 391 (1987)
15. Srivastava, S., Gupta, M.R., Frigyik, B.A.: Bayesian quadratic discriminant analysis. Journal of Machine Learning Research 8, 1287–1314 (2007)

Enabling a Driver-Specific "Real-Time Road Safety" Assessment through an "Extended Floating Car Data" and Visualization System

Kyandoghere Kyamakya , Jean. C. Chedjou, Fadi Al Machot, and Alireza Fasih

Abstract. This paper does in a first step discuss the necessity of a "real-time road safety assessment" capability. Since traditional road safety has mainly been an off-line and post-event business, the fundamental question here is that of exploring ways/concepts and technologies for enabling a form of real-time and driver-context-specific road safety assessment before fatal events such as accidents happen. The motivation is evident, as it is well known that "prevention is better than cure". Concerning "fatal events" this paper also proposes a nuance and an extension of the concept. It does suggest a full range of event categories ranging from "hard fatal events" like a real accident to "soft fatal events" such as stress situation, some abrupt braking or just a lack of ergonomy of certain parts of the road traffic network in time and space. The availability of precious real-time data does in fact open a new research avenue for road safety. Real-time safety assessment that may be coupled to eco-driving considerations will enable a real-time recommender system for driver assistance and/or car navigation.

1 Introduction

Traditional road safety data just look at the history and mainly just of "real" accidents. Nothing or just very little is available concerning stressful and dangerous situations that do not provoke a hard accident. This work will provide a framework that will enable a form of spatiotemporal real-time road safety assessment; this is an innovation. The same will be done concerning fuel-consumption aware navigation. Of particular interest is also that the particular driver state and capabilities/skills will be taken into account. This opens new avenues for the strategic and tactical measures to design more efficient and safer mobility concepts. Traditional navigation systems are static and do lack dynamic information, although dynamic components could be very useful to adapt to a series of external

Kyandoghere Kyamakya · Jean. C. Chedjou · Fadi Al Machot · Alireza Fasih
Alpen-Adria-University Klagenfurt, Institute of Smart System Technologies,
Transportation Informatics Group, 9020 Klagenfurt, Universitaetsstrasse 65-67, Austria
e-mail: Kyandoghere.Kyamakya@aau.at

geo-spatial changes: weather, traffic, safety related road changes, etc. If the driver has the opportunity to pick up a stress-free and fuel-economic route, this would improve navigation, driving ergonomics and reduce accidents immensely. Moreover, if enough participants are involved in the system, they will get enough information of the traffic situation on the roads as well as information about potentially dangerous/stressful locations at the moment.

Drivers won't be the only beneficiaries of a system addressing these issues, but also institutions like the various Road Safety Agencies (in Austria it is called KfV – Kuratorium für Verkehrssicherheit), universities, driving schools (these for involving accurate knowledge of young learning-drivers behaviours under difficult conditions), etc.

Consider the driver. Today, it is well known that driver drowsiness, eventually compounded by the high workloads and stress of the ever-increasing complexity of the surrounding car and traffic environments, is a major cause of both driver stress and un-comfort and finally of severe accidents. Statistics show that over the past couple of decades the majority of the accidents on European roads are due not only due to the poor vehicle technical conditions but mainly to the driver's inattentiveness. The major cause for the inattentiveness has been found to be a deficit in what we call in this project an extended view of ergonomics, i.e. the "extended ergonomics status" of the driving process. This deficit is multidimensional as it includes aspects such as drowsiness (sleepy), fatigue (lack of energy) and emotions/stress (for example sad, angry, joy, pleasure, despair and irritation). In order to improve the "extended ergonomics status" during driving, the European Union (EU) has introduced regulations amongst others concerning driving time and rest periods for the drivers. Even though these regulations helped partially in improving the driver attentiveness, they largely ignore the real-time dynamic "extended ergonomics status" which is influenced by diverse factors such as the traffic control strategies, road geometry, vehicle characteristics, changing traffic scenarios, weather, etc. Therefore, there is a need for a measurement system, ideally a non-intrusive one, which can be used in extensive studies, by appropriate professionals, for studying the exact impact of the real-time spatio-temporal dynamics of the "extended ergonomics status" on road safety in general.

In essence, this paper does address the three following fundamental questions:

- Is there a need for a real-time road safety assessment? How far is this approach a novel paradigm while compared to the traditional road safety related way of doing the job?
- Which key data are necessary for a real-time road safety assessment? How can these data be collected in real-time in a cost-effective way? How can these data be archived to serve both real-time and offline usage requirements and/or communities?
- Which novel driver assistance and car navigation related services can be embedded in a large-scale floating car data based driver information system?

2   Necessity for Real-Time Road Safety Assessment

In the core this paper does focus on or motivate the development of a car driver information system while considering static as well as dynamic aspects, driver behaviours, driver specific profiles, and eco-driving. One does generally have a rather rough suspicion of typical road sections/portions/sites with high accident risk. Road intersections/crossings, railroad crossings, motorway accesses, icy bridges, foggy road sections, sharp turns or overcrowded roads are among the usual suspects. The traditional situation is that while trusting the suggestion of the navigation system when driving along the proposed route, one does however generally not receive any further information about possibly dangerous spots or road sections on the recommended route. Therefore, ongoing research activities in many countries have been trying to find solutions to overcome this limitation. But this can only be done stepwise. A first proposal was done by Malta *et al.* [1]. They found out that there is a strong relationship between hazardous situations and areas of frequent strong braking. For verification they measured the brake pedal force and considered speech and video for an additional source of information. Apart from braking behaviour, the speeding behaviour is also very important. Haglund *et al.* [2] conducted a very general survey on this aspect and found out that diverse attitudes constitute an important determining factor for decisions to speed-up or not to speed-up on a specific road section. Reasons for speeding are various; mainly factors like haste, pleasure, boredom, adaptation to others, unintended or to obey limits are among the most popular candidates [3]. Besides that, the perception of other driver's behaviour is an important factor behind drivers' speed decision [2]. For this work this aspect is very critical, because whenever drivers show wrong behaviours these "wrong behaviours" related data will be collected in the central database where they become available for further analysis. As a result, a warning of the wrong driving behaviour maybe displayed to the user after an appropriate assessment has been performed. Drivers who perceive others to drive too fast are more likely to drive too fast on their own [2]. Many of the navigation systems on the market do support the information related to local speed limitation.

As mentioned before, speeding and obeying to speed limitations are related. The reasons why people try to obey limits are mainly due to safety and fines. Furthermore, the goals of developers and researches in the area of driver assistance systems lie mainly on safety aspects. Economical and ecological aspects are not seen as so important by now. But when you think of the fuel consumption and its relations to the emission of carbon dioxide and its effects on our environment, eco-driving aspects should become more important. Therefore, eco-driving and "real-time" safety concerns are two main goals of this work. Drivers don't obey speed limits for environmental or economical reasons only. But for safety concerns the chances are higher that they obey. Thus, if there is a combination of safety, economical and ecological aspects, this will be profitable not only for the environment. The driver saves money, time and anger. A possible approach in this direction was presented by Choi *et al.* [22]. Within their system no relation of speed profiles to the locations can be found. In this work however the spatio-temporal correlation of speed profiles will be a concern.

What is needed is an intelligent system within the car that will monitor, besides the relevant external environment of the car, both driver status and driver behaviour according to the respective geospatial location. Based on the observed driving behaviour an assessment of (a) safety risk; (b) fuel consumption; and (c) the driver "extended ergonomics status" will be assessed in real-time.

Such a system does in fact enable some form of on-the-fly "perfective driver training" (that is, on the job while normally driving) through an appropriate recommender system that will become active whenever the driving profile does reflects abnormality concerning both safety and eco-driving concerns.

It should be noticed that the algorithmic to calculate the safety risk should take the specific user into consideration. Thus, an online user classification based on the spatio-temporal driving behaviour must be designed and incorporated in the above mentioned in-vehicle intelligent system.

It is evident that the safety risk assessment will be highly dependent on the overall driver "extended ergonomics status". Therefore, a robust measurement of this driver status is one of the key concepts of this paper.

3 Short Description of the First Module: An Extend Floating Car Information and Evaluation Data Prototype Concept (E-Friend)

The objective of E-FRIEND is the development of a car driver information system capable of assessing and then of visualizing for both car drivers (real-time) and researchers (both real-time and offline) safety as well as fuel-consumption related attributes of all major road segments in both urban areas and intercity roads and highways. Of course this module will be limited to a "proof-of-concept".

The data collection involves a novel Extended Floating Car Data (XFCD) concept. Following data and activities will be collected from appropriate sensors in the car (most from the CAN bus): steering, brake pedal and gas pedal activities; fuel consumption; the actual position through GPS; weather information; lighting information; the voice of the driver if he does speak; and finally 'driver face observation' through an onboard camera.

The collected data will be locally pre-processed and prepared for an efficient wireless transmission to a central server. There, appropriate algorithms will extract a series of interesting spatio-temporal information attributes and patterns: spatio-temporal real-time "road safety" assessment; driver classification; spatio-temporal eco-driving; etc. The extracted information will then be provided, on demand, to a recommender system (use case 1) for real-time driver assistance, and/or (use case 2) to researchers and administrations like "Road safety agencies", Universities, Driving Schools, etc. Thereby, the system considers driver's individual preferences as well as the respective driving skills. Such a system does also perform, implicitly, a sort of customized "safety & eco-driving" auditing of the involved roads; this can be visualized in a GIS (Geographic Information System). Considering the growing popularity of car navigation systems, helping drivers to find a

(short/best) route from A to B, E-FRIEND will extend this functionality by providing information about fuel-economic routes as well as both "driving skills" and "real-time-safety" sensitive road selection through an appropriate recommender system.
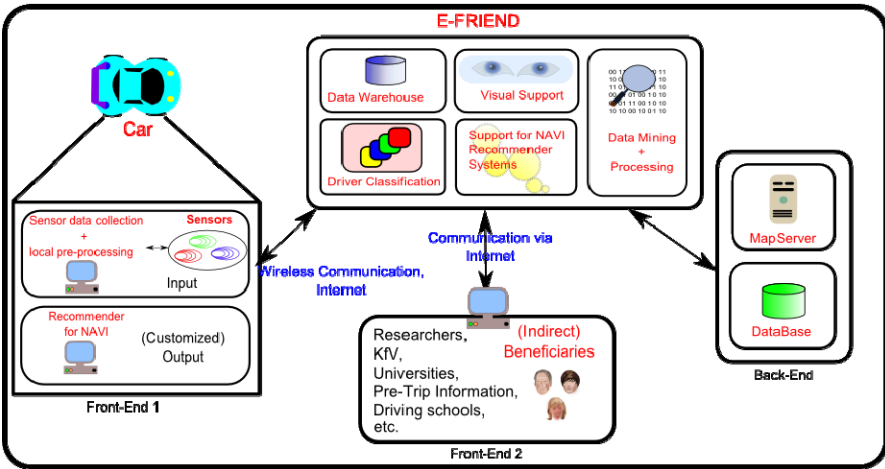
Fig. 1 Overall system structure of the core of the extended floating car data system

3.1 Scientific and Technological Aims of This E-FRIEND Module

The scientific perspective of E-FRIEND is the use of available (in the cars) traffic and safety relevant data and measurable driver behaviour parameters to expand the quality of existing navigation systems with regard mainly to real-time safety and eco-driving. As the number of different sensor technologies has considerably improved in the last couple of years, it seems not to be a big problem to gather the various data from cars. Their usage offers a great variety of possibilities and the full potential is far beyond being taped. New cars dispose of many different sensorial data. They are aimed at well-defined purposes like driver safety and comfort. Thus, there is still room for improvements through clever data combination.

The aim of this E-FRIEND module focuses on the selection of data from different sources (moving cars) to build a system that recommends routes on behalf of the observed driver's mental state, his/her driving skills and his/her preferences in real-time.

First, we will meet the problem that there is not enough solid quantitative information about driver behaviour in respect to the overall driving process as well as in respect to the behaviour in particular situations. Second, there is less information about how far the gathered information may provide insights in which issues do contribute to the crash/accident risk and stressful situations. Besides, these

car-driver dependent elements (such as the brake pedal force, gas pedal force, steering position) as well as other collectable car- driver independent parameters (like weather condition, daytime, season, etc.) take an essential position/place in our system and have to be recorded.

After the data collection and we expect that during the evaluation we will be able to find stereotypical behaviours which are essential for driver classification. To sum up, we can say that E-FRIEND observes a driver's driving attitude by collecting special data and return the findings and conclusions in an easy understandable way in form of route recommendations.

Additionally, we consider the fact, that some people are likely to plan their trip in advance. Therefore, the main source of information is the Internet. For that case, we plan a representation of general findings based on measured data in our database and some user inputs. Thus, pre-trip planning will become easier. Reference [6] did something lightly similar (and inspiring); but in contrast to this approach we will concentrate on the real-time road conditions and driver classification.

In literature, one finds many scientific approaches addressing the problem of examining the driver's behaviour and the benefits for the whole driving process [6, 7, 8]. Reference [6] for example, applies a driving simulator to record the behaviour of a group of less experienced drivers in an artificially produced episode to collect driving data and to improve their driving behaviour based on the results afterwards. The outcome of this experiment was that people taking part in this experiment are less likely to have a crash. In many cases, people don't want to be trained in the described way. They want information about possible difficult circumstances at hand (i.e. on the display), in real-time, fitted to their peculiarities and clearly comprehensible. These aspects will be addressed by the work related to this paper. As a matter of fact, drivers who are not well informed may select alternatives which are not in their best interest. Moreover if the necessary information is available, drivers with similar preferences will tend to concentrate on the same routes for the same (similar) departure times [8]. Impacts of this fact would be various: improved safety, higher quality of mobility, decreased direct and indirect costs (for society in general) of mobility and safety related concerns, reduced environmental impacts and better energy efficiencies.

3.2   Technical-Scientific Description of E-FRIEND

The core concept is presented in Figure 1. Today's cars are equipped with different sensors and technologies measuring parameters of their actual context and automatically adapt their behaviour to these aspects or inform the drivers about the changing state. As the driving context is full of information and challenging situations, keeping track of these dynamic aspects is quit demanding. The technologies used in cars of tomorrow should improve the currently used methods. To achieve this goal, new approaches are needed. One of these promising methods is the Floating Car Data concept. This approach profits from the aggregation of simple measured on-board data (i.e. speed, location, direction, etc.) into information of higher

quality. A local wireless communication unit integrated on each vehicle taking part contribute to the interaction among the cars. Here, different strategies were established stating where the working process has to take place. For example in Ref. [9] a decentralised approach was preferred. In [10] a master vehicle as dynamic data collector was integrated, which pre-processed the arrived in-formation. Afterwards the results were sent to a centralised operation centre.

The proposed concept tries to improve the quality of navigation systems through the use of Floating Car Data (FCD). As mentioned before, cars are able to collect a range of various parameters, send them to specific locations where special algorithms are preformed to produce new information. Car navigation systems (CNS), on the other hand, have a positive influence on the risk of having an accident, because they help people to reach their destination more quickly, with less driving time, less kilometres and less stress and uncertainty in-dependent of the kind of the preferred interaction mode (i.e. visual, auditory, visual-auditory-combination) [11]. It continuously keeps track of road information like direction, speed limitation, and obstacles based on the road map database. Furthermore the driver will be informed about events. The choice of the road type (expressway, local road, etc.) is open to the driver's preferences [12]. But still a recommendation showing the road best suited for him/her depending on one's own driving skills and actual mental state (drowsiness, lack of concentration, etc.) is missing. Many drivers may not want to take the shortest way to their journey destination. They perhaps favour a road that is less stressful and fuel-economic. This means, routes with traffic jams or much traffic volume, places were they have to raise their attention level (i.e. schools, hypermarkets, etc.) or accidental hotspots as identified by traffic statistics will be avoided. Others that are not familiar with the surrounding may like to know in advance what difficulties appear on their selected route.

In general the E-FRIEND system will collect real-time traffic data and information about the driver behaviour via sensors, send them though WLANs to an evaluation component where they will be saved in a Data Warehouse (see Figure 1). Furthermore, some static information about the driver like age, place of residence, gender, driving year, traffic accident history [7], etc. has to be taken into account. This information in combination with the dynamic driver behaviour contributes to the driver classification. The collected data are to be processed and as result typical driver patterns will be identified through pattern matching. A visual support is added, so that other traffic data interest groups have access to the collection via Internet whether to gain information before they start a trip or for constructional (road geometry related for example) measurements or for other diverse analysis purposes.

3.3 SOME Additional Details on the Technical Approach of E-Friend

The first step does consist of the data collection (see Figure 2), which is needed for later algorithm development. Therefore Ref. [13] has used a special kind of

simulator to gain information about the driving behaviour (control for gas pedal, break pedal, steering wheel and other peripherals inside a car). Pacaux-Lemoine *et al.* [14] have used the SHERPA-Simulator. As we do not have a comparable simulator by now, we are forced to deduce the data from cameras and some sensors like Gomez *et al.* have proposed in their work [15, 16]. They used an ordinary car equipped with two cameras, one positioned on the dashboard with connection to a laptop and one on the backseat, which has captured not only behaviour but also the audio output. As we want to capture also the pedal and the steering wheel activity, additional sensors may be used. But these data are easily obtainable from the CAN bus of the car.
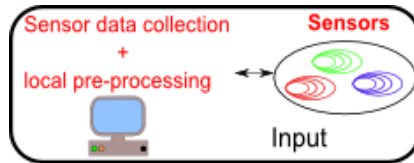
Fig. 2 Data Collection; architecture in the mobile terminal within the cars

The second step consists of the development of a data warehouse. The transmission of the collected data into the Data Warehouse is crucial and realized automatically. Subsequently a data evaluation will take place. In Figure 3 some possible results are shown for illustration purposes (this just to give an idea on how the data may look like). Parameters such as the energy-consumption, steering wheel position, vehicle speed, brake and gas pedal have to be combined with the geographical position of the car and the time the car passed this position. Then we are able to extract the location of observed irregularities and in combination with video and audio we can identify the reason why there were problems. This allows us to classify drivers and to develop strategies for the recommender system. The classification consists of two parts. First of all, personal information about the driver is acquired (age, gender, driving experience in years, place of residence, car type, etc.). For example the actual location is used to find out the details about the surrounding the driver is familiar (i.e. city, countryside). The age classifies the driver as old or young. An old driver means affected reactions. In many cases they prefer less stressful roads with a very small number of dangerous spots. The lacking experience makes very young drivers reckless. The second part is the driver classification. This will be a very challenging task, because of the many factors to be taken into account by the system.

Step 3 will contain the development of a visual component, so that the results are not only presented in code but also in a more comprehensible way. The next step will contain the development of a recommender system for NAVI systems based on the acquired knowledge about different driver classes.
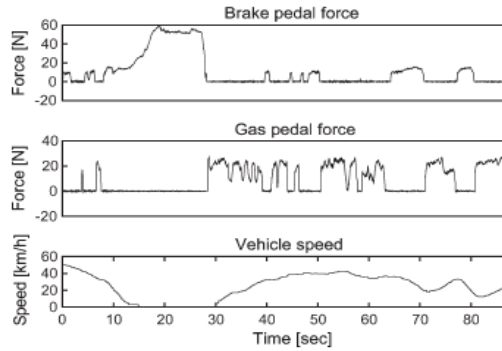
Fig. 3 Examples of driving behaviour signals; just for illustration purposes. Source: Reference [1]

3.4 Technical Problems and Risks

One problem maybe the classification of drivers based on their behaviour. As mentioned before, speed is a very popular parameter. Drivers tend to adapt their behaviour to other drivers independent of the street's difficulty-level and accident risk. The system keeps wrong measurements resulting in wrong prediction and feedback from the other drivers. A combination of individual behaviour history of this type with established speed limits on the observed road sections is necessary for an appropriate filtering. Perhaps data from accident statistic have to be integrated. Another aspect is that E-FRIEND shows a suitable, stress-less route for each driver. This can bring drivers to misuse the tool. "Less-stressful" may (but not necessarily) in E-FRIEND mean "not much traffic", not many dangerous spots, etc. Thus, the driver could increase speed and misuse the system for the calculation of fast routes. It will be very hard to overcome this problem, because the user does need the system not only for this task. Acoustical warnings or display messages are possible but sometime not very efficient solutions.

Further, the system does consist of many important components. For the production of a good application, enough data for the verification of E-FRIEND are needed. The various components are tightly connected; thus appropriate testing will be crucial. If we integrate E-FRIEND in NAVI-Systems, further tests and adaptations will be necessary. Therefore, we have planned extensive driving tests with various driver candidates.

4 Short Description of a Robust Non-intrusive Multi-modal Real-Time Monitoring System for Both Driver State and Driving Process Ergonomics

An important portion of the sensory system mentioned in Figure 2 is the sensor system destined for driver status monitoring.

4.1 Motivation

Today, it is well known that driver drowsiness, eventually compounded by the high workloads and stress of the ever-increasing complexity of the surrounding car and traffic environments, is a major cause of both driver stress and un-comfort and finally of severe accidents. Statistics show that over the past couple of decades the majority of the accidents on European roads are due not only due to the poor vehicle technical conditions but mainly to the driver's inattentiveness. The major cause for the inattentiveness has been found to be a deficit in what we call in this project an extended view of ergonomics, i.e. the "extended ergonomics status" of the driving process. This deficit is multidimensional as it includes aspects such as drowsiness (sleepy), fatigue (lack of energy) and emotions/stress (for example sad, angry, joy, pleasure, despair and irritation). In order to improve the "extended ergonomics status" during driving, the European Union (EU) has introduced regulations amongst others concerning driving time and rest periods for the drivers. Even though these regulations helped partially in improving the driver attentiveness, they largely ignore the real-time dynamic "extended ergonomics status" which is influenced by diverse factors such as the traffic control strategies, road geometry, vehicle characteristics, changing traffic scenarios, weather, etc. Therefore, there is a need for a measurement system, ideally a non-intrusive one that could be used in extensive studies, by appropriate professionals, for studying the exact impact of the real-time spatio-temporal dynamics of the "extended ergonomics status" on road safety in general.

4.2 Background

Different approaches have been proposed for monitoring the driver states, especially drowsiness and fatigue. A first class of approaches looks at the vehicle performance/behavior in order to indirectly infer the driver status. However, these indirect approaches heavily depend upon vehicle and road conditions (e.g. quality of lane markings, alternate lane markings during road repairs) as well as on environmental conditions (e.g. shadow, rain and night vision). These drawbacks have drawn the researcher's interest to directly monitoring the driver behavior. Thus, a second class of approaches does directly measure driver physiological characteristics but in an intrusive way by involving measurement systems such as the Electroencephalogram (EEG) which monitors brain activities, the Electrocardiogram (ECG) which measures heart rate variation, the Electrooculogram (EOG) which monitors eye movement, the skin potential level measurement techniques, etc. These methods of the second class of approaches do need the driver's cooperation as the electrodes are attached directly to the driver's body. Due to an expected very limited user acceptance of these intrusive methods in normal vehicles, they are more realistic for a daily use rather only in health care or similar special vehicles. A further problem is that the intrusive apparatus involved in these methods may itself contribute to the driver's distraction and fatigue. A third class of approaches is better since being non-intrusive. Non-intrusive approaches generally involve machine vision as an alternative to a direct measurement of physiological

characteristics and they do not need any cooperation from the driver; they monitor the driver behavior and status directly through visual sensors. Video sensors are placed on the dash board to measure, for example, eyelid movements (open/close interval of eyelid), head movements, mouth movements (yawning) and face expression measures.
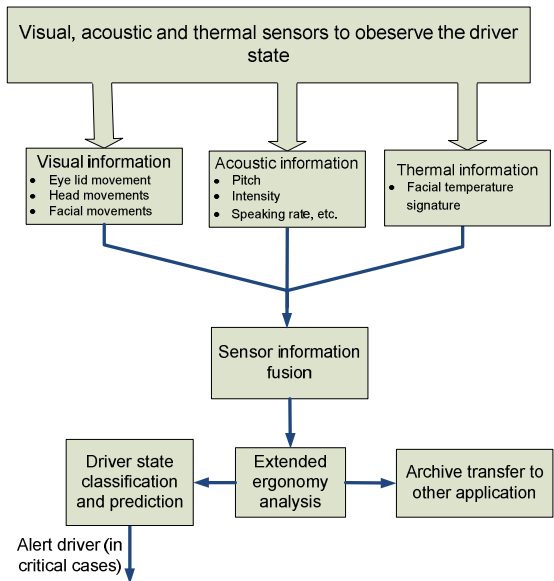
Fig. 4 Global concept description of the proposed "extended ergonomics status" monitoring system

The key objective of this module is to develop and validate a robust and at best low-cost non-intrusive system capable of reliably measuring all parameters needed for assessing the extended concept that we call in this project "extended ergonomics status". This "extended ergonomics status" concept is multidimensional and includes the following dimensions of the driver state: fatigue (lack of energy), drowsiness (sleepy) and emotions/stress, whereby emotions further include other sub-dimensions that are: anger, despair, pleasure, sadness, irritation, and joy. Fatigue is traditionally measured by looking at the eyelid movements. The drowsiness is generally measured by analyzing either head movements patterns or eyelid movements or face expressions or all the lasts together. Concerning emotion/stress recognition visual sensing of face expressions is helpful but generally not always sufficient. Therefore, one needs additional information that can be collected in a non-intrusive manner in order to increase the robustness of the emotion/stress measurement in the frame of a non-intrusive monitoring policy. We choose and find acoustic information to be appropriate. Provided the driver generates some

vocal signals by speaking, shouting, crying, etc., what is not un-common during the driving process, it is possible as has been shown in some recent works from literature, to extract from those acoustic signals specific patterns that do correlate with a set of emotional states. From the acoustic signals, this project will calculate pitch energy, intensity, speaking rate and voice quality, and correlate it to parameters/dimensions of the "extended ergonomics status" vector. Thus, a combination of visual sensing of face expressions (normal camera(s) + thermal camera) and pattern recognition from driver's voice is expected to significantly improve the robustness of the non-intrusive "extended ergonomics status" perception.

4.3   Problem Statement

The key objective of this module is to develop, implement, test, fine-tune and validate through extensive tests a 'proof of concept' of a non-intrusive multimodal measurement system for a robust real-time assessment of what we've called in this work "extended ergonomics status" of drivers. The following three sources of information will be used and fused: (a) camera-based visual perception of: face expressions, eyelids movements, head movements, and eventually hands' movements; b) face thermal signature through a thermal camera; and (c) voice analysis.

The measurement system should be able to reliably classify the perceived multimodal evidences. Then an inference of the corresponding "extended ergonomics status" parameters should be performed. Both modeling and prediction of the states is a further issue that should be solved by an appropriate paradigm; Dynamic Bayesian Networks, for example, have been extensively used in the relevant literature for tasks similar to this.

The key research questions which are also indicating the key innovation are related to: a) the overall systems engineering; b) the adaptation and fine-tuning of known algorithms and schemes for solving partial tasks of the system; c) finding the appropriate fusion and classification concept; d) designing and optimizing the appropriate 'states modeling and prediction' approach; e) prototypical implementation in software and hardware; and f) concept fine-tuning (feedbacks) and validation through extensive tests (involving different individuals), and comparison, where possible, with predicted results from published and relevant psychological and ergonomics theories (It should be however clearly noticed that the focus of this project is not to conduct psychological studies but rather to develop a non-intrusive robust measurement system that can be used, amongst others, for such studies for the driving process context).

4.4   Overall Research Approach

After a systematic systems engineering of the overall system, a well justified selection of convincing known approaches from literature will be performed to solve partial tasks. However, in many cases an adaptation or extension of the selected

schemes will be performed where appropriate. Nonetheless, novel schemes should be developed where necessary. For non-linear features extraction and a related classification from each of the modes involved, the arsenal of methods to select from does include the followings: wavelet transforms, radon transforms, cellular neural networks (CNN), support vector machines (SVM), rough set theory, adaboost, and a well dosed combination of some of these techniques. Then follows the fusion of the features extracted from each of the modes (camera based visual information, face thermal signature information, voice pattern information) that may/will be coupled to a subsequent inference of the "extended ergonomics status" vector. The inference will involve some form of dynamic bayesian networks (DBN) that will be adapted to the underlying problem setting. A prototypical implementation of the core system is so far important that the extensive experiments to be conducted will be used in a first step to fine-tune both the designs of subsystems' components and the diverse algorithmic choices. In a final stage, the issue will be of course the validation through a comparison with basic predictions based on solid knowledge to be extracted from published relevant psychological and ergonomics literature. A very useful task will be amongst others to determine and fix a very useful set of performance criteria, which is consistent with the requirements of the targeted application scenarios for the measurement system to be developed. A collaboration with both a traffic psychology and ergonomics expert/researcher of the university of Klagenfurt as well as with some road traffic safety experts/researchers (one from a polish university and one from a finish university) is planned, especially in the requirements engineering and validation phases.

5 Open Algorithmic Issues Concering the Real-Time Road Safety Assessment

An appropriate algorithm for the effective and dynamic real-time assessment of the real-time road safety while using the data collected through the systems described in the two previous sections is being developed in a research work that is still in progress. First results are expected within the next couple of months.

6 Some General Ideas about the Setting-Up of the Recommender System

Our concept does include a recommendation system. In general, recommender systems are destined to helping people through a safe navigation and thereby coping with a potential information overload in ubiquitous computing environments. But the construction of accurate user models and their use is very challenging. Furthermore, the integration of context aspects into recommender systems is a rather novel and unexplored research and application area, because recommender systems mainly concentrate on static, stored, observed or provided information focused on user preferences. Our system will include consider the specific context which is highly dynamic and not stored. The possible customer target group for this system is huge, not to mention the diversity of drivers in different contexts.

Mahmud *et al.* have conducted a survey on navigation systems amongst younger and elderly drivers. In many cases the elderly blindly trusted the advice of the navigation system (NAVI), but they could not fully relax with the system and got confused when the NAVI device has suggested an unknown or uncommon route. E-FRIEND will solve this problem, because it shows the reasons for a special route selection. A classification of drivers by age may not be very efficient. These parameters do not tell anything about the way drivers guide their car and his/her driving experiences. An elderly person may have less experience than a younger person. For this reason, we include dynamic and real-time information about a person by establishing a driver profile and classifying the driver bases of his/her profile and the current context. Driver classification is a challenging task but some interesting approaches can be found.

Woerndl *et al.* have implemented a similar recommender system. In contrast to our system their approach is aimed for mobile applications in general and not explicitly determined to driver assistant systems. Their version has to be downloaded on mobile devices. Our recommender system will be integrated in the NAVI and be available over the Internet. Additionally, besides geographic location and time, we are bearing further elements/data such as pedal force and steering wheel position and energy consumption in mind.

Roughly speaking, one can state that E-FRIEND and the other system modules are a combination of various information sources and technologies to support its system users. But one can not say that our system concept is aimed at the supply of reckless drivers with information where they can drive with one's foot down to the floor. In addition, our system is not aimed at insurance companies to help them in their calculation of insurance premiums. It is only an application that helps the car drivers in reaching his/her destination with less stress, more controlled safety, and/or low fuel consumption while taking both individual and external factors into account.

7 Conclusions

This paper has presented a series of ideas and system concepts related to the hot topic "enabling real-time road safety". In a first step it has discussed the necessity of a "real-time road safety assessment" capability. Since traditional road safety has mainly been an offline and post-event business, the fundamental question here has been that of exploring ways/concepts and technologies for enabling a form of real-time and driver-context-specific road safety assessment before fatal events such as accidents could happen. The motivation is evident, as it is well known that "prevention is better than cure". Concerning "fatal events" this paper also proposes a nuance and an extension of the concept. It does suggest a full range of event-categories ranging from "hard fatal events" like a real accident to "soft fatal events" such as stress situation, some abrupt braking or just a lack of ergonomy of certain parts of the road traffic network in time and space. Due to the fact that a majority of traffic accidents are attributable to how drivers behave in response to

the driving environment (mainly to the driver's inattentiveness), a key challenge is therefore that of setting up a cost-effective non-intrusive system that will be capable of measuring (a) the driver status or better s what we've called in this paper "extended ergonomics status" of drivers, and (b) a car driver information system capable of assessing and then of visualizing for both car drivers (real-time) and researchers (both realtime and offline) safety as well as fuel-consumption related attributes of all major road segments in both urban areas and intercity roads and highways.

The core modules of the system concept have been shortly presented, especially the quintessence as well as the motivation. This work is however still a work-in-progress. The modules presented are being implemented. After the implementation a series of extensive tests will be conducted and analyzed in order to demonstrate experimentally the feasibility of a true real-time road safety assessment.

This car driver information system will be embedded in an extended floating car data concept that will involve both intelligent mobile terminals in the cars and some central server systems. The two measurements systems (a) and (b) above are the core of a system architecture (actual development status: work-in-progress) providing real-time data that may be used for a spatio-temporal real-time road safety assessment. Appropriate algorithms for that purpose are to be developed in a future sub-sequent research. The availability of such precious real-time data does in fact open a new research avenue for road safety. Real-time safety assessment that may be coupled to eco-driving consideration will enable a real-time recommender system for driver assistance or car navigation.

References

[1] Malta, L., Miyajima, C., Takeda, K.: A Study of Driver Behavior Under Potential Threats in Vehicle Traffic. IEEE Transactions on Intelligent Transportation Systems 10(2), 201–210 (2007)

[2] Haglund, M., Åberg, L.: Speed choice in relation to speed limit and influences from other drivers. Transportation Research Part F: Traffic Psychology and Behaviour 3(1), 39–51 (2000)

[3] Institute for Road Safety Research, SWOV Fact sheet, Speed choice: the influence of man, vehicle, and road, Leidschendam, The Netherlands (December 2008)

[4] Choi, J., Yoon, D., Kim, K., Kim, H.: Analysis of the vehicle information for the economical and the ecological driving pattern. In: 11th International Conference on Advanced Communication Technology (ICACT2009), February 15-18, vol. 01, pp. 867–869 (2009)

[5] Kajiya, Y., Yamagiwa, Y., Kudo, Y.: Road Web Markup Language-XML for Roadrelated Information Distribution. Transportation Research Board-Annual meeting (2004), `http://rwml.its-win.gr.jp/papers-pdf/RWML-TRB2004-002373.pdf`

[6] Liu, Y.F., Wang, Y.M., Li, W.S., Xu, W.Q., Gui, J.S.: Improve Driver Performance by Experience of Driver Cognitive Behavior Model's Practice. In: 2009 IEEE Intelligent Vehicles Symposium, pp. 475–480 (June 2009)

[7] Lei, Z., Jianqiang, W., Furui, Y., Keqiang, L.: A Quantification Method of Driver Characteristics Based on Driver Behavior Questionnaire. In: 2009 IEEE Intelligent Vehcles Symposium, pp. 616–620 (June 2009)

[8] Jing, L., Guoxiong, P.: Dynamic drivers' behavior under the effect of advanced traveller information and guidance systems. In: Proceedings ICII 2001, International Conferences on Info-tech and Info-net, Beijing (2001)

[9] Haesevoets, R., Weyns, D., Holvoet, T., Joosen, W.: Hierachical Organizations and a Supporting Software Architecture for Floating Car Data. In: Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, pp. 31–36 (2008)

[10] Lin, L., Osafune, T., Lenardi, M.: Floating car data system enforcement through vehicle to vehicle communication. In: Proceedings of the 6th International Conference on ITS Telecommunications, pp. 122–126 (June 2006)

[11] Feenstra, P.J., Hogeman, J.H., Vonk, T.: Traffic safety effect of navigation systems. In: IEEE Intelligent Vehicles Symposium, pp. 1203–1208 (June 2008)

[12] Lee, S., Lee, D., Lee, S.: Network-Oriented Road Map Generation for Unknown Roads using Visual Images and GPS-based Location Information. IEEE Transactions on Consumer Electronics 55(3), 1233–1240 (2009)

[13] Imamura, T., Yamashita, H., Zhang, Z., Rizal bin Othman, M., Miyake, T.: A Study of Classification for Driver Conditions using Driving Behaviors. In: 2008 IEEE International Conference on Systems, Man and Cybernetics (SMC 2008), pp. 1506–1511 (2008)

[14] Pacaux-Lemoine, M.-P., Ordioni, J., Popieul, J.-C., Debernard, S., Millot, P.: Cooperating with an Assistance Tool for Safe Driving. In: Proceedings of the 16th IFAC World Congress, Human-Machine Interface Development and Collaboration, Session, Code: Fr-M16-TO, Prague, Czech Republic (July 08, 2005)

[15] Gomez, R., Popovic, V., Bucolo, S.: Driving experience and the effect of challenging interactions in high traffic context. In: Proceedings of the Future-ground International Conference (2004)

[16] Gomez, R., Popovic, V., Bucolo, S.: Driving: The Emotional Experience and Automotive Design. Paper Presented at the Fourth International Conference on Design and Emotion, Ankara (2004)

Placing Content in Hybrid Peer-to-Peer Networks by User Activities Consideration

Sunantha Sodsee, Phayung Meesad, and Herwig Unger

Abstract. To locate surrogate servers on optimal positions in hybrid P2P networks, in this paper, a locating of super nodes (or surrogate servers) by considering user activities is proposed. Active nodes are grouped by clustering algorithms and locations of super nodes are presented as centroids of clusters to increase a quality of service (QoS) and provide fast response to users, which can be applied in video distribution systems such as video-on-demand services. To evaluate our work, it is simulated on a grid overlay network with different number of both super nodes and active nodes. The simulation result shows that the communication time (response time) for accessing content is minimised and it is also related to the number of super nodes.

1   Introduction

With the tremendous growth of the Internet, content delivery networks (CDNs) [1], such as Akamai [2], become more necessary in order to support a high-performance delivery of content to a requesting user. A CDN is a technology implemented as a virtual network to distribute contents and resources. Its network is formed by server groups located in different locations. It regularly distributes the content to the edges and handles the traffic according to the contents. Finally, requests of users will be transmitted to the optimal servers, thus enabling users access to the information from the nearest server by the fastest speed. Its contents however can be various kinds of media such as audio, video, documents, images and Web pages, and one of most challenging applications is dealing with Video-on-Demand (VoD) service such as YouTube [3] because of a quality of service (QoS) requirement.

Sunantha Sodsee · Herwig Unger
Fernuniversität in Hagen, Germany
e-mail: {sunantha.sodsee,herwig.unger}@fernuni-hagen.de

Phayung Meesad
King Mongkut's University of Technology North Bangkok, Thailand
e-mail: pym@kmutnb.ac.th

In fact, CDNs are not only providing charge to their customers such as a bandwidth cost and a cost of number of surrogate servers, but also providing bandwidth grow proportionally with user population [4]. There are , however, some limitations of CDNs. First, most CDNs are client-server models. Therefore, they can face the single point of failure and bottleneck problems. For instance, popular Web services often suffer congestion and bottleneck due to the large demands made on their services. A particular server may has a heavy workload, its traffic is increased and then the Web site that holds the content becoming temporarily unavailable. Secondly, nodes of most CDNs are located at fixed location on the Internet [23]. Thus, they do not address the need of individual users to distribute their own content. In contrast, P2P networks, they are dynamic networks that nodes can be added or removed anytime. They also have an alternative method to benefit for sharing content among users (peers) on the Internet by using their available resources to propagate shared content.

Peer-to-peer (P2P) systems have been successful for large-scale content distributions. Their service cost are lower than CDNs and they can overcome some CDNs' problems. To increase qualities of service (QoS) in content distribution systems, including response time, latency, throughput and accessibility, combining advantages between client-server models and pure P2P systems are presented. Their combination are called *hybrid P2P systems* [10] that provide better scalability than centralised systems and show lower transmission latency than unstructured P2P systems. As well as, their infrastructures are also similar to CDNs in the case that there are network clusters containing super nodes providing services to their clients. Consequently, the super nodes are acting quite similar to surrogate servers in CDNs. Most well-known systems is KaZaa [10].

According to the benefits of hybrid P2P networks, in this work, a content distribution service based on hybrid P2P networks is focused. In order to place content, three factors here (see Fig. 1): content, network parameters and user activities should be considered to improve the performance of P2P networks. Both content and network parameters have been studied to distribute content, these are examples (see details in Sec. 2): [11] and [12] focused on the content consideration as a content replication distribution and a popularity of content, respectively and [14] paid attention on network parameters such as bandwidth of communication links to identify the suitable location for storing the distributed content.

In this paper, the last factor, user activities will be considered. Super nodes should be located at central points of networks based on user activities consideration to increase QoS, accessibility and fast response of users. All of them, however, are also main keys for streaming media distribution [5], then finally our proposed work can be applied in VoD systems in order to support a play-as-you-download experience.

The rest of the paper is organized as follows. State of the art is presented in Sec. 2. Sec. 3 shows basic ideas and details of our proposed work. To evaluate, an empirical proof is presented in Sec. 4. Finally, Sec. 5 draws conclusions and discusses about future works.
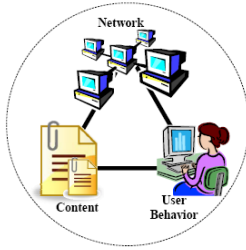
Fig. 1 Main factors influencing the performance of P2P networks

2 State of the Art

2.1 *Content Distribution Based P2P Networks*

For traditional CDNs, their reliability and performance are affected by the distributed content location, caching strategies, routing mechanism and data replication [6]. In addition, a data consistency is also important for their performance. Clients assume that they obtain content directly from the origin servers thus clients expect to receive a current content. Consequently, they also have a high cost of maintenance to support their reliability and performance. In contrast, P2P systems claim low cost and efficient distribution therefore they are increasingly used to deliver contents to end users as a P2P content distribution which provides more flexibility and higher availability through large-scale replication of contents at huge numbers of nodes (or peers)[7].

Napster [8] is a pioneer of P2P content distribution system or called P2P file sharing system. Its goals are to facilitate a location and exchange of files among a large group of independent users connected through the Internet. There are central servers to maintain an index of files that are shared by peers, which are connected to one of servers. Files querying of peers are sent directly to a connected server, later that the server return a list of matching files and locations then peers receive the results and directly exchange the files.

A few years later, Freenet [9] is implemented. It is also a well-known distributed content storage and retrieval based P2P system. Its search method is serial along random paths. Contents being forwarded are propagated from node to node and replicated on each node along the path. New links are established between the nodes participating in a request for a content and the chain of nodes visited up to the eventual source of the data. In addition, location-independent keys are used to store and retrieve contents. Freenet does not use broadcast searches nor centralised location indices, but searches serially.

To deal with the fixed-location nodes and the bottleneck problem of CDNs by using further distributed manners of P2P environments, cluster-based hybrid P2P systems or hybrid P2P systems are presented here to be content distribution systems. They are a combination of fully centralised and pure P2P systems (see Fig. 2).
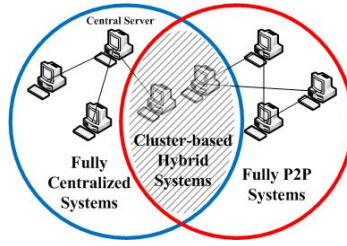
Fig. 2 Cluster-based hybrid systems

Clustering represents the small-world concept [24], because similar things are kept close together, and long distance links are added. The concept allows fast access to locations in searching and avoiding the bottleneck problem because of small number of members.

Most popular example for them is KaZaA [10]. KaZaA is one of existing P2P file sharing systems. It includes features both from the centralized sever model and the P2P model. To cluster nodes, certain criteria are used. Nodes with high storage and computing capacities are selected as *super nodes*. The normal nodes (*clients*) are connected to the super nodes. The super nodes communicate with each other via inter-cluster networks. In contrast, clients within the same cluster are connected to a central node. The super nodes carry out query routing, indexing and data search on behalf of the less powerful nodes. Hybrid P2P systems provide better scalability than centralised systems, and show lower transmission latency (i.e. shorter network paths) than unstructured P2P systems.

For a long time, many researchers have been proposed different techniques to increase the QoS in existing content distribution systems. They are

- A surrogate server concept: providing content stored in several servers locating near users [2].
- A content replication distribution: reducing congestion point and improve the query efficiency by selecting query traffic hubs and frequent requesters as replica nodes, and dynamically adapting to nonuniform and time-varying content popularity and node interest [11].
- A partial indexing search: building a partial index of shared content in order to improve the success rate and search speed in locating content by maintaining two types of information: the top interests of nodes and unpopular content [12].
- An optimal path selection: a new routing algorithm with the concurrent idea of Petri nets and the techniques of Ant Algorithm. It can achieve a high routing efficiency and its reliability and hardware cost will be far less than the traditional CDNs [13].
- A server's load reduction: analysing user activities to reduce the server loading. When a live video is watched by users, these users can help each other to reduce the load on the server, consequently the QoS is increased [5].

The authors have also suggested in [14] that an importance of nodes in P2P networks, which is influenced by network parameters such as bandwidth of communication links, can be, however, applied to find the suitable location for storing shared content in order to access it conveniently. This idea has been motivated by the PageRank algorithm [25] from the Google search engine.

2.2 Clustering Algorithms

In content distribution networks, there are a lot of active users, which are called active nodes: a node whose activities relate to activities of a user that are performed frequently, joining to the networks in order to get their services. In contrast, the content, which is a shared file or content stored on a node available for other nodes, is regularly provided on a few flexible nodes, only. Consequently, active users should be assigned to the good QoS of content, as well as content servers should be placed at the central point of active users, which is called an optimal position.

The optimal position is defined by closing up to active users measured with distance functions. The distance between a user and a server is denoted as a communication time between them. The QoS, response time, latency, throughput and accessibility are also depended on the communication time. If the communication time is high then the QoS, response time, throughput and accessibility will be low, otherwise low latency. According to previous mentioned address, clustering will be applied to find a central point of active users and assign active users to optimal content servers.

A clustering is the process of grouping the active users into clusters, which are active-node groups: a group of active nodes whose distance to the group's centroid is smaller than their distance to the centroids of other groups, so that active users within the same cluster have high similarity in comparison to one another, but are very dissimilar to active users in other clusters. Referring to a member of cluster, it is a set of active nodes which can be found in the same cluster.

Herein, content server (or called super node or surrogate server) is a node which stores files or content available to other nodes. It stays centrally and close to the active nodes to provide a convenient content accessibility. The number of surrogate servers is increased in order to provide a good quality of service to the active nodes. The number of surrogate servers is simply equal to the number of active-node groups.

At present, there are several existing clustering approaches such as neural networks, K-nearest neighbors, hierarchical clustering, and K-means clustering [26], which are categorised depending on various aspects as follows [27].

- Hard clustering, e.g. K-means [26], it assigns each object to one cluster
- Hierarchical clustering, e.g. hierarchical agglomerative clustering [26], it splits clusters into sub-clusters by using dendrogram creation.
- Density-based clustering, it forms clusters by finding density-connected regions in the feature space.
- Neural network-based clustering, e.g. neural networks [26], it clusters objects based on its tuning weights.

In our work, clustering algorithms is applied to estimate the central points in order to locate content servers [15] because, in the real networks, central points are complicated to identify by other techniques due to a size and a dynamism of networks. It is because of our goal that each active users will be assigned to exclusively one cluster and there are fixed number of clusters, then K-means clustering algorithm is used. Its procedure is to group the given active users with a certain number of clusters, which is called k cluster. As well as each active user is assigned to the nearest cluster by concerning the distance measurement because the algorithm aims to minimise an objective function, which is a squared error function shown in Eq.1.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \|x_i^j - c_j\|^2 \qquad (1)$$

where, k is the number of clusters, n is the number of active users, and $\|x_i^i - c_j\|^2$ is a distance measurement, such as Euclidean function, between the active user x_i^j and the centroid of cluster c_j.

In addition, K-means clustering algorithm also has been applied to solve the classification problems in P2P environments [28].

2.3 Exploring Networks

P2P systems are decentralized systems, which are important structures for information and content management. One of the crucial methods for information and content management is an effective search.

The usually employed search method based on flooding [16] works by broadcasting query messages hop-by-hop across networks. This approach is simple, but not efficient in terms of network bandwidth utilisation. Another method, distributed hash tables based search (DHT) [17] is efficient in terms of network bandwidth, but causes considerable overhead with respect to index files. DHT does not adapt to dynamic networks and dynamic content stored in nodes. Exhibiting fault tolerance, self-organisation and low overhead associated with node creation and removal, conducting *random walking* is a popular alternative to flooding [18]. The random walking requires no knowledge of network structure, and is attractive to be applied in large-scale dynamic P2P networks, because it uses local up-to-date information, only. Moreover, it can easily manage connections and disconnections occurring in networks and provides a completely distributed framework with a less consumption of bandwidth [19]. Its shortcomings, however, are time consumption, especially in the case of large networks [20], no co-ordination among random walkers and its randomised procedures, which is difficult to estimate [19].

Another self-organizing methods are ant-based ones. In nature, ants are social insects that use a stigmergy [21] as an indirect way of co-ordination between them or their actions. This gives to a form of self-organisation, producing intelligence structures without any plans, controls or direct communication between the ants then imitating the behavior of ant societies was first proposed to solve optimisation problems by *Dorigo* [22]. Due to their behavior, ants can help each other to solve search problems in the networks. They will deposit the pheromone along the visited paths and follow the paths by the highest pheromone concentration. In fact, for a network exploration, all nodes of the networks should be visited then the ant should follow the trail with the lowest pheromone concentration, which is called MinorityAnt proposed by [29]. The MinorityAnt can visit all nodes in the network with almost the same probability and there is an indirect communication or co-ordination between ants by using the pheromone. Figure 3 shows a searching ability of MinorityAnt on a small-world network with 2,048 nodes in different time steps.
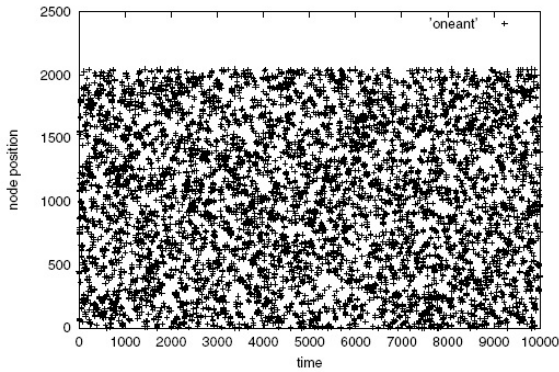
Fig. 3 The MinorityAnt

3   Placing Content by Considering User Activities

In the present section an algorithm for the place of content by considering user activities is presented. The contents' locations are found in decentralised systems in the course of agents, user activities and clustering approaches.

3.1   Basic Ideas

For our proposed work, it is focusing on content distribution problems in P2P networks. User activities are considered in order to place content in the network. Content should be delivered quickly and efficiently to querying users as well as users should browse content with better performance and reliability, these addressing are our objectives.

The distribution of queries in P2P networks generally follows a power law, that the most traffic is generated by a small number of nodes. To handle the most traffic in the networks, the active nodes will be focused. If the active nodes can access or retrieve their desired content conveniently, then the whole network traffic can be reduced. For our ideas the super nodes or surrogate servers should be located nearby the active nodes and the active nodes should be grouped based on their locations to get the service from closet servers. When searching for content, active nodes will send queries directly to their servers and therefore reduce the network traffic because a specific content does not need to be placed on many locations. Moreover, the problems of congestion and bottlenecks are avoided because the number of active nodes in the group is not large. Hence, a challenging question is *how to determine such a suitable location of server?* Fig. 4 presents a server location in different positions among the nodes which having the same level of active. If a server is located at the corner of active-node group (see Fig. 4(a)), then it is nearly staying to few active nodes and locating far away from some active nodes. It means that only few active nodes can access the content in the server conveniently because the distance between them is short, otherwise not. On the other hand, if a server is located at the central point of active-node group (see Fig. 4(b)), then all active nodes can access the content conveniently with a balance. Consequently, the server should be located at the central point of active-node group in order to perform an efficient service with a balance distance among users.

(a) Server located at the corner of active-node group　　(b) Server located at the central position of active-node group
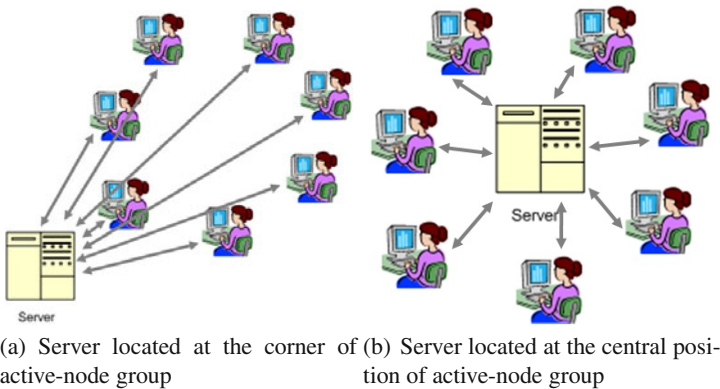
Fig. 4 Server location on different node positions

To concern the above mentioned address, in this work, agents are used to travel around the network to look for active nodes and K-means clustering approaches are used for finding the central points of active-node groups based on user activities in

order to represent the locations of servers. Active nodes are grouped by using K-means clustering approaches. The super nodes will be located centrally as centroids of active nodes' groups and stayed nearby active nodes that frequently do activities. When an active node tries to access content, the respective query is routed to the surrogate server closest to it in order to speed up the delivery of content and provide the better performance and reliability. Consequently, QoS will be increased as well as a network traffic and response time will also be decreased respectively.

Due to reliability, toleration of the nodes failure and no redundancy of connection, hereby, our work will be implemented on grid overlay network structure, which is traveled by agents. The sizes of networks are represented as the multiplication between the number of x-columns and y-rows, and a node is represented by a cross between x-columns and y-rows. However, general networks are also possible to work with, if they are represented as connected graphs.

Starting travel, herein, agents have a memory for storing active nodes' information and servers' information such as *members of clusters*, *centroids of clusters*, and they move from their present location to a neighbor selected based on their transition probability. If an agent visits to an active node, it will be grouped based on the agent's remembered information, and then its information will be updated such as *members of clusters* and it moves further to a neighbor selected based on its transition probability. Otherwise, it moves to a neighbor selected based on its transition probability without any processing. Later, when its remembered information, *members of clusters*, is not be changed or updated for a while, then its remembered information, *centroids of clusters*, is updated based on members of each cluster. The agent will remember new centroids of clusters and empty the members of clusters information. It will travel around the network to look for active nodes repeating the first and the second process until its remembered information, *centroids of clusters*, is not be changed or updated for a while, then *centroids of clusters* is represented the location of super nodes (or server) and *members of clusters* shows the active-node members of each server.

3.2 System Model

Let $G = (V, E)$ be an undirected graph to represent network topologies, where V is the set of nodes v_i, $i = \{1, 2, \ldots, n\}$, and $E = V \times V$ is the set of links e_{ij} and n is the number of nodes in the network. In addition, the neighbourhood of node i is defined as $N_i = \{v_j \in V | e_{ij} \in E\}$.

In our model, each pair of nodes is associated with a Euclidean distance representing a communication time or a response time between them. The Euclidean distance between a specific pair of nodes are measured directly by a virtual coordination of node, which is denoted as (x_i, y_i). On the other hand, a node that is represented as a centroid of cluster, its virtual coordination is (x_{c_k}, y_{c_k}), where $k = \{1, 2, \ldots, m\}$ is the number of clusters, which its node member is v_i^k. Finally, the

response time (R) calculation between a node member of cluster k and its centroid k is defined as

$$R = \sqrt{(x_i^k - x_{c_k})^2 + (y_i^k - y_{c_k})^2} \qquad (2)$$

where, the centroid of cluster is represented a location of server then the response time is informing a latency when the node member get a service from the server.

For exploring the network, herein, agents are applied to look for the active nodes in the network because of the dynamism of network. The used agents are Random Walking and MinorityAnt.

- Random Walking: a random walker on G starts at any node v_i at a time step $t = 0$. At $t = t + 1$, it moves to $v_j \in N_i$ selected randomly with a uniform probability p_{ij}, where $p_{ij} = \frac{1}{|N_i|}$ is the transition probability of the random walker to move from v_i to v_j in one step.
- MinorityAnt: an ant moves from any node v_i to $v_j \in N_i$ selected based on the lowest pheromone concentration. The pheromone of v_j is represented by p_j, which is increased when any ant visits the node v_j. Its transition probability is defined as $p_{ij} = 1 - \frac{p_j}{\sum_j p_j}$.

For both agents, they have different transition probabilities, consequently their abilities for exploring network are also unlike. The comparison between them will be shown in Sec. 4.

3.3 Representing User Activities

Here, a definition of active nodes is described. In P2P networks, especially content distribution issue, a user in the network has many activities such as downloading, sharing, searching, updating information or content and so on.

In our work, three activities of user (i), which are searching, downloading and making files or content available for other nodes, are considered based on only the quantity of them during a short period of time without a bandwidth utilization consideration. Their descriptions are presented as follows.

- Searching (a_{s_i}): to search for a quantity of desired files or content during a week, $a_{s_i} = [0, 100]$
- Downloading (a_{d_i}): to download a quantity of desired files or content during a week, $a_{d_i} = [0, 100]$
- Sharing (a_{f_i}): to share a quantity of files during a week, $a_{f_i} = [0, 100]$

The total activities of each node i is represented as a temperature, $T_{a_i} = \frac{a_{s_i} + a_{d_i} + a_{f_i}}{3}$, then $T_{a_i} = [0, 100]$ and a weight, w_i showing a level of its activities, consequently, $w_i = \frac{T_{a_i}}{100}$, then $w_i = [0, 1]$. To identify the status of node to be active or non-active one, a reference temperature T_{ref} is applied, if $T_{a_i} > T_{ref}$, then node i is an active one. Otherwise, node i is non-active. In our work, T_{ref} is defined to be 50.

3.4 *Identifying Location of Content Server by K-Means Clustering*

In our work, active nodes are clustered by K-means clustering algorithm to find locations of super nodes as centroids of active-node clusters.

Herein, the number of super nodes are fixed because the number of servers will be fixed based on the business cost or investment, then the approaches for clustering active nodes should support the fixed number of clusters. The clustering is done by minimizing response time (Euclidean distance), between active nodes and the corresponding cluster centroid (or server). Finally, the locations of super nodes are represented as the locations of content servers.

The super nodes should be located nearby not only the active nodes but also frequently doing activities, which is high w_i. It means that *high demand or contribution nodes should get high efficient service*, then a weighted mean is deployed to calculate the new centroids as follow.

$$(x_{c_k}, y_{c_k})_{new} = \frac{\sum_{i=1}^{s} w_i(x_i, y_i)}{\sum_{i=1}^{s} w_i} \tag{3}$$

where, $(x_{c_k}, y_{c_k})_{new}$ is the new centroid and s is the total number of active nodes in cluster k.

To identify the locations of servers by K-means algorithm, its processes are listed as follows.

Steps of Algorithm

1. Identify the number of K clusters
2. Initial value of K centroids
3. Start the network exploration by agents to look for active nodes
4. Calculate the distance between cluster centroid to each active node
5. Determine new centroids by weighted mean
6. Compute the distance of all active nodes to the new centroids
7. Assign each active node to the clusters based on the minimum distance
8. Represent locations of super nodes as centroids of each cluster
9. Calculate average response time by distance measurement between centroid and its members
10. Reach convergence then terminate, otherwise go to step 5

Finally, locations of servers and their active-node members will be presented. The servers will be located at the central point of clusters depended on w_i of each member and calculated response time of each member will be minimized. It will be summarised that each active node will be a member of the nearest cluster in order to get services from the nearest server with high QoS, fast response and high accessibility respectively.

4   Performance Evaluation

The objective pursued in this section is an empirical proof of concept to show that the response time of users will be low if the servers are located at the central point among them and the optimal locations of servers are found when the values of response time are converged.

For setting up the simulations, the P2PNetSim [30] is utilized to be a simulation tool. It is a distributed network simulator supporting a large scale P2P network simulations. It needs a network configuration, which is written in XML file format (for comprehensive details see [30]). It contains controller, simulator, and node parameters such as communication port, ID, IP address and node neighbors. To deal with behaviors of nodes, they can be implemented by JAVA programming and distributed over the network.

4.1   Simulation Results

4.1.1   Increasing Number of Servers

To conduct comparative simulations, a rectangular network (or grid) with the size of 20×20 was used, as well as the Random Walking was applied to search for active nodes in the network with the reference temperature $T_{ref} = 50$. In addition, the weighted mean was applied in K-means clustering algorithm and the number of servers was changed increasingly from 1 to 15.

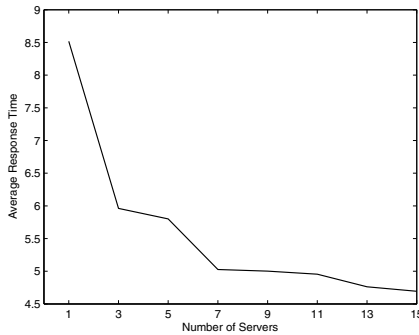**Fig. 5** Average response time comparison when increasing number of servers

Fig. 5 illustrates that the average response time of active users is around 8.5 if there is only one server. After increasing the number of servers to 3, 5, 7, 9, 11, 13 and 15, the average response time is decreased continuously until around 4.5 when fifteen servers are located. It concludes that the response time of active users

is depended on the number of servers. Content can be delivered to active users faster when increasing the number of servers because the active users have more chance to stay near the servers, then the objective of increasing number of servers is to provide high quality of service to active users and decrease the traffic in the network simultaneously.

4.1.2 Exploring the Network by Random Walking and MinorityAnt

In the network exploration, nodes are visited by agents, which are the Random Walking and the MinorityAnt. Both agents look for active nodes in order to group them by using K-means clustering algorithm. They travel around the network following their transition probability. The efficiency of agents to explore the network is necessary for our purpose. If the agents can explore the network fast then our algorithm will reach to the convergence of average response time rapidly too.

To conduct the simulations here, first the comparison between Random Walking and MinorityAnt for exploring the network on 20×20 of grid was used. The results are shown in Fig. 6(a) and Fig. 6(b) that the MinorityAnt can explore the network faster than Random Walking. For instance in Fig. 6(a) at $t = 1,000$, the MinorityAnt can visit all nodes in the network, in contrast the Random Walking can visit only around 65% of nodes in the network. Later, Fig. 6(b) presents the number of found nodes in each period of time, herein the period of time is 500. At $t = 500$, the MinorityAnt can find around 88% of nodes in the network but 50% of nodes are found by the Random Walking, only. Next period of time, at $t = 1,000$, the MinorityAnt can find all remained nodes, in contrast the Random Walking can find some of remained nodes until $t = 4,000$, then the Random Walking can find all nodes in the network.

In addition, increasing the size of network to 50×50, the results are shown in Fig. 6(c) and Fig. 6(d) that the MinorityAnt still explore the network faster than Random Walking. From these simulations, Fig. 7 concludes that the MinorityAnt can explore the network faster than the Random Walking around $3 - 5$ times, which is depended on the size of network. Secondly, the average response time comparison was simulated by using both Random walking and MinorityAnt. It was conducted on 20×20 of grid with $T_{ref} = 50$ that the number of servers was three and the weighed mean was applied in K-means clustering algorithm.

Fig. 8(a) shows the average response time of active users, which is around 6 when there are three servers located. The MinorityAnt can reach to the convergence of this value at $t = 30,000$, in contrast the Random Walking spends much time for reaching to the convergence until $t = 300,000$. To summary, the MinorityAnt can reach to the convergence faster than the Random Walking around 10 times.

In addition, increasing the size of network to 50×50 and fifteen super nodes, the result is presented in Fig. 8(b) that the Random Walking still reach to the convergence value of the average response time, around thirteen, slower than the MinorityAnt. It means that the searching ability of MinorityAnt is better than the Random Walking on not only the medium size of network 20×20 but also the large size of network 50×50.

(a) Searching for nodes on $20{\times}20$

(b) Average number of found nodes on $20{\times}20$

(c) Searching for nodes on $50{\times}50$

(d) Average number of found nodes on $50{\times}50$
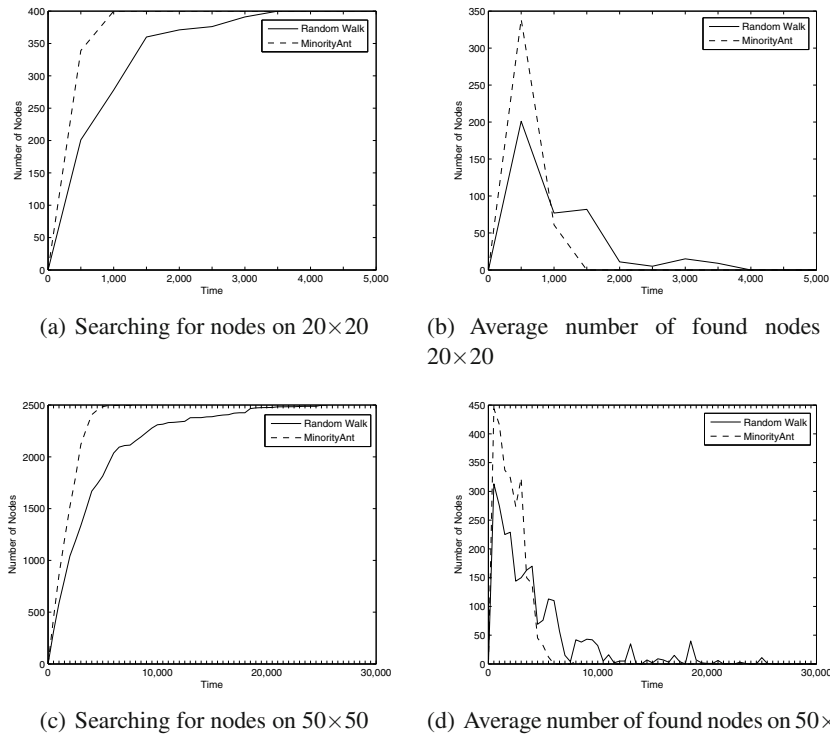
Fig. 6 Exploring network comparison between Random Walking and MinorityAnt
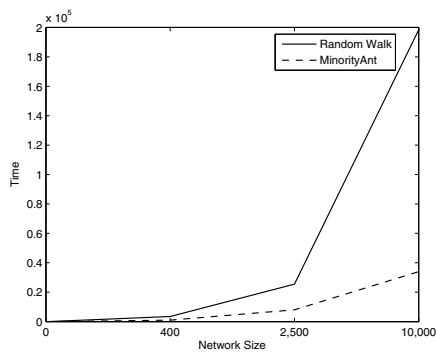
Fig. 7 Comparison between Random Walking and MinorityAnt on different network sizes

(a) Average response time comparison on 20×20

(b) Average response time comparison on 50×50
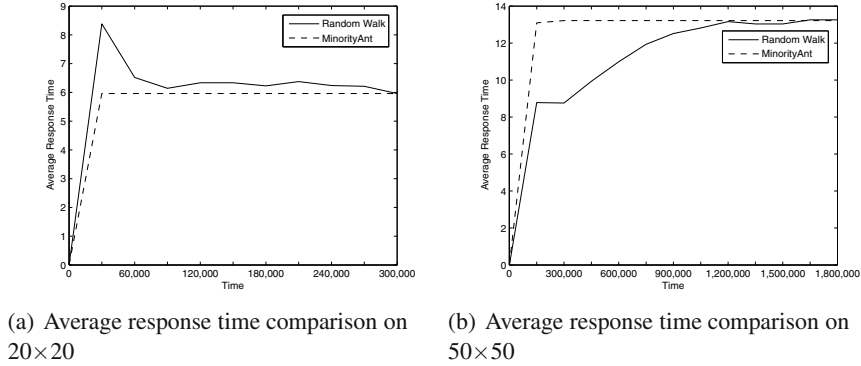
Fig. 8 Finding optimal locations of servers on different network sizes

5 Conclusion

As a consequence of user activities consideration, content are located not only nearby active nodes but also close up to active nodes who frequently do activities such as searching or downloading content and making files or content available to other nodes. Consequently, searching traffic is decreased and enable faster response to requesting nodes because the content are distributed closer to target nodes.

In this paper, the K-means clustering algorithm is applied. Active nodes are grouped by it and the content are located at nodes who are the centroids of active nodes clusters in order to increase a fast response to users. The results show that the average response time of active users will be decreased the number of content servers (or super nodes) is increased, as well as the convergence of response time values will be reached quickly if the MinorityAnt is used for exploring the network.

For future works, to increase the quality of service, three main factors shown in Fig. 1 are combined to identify the locations of content. The distributed content should be placed on nodes as follows:

- Nodes with a central position
- Nodes with high speed and low-latency network connections, which support the quality of service requirements
- Nodes, which close to those users, who frequently access content or do activities

It concludes that user activities consideration will be included in our PageRank calculation [14], which will be applied to optimise the delivery of content to target users.

References

1. Pathan, A.K., Buyya, R.: A Taxonomy and survey of content delivery networks. In: Technical report, GRIDS-TR-2007-4. Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia (2007),
   http://www.cloudbus.org/reports/CDN-Taxonomy.pdf

2. Akamai Technologies, Inc., `http://www.akamai.com`

3. YouTube website, `http://www.youtube.com`

4. Liu, Y., Guo, Y., Liang, C.: A survey on peer-to-peer video streaming systems. Journal of Peer-to-Peer Networking and Applications (2008)

5. Huang, Y., Fu, T.Z.J., Chiu, D.M., Lui, J.C.S., Huang, C.: Challenges, design and analysis of a large-scale p2p-vod system. ACM SIGCOMM Computer Communication Review 38 (2008)

6. Saroiu, S., Gummadi, K.P., Dunn, R.J., Gribble, S.D., Levy, H.M.: An analysis of Internet content delivery systems. ACM SIGOPS Operating Systems Review 36, 315–328 (2002)

7. Sripanidkulchai, K., Zhang, H.: Content location in Peer-to-Peer systems: exploiting locality. In: Web Content Delivery, Springer Series in Web Information Systems Engineering and Internet Technologies Book Series, vol. 2 (2005)

8. Napster website, `http://napster.com`

9. Freenet website, `http://freenetproject.org`

10. KaZaA website, `http://www.kazaa.com`

11. Shen, H.: An Efficient and adaptive decentralized file replication algorithm in P2P file sharing systems. IEEE Transactions on Parallel and Distributed Systems 21, 827–840 (2010)

12. Zhang, R., Hu, Y.C.: Assisted Peer-to-Peer search with partial indexing. IEEE Transactions on Parallel and Distributed Systems 18, 1146–1158 (2007)

13. Li, K., Zhou, W., Yu, S., Li, P.: Novel data management algorithms in peer-to-peer content distribution networks. In: Zhang, Z., Siekmann, J.H. (eds.) KSEM 2007. LNCS (LNAI), vol. 4798, pp. 538–543. Springer, Heidelberg (2007)

14. Sodsee, S., Kubek, M., Meesad, P., Unger, H.: Calculating PageRank in decentralised systems from random walks and network parameters (2010) (submitted)

15. Jesi, G.P., Montresor, A., Babaoglu, O.: Proximity-aware superpeer overlay topologies. IEEE Transactions on Network and Service Management 4, 74–83 (2007)

16. Zhu, Y., Hu, Y.: Enhancing search performance on Gnutella-like P2P systems. IEEE Transactions on Parallel and Distributed Systems 17, 1482–1495 (2006)

17. Joung, Y., Yang, L., Fang, C.: Keyword search in DHT-based peer-to-peer networks. IEEE Journal on Selected Areas in Communications 25, 46–61 (2007)

18. Bisnik, N., Abouzeid, A.A.: Optimizing random walk search algorithms in P2P networks. Computer Networks 51, 1499–1514 (2007)

19. Bui, M., Bernard, T., Sohier, D., Bui, A.: Random walks in distributed computing: A survey. In: Böhme, T., Larios Rosillo, V.M., Unger, H., Unger, H. (eds.) IICS 2004. LNCS, vol. 3473, pp. 1–14. Springer, Heidelberg (2006)

20. Zhong, M., Shen, K., Seiferas, J.: The convergence-guaranteed random walk and its applications in peer-to-peer networks. IEEE Transaction on Computers 57, 619–633 (2008)

21. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm intelligence: from natural to artificial systems. Oxford University Press, New York

22. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics, Part B 26, 29–41 (1996)

23. Tran, M., Tavanapong, W.: On using a cDN's infrastructure to improve file transfer among peers. In: Dalmau Royo, J., Hasegawa, G. (eds.) MMNS 2005. LNCS, vol. 3754, pp. 289–301. Springer, Heidelberg (2005)

24. Milgram, S.: The small world problem. Psychology Today, 60–67 (1967)

25. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web. In: Technical report, Stanford Digital Library Technologies Project (1998)
26. Schurmann, J.: Pattern classification. John Wiley & Sons, Singapore
27. Hammouda, K.M.: Distributed document clustering and cluster summarization in peer-to-peer environments. In: PhD thesis, University of Waterloo, Waterloo, Ontario, Canada (2007),
    `http://uwspace.uwaterloo.ca/bitstream/10012/3099/1/`
    `hammouda-phd-thesis.pdf`
28. Datta, S., Bhaduri, K., Giannella, C., Wolff, R., Kargupta, H.: Distributed data mining in Peer-to-Peer networks. IEEE Internet Computing 10, 18–26 (2006)
29. Unger, H., Wulff, M.: Towards a decentralized search engine for P2P-network communities. In: Proc. Eleventh Euromicro Conference on Parallel, Distributed and Network-Based Processing, pp. 492–499 (2003)
30. P2PNetSim, User's manual. JNC, Ahrensburg

A Note on Semi-steady States in Stochastic Cellular Automata

Thomas Böhme, Jens Schreyer, and Erika Škrabuľáková

Abstract. We consider stochastic cellular automata and use a theory developed by H. Peyton Young to analyze semi-steady states in such automata.

1 Introduction

Consider a rectangular array of pixels. For the sake of simplicity we assume periodic boundary conditions, i.e. we identify the upper and the lower boundary as well as the left and the right boundary. In other words, the array may be thought of as embedded into the surface of a torus. Each pixel can show two colors, say *black* and *white*, and for each pixel v, we fix a square neighborhood consisting of the eight adjacent pixels and v itself. We define a stochastic process on the set of all *black-white*-colorings. Initially, we chose a coloring uniformly at random. Then in every time step a pixel v is chosen uniformly at random. The pixel v is recolored with the color which appears more frequently in its neighborhood. This process has many steady states, i.e. colorings that remain unchanged regardless of what pixel is chosen. Some typical instances of such stable colorings are shown in Figures 1, 2 and 3. We consider a perturbed version of this stochastic process as follows. Let ε be a small positive number. Again, the initial coloring is chosen uniformly at random, and in each time step a pixel v is chosen uniformly at random. After choosing the pixel v, an unfair coin is tossed. It shows heads with probability $1 - \varepsilon$ and tail with probability ε. If the coin comes up heads, we proceed as in the unperturbed process, i.e. the pixel v is recolored with the color which appears more frequently

Thomas Böhme · Jens Schreyer
Technische Universität Ilmenau, PF 100565, 98684 Ilmenau, Germany
e-mail: thomas.boehme@tu-ilmenau.de

Erika Škrabuľáková
Department of Applied Mathematics and Informatics Faculty
of Mechanical Engineering Technical University of Košice Košice Slovakia
e-mail: erika.skrabulakova@tuke.sk

in its neighborhood. Otherwise, a neighbor w of v is chosen uniformly at random and v is recolored with w's color. The perturbed process has precisely two steady states, namely, every pixel is colored *black* or every pixel is colored *white*. Standard arguments from the theory of Markov chains show that the perturbed process converges for every $\varepsilon \in (0,1)$ with probability 1 to one of these uniform colorings (see Proposition 3).

About ten years ago, Frank Göring (Technische Universität Chemnitz, Germany `http://www.tu-chemnitz.de/mathematik/discrete/goering/`) and one of the authors of this note conducted a computer simulation of the two processes sketched above. The results of this simulation showed an interesting phenomenon. Even though the perturbed process converges with probability 1 to a uniform coloring, it tends to spend a significant amount of time in (or close to) the steady states of the unperturbed process. These steady states are the ones meant by the term semi-steady states in the title. To be more precise, the coloring of the array shows an evolution which may be described as follows. The random initial coloring appears typically as a uniform grey screen. Then the coloring rather quickly forms a pattern consisting of white and black spots (dubbed 'cow spots') similar to the one shown in Figure 3 on the right hand side. Later, these spots slowly merge and form a pattern consisting of a black and a white ribbon looped around the torus as shown on the left hand side of Figure 3 (dubbed 'stripes'). From this state on the process converges extremely slowly to a uniform coloring.

The main purpose of this note is to present a first step toward a mathematical analysis of this phenomenon. The main ingredient of this theory is the notion of 'stochastically stable states' of regularly perturbed Markov chains introduced by Young in [3] (see also [1]).

The paper is organized as follows. In section 2 we recall some basic facts about Markov chains and explain the main theorem of Young's theory. In section 3 we give a definition of the notion of a stochastic cellular automaton, apply Young's theory of stochastically stable states to these automata, and give a more detailed analysis of a special instance of a stochastic cellular automaton.

We wish to thank our colleagues Stephan Brandt and Erhard Hexel (Technische Universität Ilmenau) for many stimulating discussions.

2   Some Facts about Markov Chains

A *discrete time stochastic process* $(Z_i)_{i=0,1,2,\ldots}$ is a sequence of random variables that all take values in some nonempty set M called the *state space* of the stochastic process. A discrete time stochastic process $(Z_i)_{i=0,1,2,\ldots}$ with finite or countable state space M is called a *Markov chain* if for all $t \in \{0,1,2,\ldots\}$ and all finite sequences $(m_0,\ldots,m_{t+1})$ of elements of M

$$\text{Prob}(Z_{t+1} = m_{t+1} \mid Z_t = m_t,\ldots,Z_0 = m_0) = \text{Prob}(Z_{t+1} = m_{t+1} \mid Z_t = m_t). \quad (1)$$

A Markov chain $(Z_i)_{i=0,1,2,\ldots}$ is called *stationary* (or *time homogeneous*) if for all $t \in \{0,1,2,\ldots\}$ and all $m, m' \in M$

$$\mathrm{Prob}(Z_{t+2} = m' \mid Z_{t+1} = m) = \mathrm{Prob}(Z_{t+1} = m' \mid Z_t = m). \qquad (2)$$

For a stationary Markov chain $(Z_i)_{i=0,1,2,\ldots}$ with state space M, we define a function $p : M \times M \to [0,1]$ by $p(m,m') = \mathrm{Prob}(Z_1 = m' \mid Z_0 = m)$ for all $m, m' \in M$. The number $p(m,m')$ is called the *transition probability* from m to m', and we call p the *transition probability function* of the Markov chain. Note that, because of stationarity, $\mathrm{Prob}(Z_{t+1} = m' \mid Z_t = m) = p(m,m')$ for all $t \in \{0,1,2,\ldots\}$ and all $m, m' \in M$. A state m' is called *reachable* from a state $m \in M$ if there is an integer n such that $\mathrm{Prob}(Z_n = m' \mid Z_0 = m) > 0$. Again by stationarity, it follows that $\mathrm{Prob}(Z_n = m' \mid Z_0 = m) > 0$ if and only if $\mathrm{Prob}(Z_{n+k} = m' \mid Z_k = m) > 0$ for all positive integers k.

Since all Markov chains considered in this note are stationary and have a finite state space, we will henceforth simply speak of Markov chains whenever we mean a stationary Markov chain with finite state space.

Consider a Markov chain with state space M and transition probability function p. We define the *transition graph D* as the directed graph D with vertex set M and edge set $E(D) = \{(m,m') \mid p(m,m') > 0\}$. A nonempty subset C of the state space M of a stationary Markov chain is a *recurrent communication class* if the following conditions are fulfilled.

- Any state $m' \in C$ is reachable from any other state $m \in C$.
- For all $m \in C$ and all $m' \in M \setminus C$, $p(m,m') = 0$.

A state $m \in M$ is said to be *absorbing* if $\{m\}$ is a recurrent communication class. Obviously, a state $m \in M$ is absorbing if and only if m is a sink in the transition graph. We say that a Markov chain $(Z_i)_{i=0,1,2,\ldots}$ *reaches* a state $m' \in M$ from starting state $m \in M$ in n steps if $Z_0 = m$ and $Z_n = m'$.

It is not hard to see that any state m of a Markov chain $(Z_i)_{i=0,1,2,\ldots}$ is either contained in a recurrent communication class, or there is a state m' contained in a recurrent communication class that can be reached from m. Using this observation one can easily prove the following proposition.

Proposition 1. *The probability that a Markov chain will reach a state in a recurrent communication class in finitely many steps is 1 for any starting state.*

A *probability distribution* on a finite set X is a function $v : X \to [0,1]$ such that $\sum_{x \in X} v(x) = 1$. A probability distribution v on the state space M of a Markov chain with transition probability function p is called a *stationary distribution* of the Markov chain if $v(m) = \sum_{x \in M} v(x) p(x,m)$ for all $m \in M$.

A Markov chain is called *irreducible* if its transition graph is strongly connected, i.e. for any two states $m, m' \in M$ there is a directed path from m to m' and a directed path from m' to m. Clearly, a Markov chain is irreducible if and only if its state space is a recurrent communication class. The *period* of a state $m \in M$ is the greatest common divisor of all integers n for which $\mathrm{Prob}(Z_n = m \mid Z_0 = m) > 0$. A state is

called *aperiodic* if its period is 1. It is easy to see that a state $m \in M$ is aperiodic if $p(m,m) > 0$. A Markov chain is called *aperiodic* if all its states are aperiodic. It is not hard to see that any two states in an irreducible Markov chain have the same period. An immediate consequence of this is, that whenever $p(m,m) > 0$ for some state m of an irreducible Markov chain, then the Markov chain is aperiodic. The following theorem may be found in any standard text on Markov chains.

Theorem 1. *If a Markov chain is irreducible and aperiodic, then there is a uniquely determined stationary distribution of the Markov chain.*

It can be shown that for any state $m \in M$ of an irreducible and aperiodic Markov chain $(Z_i)_{i=0,1,2,\ldots}$ with state space M and stationary distribution π and for any positive real ε

$$\lim_{n \to \infty} \text{Prob}(|\frac{|\{t \in \{0,1,2,\ldots\} \mid Z_t = m\}|}{n+1} - \pi(m)| > \varepsilon) = 0. \tag{3}$$

Equation 3 says that for large values of n the relative number $\frac{|\{t \in \{0,1,2,\ldots\} \mid Z_t = m\}|}{n+1}$ of time steps t in which $Z_t = m$ approaches $\pi(m)$.

In [2] (p.177, Lemma 3.1) it is shown that the stationary distribution π of an aperiodic irreducible Markov chain can be computed in terms of the transition graph G. In order to describe this method, we need the following definition. Let x be a vertex of a strongly connected directed graph $D = (V, E)$. An *x-tree* in D is a subgraph T of D that satisfies the following conditions

- T contains precisely one directed edge starting at y for any vertex y of $V \setminus \{x\}$.
- T does not contain any directed cycles.

Theorem 2 ([2], Lemma 3.1). *Consider an aperiodic irreducible Markov chain with state space M, transition graph G, and stationary distribution π. For an m-tree T in G, let $\mu(T) = \prod_{(x,y) \in E(T)} p(x,y)$, and put $Q_m = \sum_{T \in \mathscr{G}(m)} \mu(T)$ where $E(T)$ denotes the edge set of T and $\mathscr{G}(m)$ the set of all m-trees in G. Then for all $m \in M$*

$$\pi(m) = \frac{Q_m}{\sum_{x \in M} Q_x}. \tag{4}$$

The following definitions are taken from the appendix of [3]. Consider a Markov chain $(Z_i)_{i=0,1,2,\ldots}$ with state space M and transition probability function p. Let $\{(Z_i^\varepsilon)_{i=0,1,2,\ldots} \mid \varepsilon \in (0,1)\}$ be a family of Markov chains with the same state space M, and let p^ε denote the transition probability function of the Markov chain $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$. Then the family of Markov chains $(Z_i^\varepsilon)_{i=0,1,2,\ldots} \mid \varepsilon \in (0,1)\}$ is called a *regular perturbation* of the Markov chain $(Z_i)_{i=0,1,2,\ldots}$ if the following conditions hold for all states $a, b \in M$

- The Markov chain $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$ is aperiodic and irreducible for all $\varepsilon \in (0,1)$.
- $\lim_{\varepsilon \to 0} p^\varepsilon(a,b) = p(a,b)$.
- If $p^\varepsilon(a,b) > 0$ for some $\varepsilon \in (0,1)$, then there is a number $r(a,b) = r \geq 0$ such that $0 < \lim_{\varepsilon \to 0} \frac{p^\varepsilon(a,b)}{\varepsilon^r} < \infty$.

Note that the numbers $r(a,b)$ are uniquely determined if they exist. Furthermore, it is easy to see that $r(a,b) = 0$ if and only if $p(a,b) > 0$.

Theorem 3 ([3],Theorem 4). *Let $\{(Z_i^\varepsilon)_{i \in \{0,1,2,...\}} \mid \varepsilon \in (0,1)\}$ be a regular perturbation of a Markov chain $(Z_i)_{i \in \{0,1,2,...\}}$ with state space M. For $\varepsilon \in (0,1)$ let π^ε denote the stationary probability distribution of the Markov chain $(Z_i^\varepsilon)_{\in \{0,1,2,...\}}$. Define a directed graph H with vertex set M and edge set $E(H) = \{(m,m') \mid p^\varepsilon(m,m') > 0 \text{ for some } \varepsilon\}$. For $m \in M$, let $\mathscr{H}(m)$ denote the set of all m-trees in H, and let $\gamma(m,T) = \sum_{(x,y) \in E(T)} r(x,y)$ for $T \in \mathscr{H}(m)$. Finally, put $\gamma(m) = \min_{T \in \mathscr{H}(m)} \gamma(m,T)$ and $\gamma^* = \min_{m \in M} \gamma(m)$. Then the limit $\lim_{\varepsilon \to 0} \pi^\varepsilon = \pi$ exists, π is a stationary probability distribution of $(Z_i)_{i \in \{0,1,2,...\}}$, and $\pi(m) > 0$ if and only if $\gamma(m) = \gamma^*$.*

The numbers $r(a,b)$ are called *resistances* for the transitions from state a to state b. Moreover, the states $m \in M$ for which the probability $\pi(m)$ for the limit distribution π is strictly greater than 0 are called *stochastically stable*.

3   Stochastic Cellular Automata

Let G be a finite undirected graph with vertex set V and edge set E, and let C be a nonempty finite set. Let C^V denote the set of all mappings from V into C, called *C-colorings* (of G). The graph G is called the *underlying graph* of the stochastic cellular automaton. For $v \in V$ let $N(v) = \{w \in V \mid \{v,w\} \in E\}$. We define a *stochastic cellular automaton* on G as a Markov chain $(X_t)_{t=0,1,2,...}$ with state space C^V that satisfies the following condition. For all $v \in V$, all colorings $c, c' \in C^V$, and all nonnegative integers t

$$\text{Prob}(X_{t+1}(v) = c_v \mid X_t = c) = \text{Prob}(X_{t+1}(v) = c_v \mid X_t = c') \qquad (5)$$

as long as $c(x) = c'(x)$ for all $x \in N(v) \cup \{v\}$.

Next, we define a concrete stochastic cellular automaton $(X_i)_{i=0,1,2,...}$ on a graph G with state space C^V.

1. Let X_0 be a C-coloring of G drawn uniformly at random from C^V.
2. For $t \geq 1$ let the coloring X_t be obtained from the coloring X_{t-1} as follows. Pick a vertex $v \in V$ uniformly at random, let s be the number of vertices $w \in N(v)$ such that $X_{t-1}(w) = X_{t-1}(v)$, and let $r = |N(v)| - s$. If $s \geq r$, we let $X_t(v) = X_{t-1}(v)$. Otherwise, the color of x will be swapped, i.e. $X_t(v) = white$ if $X_{t-1}(v) = black$, and $X_t(v) = black$ if $X_{t-1}(v) = white$. All vertices $w \in V \setminus \{v\}$ retain their old color, i.e. $X_t(w) = X_{t-1}(w)$.

Let b denote the C-coloring with $B(v) = black$ for all $v \in V$ and w the C-coloring with $W(v) = white$ for all $v \in V$. It is easy to see that $\{b\}$ and $\{w\}$ are recurrent communication classes of $(X_i)_{i=0,1,2,...}$. Depending on the underlying graph there may be much more recurrent communication classes than these two. To illustrate this fact, we consider the following examples.

Example 1. If the graph G is a cycle C_n then a coloring $c \in C^V$ is an absorbing state if and only if every vertex has a neighbor of the same color. That means the cycle is divided into monochromatic segments each of which contains at least 2 vertices.

Example 2. Let $[n]$ denote the set of the first n nonnegative integers, i.e. $[n] = \{0, 1, 2, \ldots, n-1\}$, and let $H_{n,m}$ be the graph on the vertex set $[n] \times [m]$ where two vertices (a, b) and (c, d) are adjacent if and only if $|a - c| \in \{0, 1, n-1\}$, $|b - d| \in \{0, 1, m-1\}$, and $(a, b) \neq (c, d)$.

Intuitively, $H_{n,m}$ is obtained from a (special) quadrangulation of the torus by adding both diagonals to every facial 4-cycle. (This is the cellular automaton considered in the introduction.) The following figures show some examples of colorings of $H_{n,m}$ that form absorbing states in the corresponding Markov chain $(X_i)_{i=0,1,2,\ldots}$. Note that the vertices on the boundary of the picture are connected to corresponding vertices on the opposite boundary and that the diagonals are not shown in the figures.
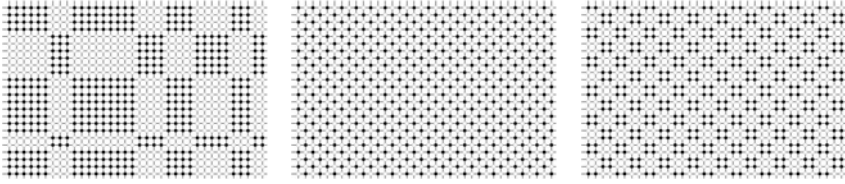
Fig. 1 Chessboard like colorings that are absorbing states of the Markov chain
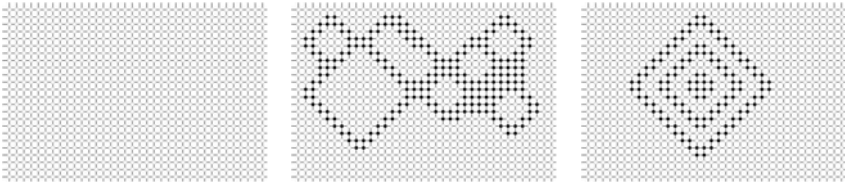
Fig. 2 Circles and a monochromatic coloring

For every coloring c let B_c of G be the subgraph of G induced by the set of black vertices and $W_c(G)$ the subgraph induced by the set of white vertices. Now it is easy to see, that a coloring c of $H_{n,m}$ constitutes an absorbing state of the Markov chain $(X_i)_{i=0,1,2,\ldots}$ if and only if the minimum degree of $B_c(G)$ and $W_c(G)$ is at least 4. It remains to investigate, whether there can be more recurrent communication classes, than only the 1-element classes formed by absorbing states. The answer is given by the following proposition.
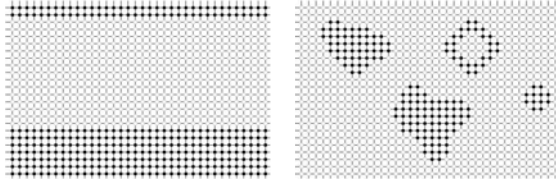
Fig. 3 Stripes and cow spots

Proposition 2. *For any underlying graph G the transition graph of the Markov chain $(X_i)_{i=0,1,...}$ is acyclic.*

Proof. Let c denote a state of the Markov chain that means a C-coloring of G. We call an edge $(u,v) \in V$ *monochromatic* in c if $c(v) = c(w)$, otherwise it is *heterochromatic*. By $m(c)$ we denote the number of monochromatic edges of G colored by c. A transition from c_1 to a state $c_2 \neq c_1$ is only possible if the colorings differ in the color of exactly one vertex v. Moreover, by the definition of the automaton there must be more vertices in $N(v)$ colored differently from v in c_1 than colored like v. That means, if the color of v is changed, the number of monochromatic edges increases. This shows that for every possible transition c_1 to c_2 it holds $m(c_1) < m(c_2)$. If there was a cycle $c_1, ... c_k, c_1$ in the transition graph, this would imply $m(c_1) < ... < m(c_k) < m(c_1)$, a contradiction. □

The proposition implies that every recurrent communication class of the process is of cardinality 1. That means the colorings will eventually converge to a stable coloring.

Let ε be a small positive number and consider the following perturbation of the automaton defined above.

1. Let Y_0^ε be a C-coloring of G drawn uniformly at random from C^V.
2. For $t \geq 1$ let the coloring Y_t^ε be obtained from the coloring Y_{t-1}^ε as follows. Pick a vertex $v \in V$ uniformly at random. Let s be the number of vertices $w \in N(v)$ such that $Y_{t-1}^\varepsilon(w) = Y_{t-1}^\varepsilon(v)$, and let $r = |N(v)| - s$.

   - With probability ε, choose a random neighbor $w \in N(v)$ and set $Y_t^\varepsilon(v) = Y_{t-1}^\varepsilon(w)$.
   - With probability $1 - \varepsilon$, we let $Y_t^\varepsilon(v) = Y_{t-1}^\varepsilon(v)$ if $s \geq r$, and swap the color of v if $s < r$.

   All vertices $w \in V \setminus \{v\}$ retain their old color.

Proposition 3. *Let G be a connected graph and $\varepsilon \in (0,1)$. Then $\{b\}$ and $\{w\}$ are the only recurrent communication classes of the Markov chain $(Y_i^\varepsilon)_{i=0,1,2,...}$.*

Proof. It is easily to be seen that $\{b\}$ and $\{w\}$ are recurrent communication classes of the process $(Y_i^\varepsilon)_{i=0,1,2,...}$. Let c be a C-coloring of G such that there are two vertices $v, w \in V$ with different colors, say $c(v) = white$ and $c(w) = black$. Since G is connected there is an edge $\{v', w'\} \in E$ such that $c(v') = white$ and $c(w') = black$. Assume that for some $t \geq 1$, $Y_{t-1}^\varepsilon = c$. In period t there is a positive probability that v' will be picked and a positive probability that the color of the neighbor $w' \in N(v')$ will be chosen for the color of v' for the next round, i.e. with positive probability the number of black vertices is increased by precisely one. It follows that every recurrent communication class of the Markov chain $(Y_i^\varepsilon)_{i=0,1,2,...}$ containing c also contains b. Since b is already in the one element recurrent communication class $\{b\}$, this implies that no state c that has vertices of both colors can be in any recurrent communication class of $(Y_i^\varepsilon)_{i=0,1,2,...}$, which proves the claim. □

Simulating the perturbed Markov chain it turned out that some of the steady states of the unperturbed chain are reached quickly and kept for a long time, until finally a uniform coloring was attained. We wish to use Young's theory of stochastic stability (Theorem 3) to give a mathematical explanation of this phenomenon. To this end we change the Markov chain once more. When one of the states b or s is reached, we choose a new coloring uniformly at random for the next round. This corresponds to a restart of the experiment. It should be noted that the resulting Markov chain $(Z_i^\varepsilon)_{i=0,1,2,...}$ is not a cellular automaton. The Markov chain $(Z_i^\varepsilon)_{i=0,1,2,...}$ has identical transition properties to the chain $(Y_i^\varepsilon)_{i=0,1,2,...}$ apart from the transitions from b and s to all other states. Moreover, let $(Z_i)_{i=0,1,2,...}$ be the Markov chain which is derived from $(X_i)_{i=0,1,2,...}$ in the same way by adding transitions from b and s to all states with equal probability.

Proposition 4. *The recurrent communication classes of $(Z_i)_{i=0,1,2,...}$ are the same as those of $(X_i)_{i=0,1,2,...}$ without the classes $\{b\}$ and $\{w\}$ if $\{b\}$ and $\{w\}$ are not the only recurrent communication classes. Otherwise the single recurrent communication class is the whole state space C^V, i.e. the Markov chain is irreducible.*

Proof. Let c be any state of C that is not an absorbing state of $(X_i)_{i=0,1,2,...}$. Assume c is an element of a recurring communication class. As in the proof of Proposition 3 it follows, that b belongs to the same recurrent communication class. Because every transition is possible from b in the Markov chain $(Z_i)_{i=0,1,2,...}$, every state C^V belongs to the same recurrent communication class. Hence, there is no absorbing state of the Markov chain $(Z_i)_{i=0,1,2,...}$. If there are absorbing states of $(X_i)_{i=0,1,2,...}$ apart from b and w, this is a contradiction because these states stay absorbing in the new Markov chain. □

Proposition 5. *The Markov chain $(Z_i^\varepsilon)_{i=0,1,2,...}$ is a regular perturbation of the Markov chain $(Z_i)_{i=0,1,2,...}$. Moreover, the resistance $r(c_1, c_2)$ of a possible transition from the state c_1 to the state c_2 in the perturbed Markov chain $(Z_i^\varepsilon)_{i=0,1,2,...}$ (i.e. $p^\varepsilon(c_1, c_2) > 0$) is either 0 if the transition is possible in the unperturbed Markov chain $(Z_i)_{i=0,1,2,...}$, too, or 1 if this is not the case.*

Proof. It is easy to see that for the transition probability functions p and p^ε of the Markov chains $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$ and $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$, respectively, it holds $\lim_{\varepsilon \to 0} p^\varepsilon = p$. Because every transition that is possible in $(Y_i^\varepsilon)_{i=0,1,2,\ldots}$ is also possible in $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$, it follows from the proof of Proposition 3, that from every state c the state b is reachable and also from b every state c is reachable. Hence, $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$ is an irreducible Markov chain. It is also aperiodic because the transition graph contains loops (e.g. $p^\varepsilon(b,b) > 0$). Let c_1 and c_2 be 2 states such that $p^\varepsilon(c_1,c_2) > 0$. If $p(c_1,c_2) > 0$ then we have $0 < (1-\varepsilon)p(c_1,c_2) < 1$. The inequality still holds for $\varepsilon \to 0$. Hence, $r(c_1,c_2) = 0$. If $p(c_1,c_2) = 0$ then the transition from c_1 to c_2 is only possible in the perturbed Markov chain if the following happens. First the only vertex v in which both colorings differ is chosen. Then it is decided that the new color of v only depends on the color of a randomly chosen neighbor w of v, and finally, the color $c_1(w)$ must be different from $c_1(v)$. Hence, $1/|V| \cdot \varepsilon \cdot 1/|N(v)| < p(c_1,c_2) < 1/|V| \cdot \varepsilon \cdot 1/2$. Dividing the inequality by ε and taking the limit for $\varepsilon \to 0$ shows that $r(c_1,c_2) = 1$. □

With the propositions 2,4 and 5 we can deduce the following theorem as a direct consequence of Theorem 3.

Theorem 4. *For an arbitrary underlying graph G the stationary distribution π^ε of $(Z_i^\varepsilon)_{i=1,\ldots}$ has the property that every state c with $\pi^\varepsilon(c) > k$ for some constant k independent of ε is an absorbing state of the unperturbed Markov chain $(Z_i^\varepsilon)_{i=1,\ldots}$.*

This theorem exactly explains the observation in the experiments. However, it does not answer the question, which absorbing states occur with positive probability. For the cycle graphs in example 1 it can be shown, that all absorbing states of the unperturbed Markov chain $(Z_i)_{i=0,1,2,\ldots}$ are stochastically stable.

Theorem 5. *Let the underlying graph G be a cycle C_n and π^ε be the unique stationary distribution of the Markov chain $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$ and $\pi = \lim_{\varepsilon \to 0} \pi^\varepsilon$. Then π is a stationary distribution of $(Z_i)_{i=0,1,2,\ldots}$ and $\pi(c) > 0$ precisely if c is an absorbing state of $(Z_i)_{i=0,1,2,\ldots}$.*

Proof. From the previous theorem we already know that only absorbing states of $(Z_i)_{i=0,1,2,\ldots}$ can be stochastically stable. It remains to show that for **every** absorbing state c the condition $\pi(c) > 0$ is true.

Let S denote the set of absorbing states in the Markov chain $(Z_i)_{i=0,1,2,\ldots}$ and $s = |S|$. Moreover, we consider the transition graphs D and D^ε of The Markov chains $(Z_i)_{i=0,1,2,\ldots}$ and $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$, respectively. Let $E_0 := E(D)$ and $E_1 := E(D^\varepsilon) \setminus E_0$. The resistance $r(c,c')$ of a transition from state c to c' in the Markov chain $(Z_i^\varepsilon)_{i=0,1,2,\ldots}$ equals 0 if $(c,c') \in E_0$ and 1 if $(c,c') \in E_1$.

Claim: In the transition graph D^ε of the Markov chain there is no directed cycle containing an edge of E_0, unless the cycle contains the state b or w.

Proof. To see this, consider a transition from the state c to the state c', where $c, c' \notin \{b, w\}$. If the edge $(c, c') \in E_0$, then there is a vertex v colored differently from its neighbors in c and the color of which is changed in c'. Hence, the number of components of at least one of the subgraphs $B_c(C_n)$ and $W_c(B_n)$ is decreasing by one. If $(c, c') \in E_1$, then in c' a color of a vertex is changed that has a black and a white neighbor in c. Hence, the number of components of the black and the white subgraph remain the same. This shows that for a sequence of transitions the number of the components of the white and the black subgraph is not increasing, and it is strictly decreasing if a transition of E_0 is involved, unless the sequence contains the state b or w. Hence, a cycle in the transition graph without the states b and w is only possible if all edges belong to E_1 which proves the claim.   □

Now consider any state $c \in C^V$ and a c-tree T which is a subgraph of D^ε. The edge set of T contains one outgoing edge for each state apart from c. Since there is no outgoing edge of an absorbing state in $(Z_i)_{i=0,1,2,\ldots}$, the edge set must contain at least s edges of resistance 1 if $c \notin S$ and $s - 1$ edges of resistance 1 if $c \in S$. Hence, we have

$$\gamma(c, T) = \sum_{e \in E(T)} r(e) \geq \begin{cases} s & \text{if } c \notin S \\ s - 1 & \text{if } c \in S \end{cases}$$

Since this holds for every c-tree T it is also true for the minimal one. Therefore, we have

$$\gamma(c) \geq \begin{cases} s & \text{if } c \notin S \\ s - 1 & \text{if } c \in S \end{cases}.$$

This implies, that $\gamma^* \geq s - 1$. To prove the statement of our theorem we have to show that $\gamma^* = \gamma(c)$ for all $c \in S$. By Theorem 3 it is sufficient to prove that for every absorbing state $c \in S$ there is a c-tree T such that $\gamma(c, T) = s - 1$. We choose the edge set $E(T)$ in the following way. First choose an arbitrary outgoing edge $(c', c'') \in E_0$ for all states $c' \in V \setminus S \setminus \{b, w\}$. Such edges do exist, because all the considered states c' are not absorbing. The resulting graph T has $s + 2$ sinks which are exactly the states in $S \cup \{b, w\}$. Moreover, the total resistance of T is 0 since only edges in E_0 have been chosen, and by the claim the graph T is acyclic. The resistance is still 0 when the states b and w are directly connected to c by the edges (b, c) and (w, c) both of which belong to E_0. Both edges cannot close a directed cycle, because c is a sink. Now the graph T consists of s components, rooted at the absorbing states in S and all components are directed towards the roots. For every absorbing state c' where one component of $B_{c'}$ or $W_{c'}$ is of cardinality 2, there is a transition $(c' c'') \in E_1$ where c'' is not absorbing. Add one of those edges for each such state c'. The resulting graph T is still acyclic (since the edge starting from c'' is in E_0 and each component is directed towards the root. Now we iteratively do the following. We choose a component T' of T not containing c in such a way that for the root c' of T' the smallest component of $B_{c'}$ and $W_{c'}$ has minimum cardinality. W.l.o.g. let such a component of cardinality $k \geq 3$ belong to $B_{c'}$. There is a transition (c', c'') in E_1 that decreases the number of elements of the component to $k - 1 \geq 2$.

Since all components of $B_{c''}$ and $W_{c''}$ have cardinality at least 2, c'' is an absorbing state. Moreover, since the smallest component of $B_{c''}$ has cardinality $k - 1$ by the choice of c' the state c'' must belong to the component of T containing c. Adding the edge $(c'c'')$ to the edge set of T reduces the number of components of T and retains the property that all components are directed towards the root. After finitely many iterations we get a c-tree T, that contains exactly $s - 1$ edges of E_1, one for each state $c' \in S \setminus \{c\}$. Hence, $\gamma(c, T) = s - 1$ which completes the proof of the theorem. □

Let us eventually compute the probability that the Markov chain $(Z_i^\varepsilon)_{i=0,1,...}$ leaves an absorbing state c of the unperturbed Markov chain. Such a transition is only possible if the following happens. A vertex x incident to at least one heterochromatic edge is chosen, then with probability ε it is decided that only the color of one neighbor y of x determines the new color of x. Finally, the randomly chosen neighbor y must have a color different from x. Now it is easy to check, that the resulting probability $p = \varepsilon \cdot \frac{h(c)}{e}$ where e is the number of edges of G and $h(c)$ denotes the number of heterochromatic edges. The number of rounds that the perturbed Markov chain remains in such a state is then geometrically distributed with parameter $1 - p$. Hence, the expected number of further rounds, that the chain remains in the state is $\frac{1-p}{p} = \frac{1}{\varepsilon} \cdot \frac{e}{h(c)} - 1$. So the mean residence time of the state is $\frac{1}{\varepsilon} \cdot \frac{e}{h(c)}$ rounds. For the graph $H[n, n]$ we have $e = 4n^2$. For the chessboard coloring c_1 of $H[n, n]$ (see the middle part of Figure 1)this implies $h(c_1) = 2n^2$. Hence, the mean residence time of the chessboard coloring is $2/\varepsilon$. On the other hand, for a stripe coloring c_2 of $H[n, n]$ the number of heterochromatic edges is $6n$ which yields a mean residence time of $\frac{2n}{3\varepsilon}$. This is an indication for the observation that stripe like structures appear much longer in the experiment than structures like the chess board.

References

1. Foster, D.P., Young, H.P.: Stochastic Evolutionary Game Dynamics. Theoretical Population Biology 38, 219–232 (1990)
2. Freidlin, M., Wentzell, A.: Random Perturbations of Dynamical Systems. Springer, New York (1984)
3. Young, H.P.: The Evolution of Conventions. Econometrica 61(1), 57–84 (1993)

Search Word Extraction Using Extended PageRank Calculations

Mario Kubek and Herwig Unger

Abstract. This paper describes a new method to determine characteristic terms from texts by weighting them using extended PageRank calculations. Additionally, this method clusters found semantic term relations to assign each term a level of specifity to be able to distinguish between general and specific terms. This way, it is also possible to differentiate between terms of different semantic orientations in the same specifity level. In the experiments, it is shown which terms can be used for the automatic retrieval of semantically similar documents from large corpora like the World Wide Web through automatic query formulation. The selection of query terms of a different specifity level is also a useful instrument in interactive document retrieval to express the intended similarity of documents to be found. An added advantage of this method is, that it does not rely on third-party datasets and works on single texts.

1 Introduction and Motivation

The selection of characteristic and discriminating terms in texts through weights, often referred to as keyword extraction or terminology extraction, plays an important role in text mining and information retrieval. The popular TF-IDF measure [1] can be used to assign terms a weight in a document corpus depending on how often it occurs in a specific document and in the whole document corpus. A term will get a high weight, when it often occurs in one document, but less often in other documents in the corpus. However, this measure cannot be used when there is no corpus available and just a single document needs to be analysed and it does not take into account the relations between the terms in the text. Another measure is the keyword density which determines the percentage of times a term occurs in a specific document compared to the total number of terms in this document. This simple approach works for single documents but assigns term weights without considering

Mario Kubek · Herwig Unger
Faculty of Mathematics and Computer Science, FernUniversity in Hagen
e-mail: `{mario.kubek,herwig.unger}@fernuni-hagen.de`

term relevancy. Moreover, terms will receive a high weight just because they appear very often in a document which is a major drawback as stopwords, general terms and a maybe high number of maliciously placed terms in texts will be illegitimately granted high importance. Another approach from statistical text analysis to find discriminating terms is called difference analysis [2]. Terms in a text are assigned a weight according to the comparison of the probability of their occurrence in the text and their occurrence in a general language reference corpus. Terms that occur above chance are assigned a high weight as they are likely technical terms. If such a reference corpus is not available, this method cannot be used.

Under the assumption that a local weight for terms in single texts needs to be determined it is therefore sensible to consider the semantic relations between terms in order to determine their importance. These relations can be obtained using statistical co-occurrence analysis and represented as a graph of connected nodes. This paper introduces a method to obtain characteristic terms in texts based on the text clustering technique presented in [3]. This clustering technique iteratively determines the weight of terms in co-occurrence graphs using PageRank calculations [4] and rules out terms that have a higher PageRank than their neighboring terms. This way, the co-occurrence graphs are separated into clusters of topically homogeneous terms. Herein, it is discussed which of these ruled out terms are suitable candidates as search words to find similar documents. Additionally, the originally used PageRank calculation is extended by taking into account the strength of the terms' semantic relations. Similar approaches to determine keywords of documents have been presented in [5] and [6] that rely on semantic graphs gained from synsets from WordNet. The method presented herein, however, deals with the co-occurrence graph of a text as it is and does not rely on third-party datasets.

Therefore, the paper is organized as follows: the next section explains the methodology used. Then, the algorithm to determine the weights of terms from text documents is presented. Section three focuses on the conducted experiments for automatic document retrieval from the World Wide Web based on the results of this method. Section four concludes the paper and provides a look at options to employ this algorithm in solutions for automatic taxonomy extraction.

2   Methodology

In this section, the algorithm to gain characteristic terms from texts through clustering of co-occurrence graphs based on extended PageRank calculations is presented. For this purpose, the new approach to take into account directed co-occurrence significances in order to determine a term's importance is explained.

2.1   Preliminary Considerations

In [3] a text clustering approach using PageRank calculations has been introduced. This algorithm iteratively rules out terms in co-occurrence graphs that have a higher

PageRank than their neighboring terms. Because these graphs are scale-free and have the small-world property the deletion of such terms leads to a separation of semantically related components. It was shown that the gained clusters contain terms of a high mean term-term-similarity. It was also found that removed terms with a high PageRank are likely to be characteristic for the respective document. However, the original PageRank formula only takes the structure of a graph into account to determine a node's importance. It is therefore sensible to extend it by employing the significance values of co-occurrences $Sig(t_i, t_j)$ to also consider the strength of the found semantic term relations. The idea to also regard the bandwidth of links between nodes for the PageRank calculation has been elaborated on in detail in [7]. The extended PageRank formula to calculate the PageRanks for all terms t_i in a co-occurrence graph is given in equation 1:

$$PR_{t_i} = (1 - \eta) + \eta \sum_{t_j \in J} \frac{PR_{t_j} * Sig(t_j, t_i)}{|N_{t_j}|}, \tag{1}$$

Well known measures to gain co-occurrence significance values $Sig(t_i, t_j)$ are for instance the mutual information measure [8], the Dice and Jaccard coefficients [8], the poisson collocation measure [9] and the log-likelihood ratio [10]. While these measures return the same value for the relation of a term A to another term B and vice versa, an undirected relation of both terms often does not represent real-life relationships very well, e.g. an Audi is a German car but not every German car is an Audi. Therefore, it is sensible to deal with directed relations of terms. To measure the directed relation of term A with term B the following formula can be used, whereby $|A \cap B|$ is the number of times term A and B co-occurred in the text on sentence level and $|A|$ is the number of sentences term A occured in:

$$Sig(t_A, t_B) = \frac{|A \cap B|}{|A|} \tag{2}$$

Furthermore, three aspects need to be considered when determining a term's relevance within a text for its further usage e.g. as a search word when applying the mentioned clustering approach using the extended PageRank formula:

1. Stopwords will receive a very high PageRank compared to other more meaningful terms and are therefore ruled out from the co-occurrence graph first. The reason for this is, that stopwords usually co-occur with other stopwords and many other terms on sentence level.
2. The semantic context of a term is the set of terms it significantly co-occurs with and can represent different semantic aspects which will become visible in form of separated clusters during the runtime of the algorithm when this term is deleted. A term is more general and less specific than other terms when its semantic context is larger than the contexts of other terms and will therefore receive a higher PageRank than its co-occuring terms in the same iteration of the clustering algorithm. A term's level of specifity therefore depends on the iteration during which it was deleted from the co-occurrence graph.

3. If different terms are ruled out from the same cluster in the same iteration, it means that these terms are not interconnected with each other and therefore have different semantic contexts, but on the same specifity level. They represent semantically different, yet more specific aspects of the deleted term that caused this cluster to separate from the originating cluster.

Because of these aspects to be regarded when determining a term's importance in a text, the following data need to be collected:

- the iteration (specifity level) during which this term was ruled out
- the PageRank of the term as its weight in the initial co-occurrence graph to save its global importance
- the designation of the cluster this term was removed from
- the designation of the cluster this term was a member of before it was ruled out

Using this information it is possible to determine different horizontal aspects of the terms in the text and also their vertical dependencies. These dependencies will not only become useful when it comes to automatic query formulation to find similar documents in corpora but also when semantic aspects of documents need to be explored interactively in different levels of specifity.

2.2   The Algorithm

In this section, the algorithm to determine the weight of terms from texts along with the mentioned additional information is given:

1. Remove stopwords and apply stemming algorithm on all terms in the text. (Optional)
2. Calculate the significant co-occurrences for all terms and phrases in the text and save them in an adjacency matrix in order to gain the initial co-occurrence graph.
3. Determine all separate components in the (remaining) co-occurrence graph.
4. If there are components with more than 1 term in them continue, otherwise terminate the algorithm.
5. Calculate the extended PageRanks for all terms and phrases using formula 1.
6. For all terms i in the co-occurrence graph check if the PageRank of term i is greater than the PageRank of its neighbours. If yes, mark term i for removal.
7. Remove all marked terms and phrases from the co-occurrence graph and save them in a list for removed terms along with the following data: the iteration during which these terms were ruled out, the PageRank of the removed terms in the initial co-occurrence graph, the designation of the clusters these terms were removed from, the designation of the clusters these terms were a member of before they were ruled out.
8. Go to step 3.

The list of removed terms is the result of this algorithm and contains the terms of the text ordered according to the iteration of their removal from the co-occurrence graph. If the initial graph contains stopwords they will receive a high PageRank and will be ruled out first. Therefore, the first entries in the list of removed terms will likely be these stopwords. Of course, these terms are not characteristic and discriminating and must not be regarded as keywords. If it is not possible to rule them out in the optional first step of the given algorithm, because of a maybe not available stopword list, it is still possible to detect the state transition point when the co-occurrence graph first separates into clusters which will most likely contain meaningful terms. At that point, most of the stopwords have been removed from the graph and the mean term-term-similarity of all clusters rises significantly, as it is shown in figure 1 for five example texts with around 250 til 500 words taken from the English Wikipedia.

Fig. 1 Significant increase in mean term-term-similarity
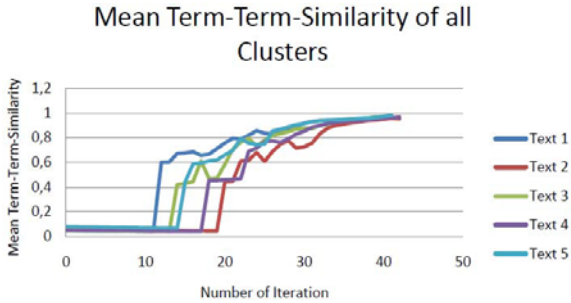
This significant increase in mean term-term-similarity correlates with the number of emerging clusters during the execution time of the algorithm, as it can be seen in figure 2. Therefore, the point in time from which on the number of clusters constantly rises, indicates that the most common terms have been ruled out.
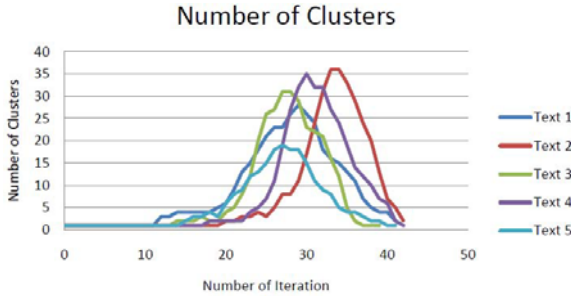
Fig. 2 Number of clusters in each iteration

These approaches, however, will not return perfect results because it is possible that meaningful terms could be disregarded as well and that stopwords can still be found in the list of removed terms when they are ruled out in a later iteration of the algorithm. Therefore, it is assumed from now on that stopwords have been successfully removed using stopword lists from the co-occurrence graph in the first step of the algorithm so that the list of removed terms will contain meaningful terms only. Also, as of now, the list of removed terms is called term list.

2.3 Detecting Keywords

Choosing the right keywords from the term list in order to obtain the most characteristic terms from an analysed text presents a challenge. However, the data collected during the runtime of the algorithm can help to make this task easier. The following criteria can be used to automatically identify such terms:

- keyword candidates should have a significantly higher PageRank than most of the other terms
- keyword candidates should originate from different clusters in a low specifity level to include terms of different topical orientations
- keyword candidates must not be stopwords

It could be that terms have been ruled out in an early iteration of the algorithm and still did not receive a high PageRank. These terms can be regarded as outliers and must not be considered as keyword candidates. Usually, they appear in a separate graph component or co-occur with only the most frequent terms in a text like stopwords, but have no relation to most of the other terms. Thus, they often carry a less important and different meaning than the rest of the text. An example is the structural information in Wikipedia articles that does not relate to the actual content.

3 Experiments

In this section, excerpts from term lists of analysed documents will be given. It is also shown that the proper selection of terms from the term lists can be used to find similar documents in the World Wide Web.

3.1 Term Lists

Next, excerpts from the term lists of three texts from the English Wikipedia are presented. The listed entries are the terms ruled out in the first iterations of the introduced algorithm. Wikipedia texts are well suited for these kind of tests, because

they often cover a topic to a great extent and their length is not too excessive. Stopwords have been removed in the first step of the algorithm. Stemming has not been applied in order to maintain the texts' specific word forms. The listed terms are ordered according to their specifity level which is the iteration of their removal from the co-occurrence graph. Also their respective PageRank as their weight in the initial co-occurrence graph, their cluster which they have been ruled out from and their previous cluster of a lower specifity level are given. The designation of a previous cluster of -1 suggests that the current cluster has no ancestors and that the term originates from the initial co-occurrence graph.

Table 1 Term list of Wikipedia's text: Mobile phone

Term	PageRank	Specifity	Cluster	Previous Cluster
wireless	2.21370199851082	1	1	-1
network	2.174379538725529	2	2	1
standard	2.044493364168943	3	4	2
manufacturers	1.978364338733551	4	7	4
mobile phone	1.9252428172465148	5	11	7
sim card	1.820872727386015	6	16	11
mobile	1.8207363358412767	7	22	16
world	1.6887491722144956	8	29	22
Nokia	1.548540647702778	8	29	22
RIM	1.4956975717224252	9	37	29

Table 2 Term list of Wikipedia's text: Food chain

Term	PageRank	Specifity	Cluster	Previous Cluster
energy	2.019349917998684	1	1	-1
hydrogen	1.149570845663863	2	2	1
food webs	1.9763575016177377	2	2	1
compounds	1.191642468961624	3	6	2
ecological	1.5593783875583225	3	6	2
food chain	1.546781525865058	3	6	2
water	0.9432458515820686	4	12	6
species	1.320258205792388	4	12	6
lake	0.28306904635885033	4	20	6
relationships	1.141761629539213	5	22	12

Table 3 Term list of Wikipedia's text: Earthquake

Term	PageRank	Specifity	Cluster	Previous Cluster
seismic	3.470016964376655	1	1	-1
activity	2.508898047198758	2	2	1
earthquake	2.1603849140374076	3	4	2
rupture	2.110088511483659	4	7	4
focal	0.3462105670913378	4	10	4
magnitude	1.978306099189843	5	12	7
energy	1.7431972228177406	5	12	7
mechanisms	0.3462105670907294	5	17	10
waves	1.595601329852196	6	20	12
areas	1.5716395135726708	7	31	20
water	1.6141782560773072	7	31	20
volcanic	1.5333069176524274	7	31	20

As one can see from the results, the weights (PageRanks) of the entries in the term lists of the analysed Wikipedia documents fulfill the expectations of a human reader. To show that the extended PageRank algorithm returns results that are comparable with the results obtained using the widely accepted difference analysis technique that does not take the co-occurrence graph structure into account but assigns term weights depending on the difference of the relative frequency of a term in the analysed document and its relative frequency in a well-balanced reference corpus, the following terms from the Wikipedia's text Earthquake have been found significant by this technique whereby the log-likelihood (LL) significance measure was used:

Table 4 Significant terms of Wikipedia's text: Earthquake

Term	Significance (LL)
earthquakes	1480.2216223383366
earthquake	959.1882822099142
seismic	300.2590716264385
fault	244.22933544492116
magnitude	206.9497951780795
rupture	132.9547802334564
Richter	125.21990077985907
waves	123.68074321950553
Earth	104.12132039805874
occur	103.99677291687112
faults	90.79552106148913
volcanic	90.01333533249635

Although these two approaches to calculate term weights differ from each other greatly, there is often an overlap in the set of the found characteristic terms as it is shown in the last example. The same observation could be made when using the TF-IDF measure to find these terms when comparing them to the results from the extended PageRank calculations. The following table shows the mean similarities between the term lists of the most characteristic terms of 200 documents from the English Wikipedia gained from the extended PageRank calculations compared to the returned term lists by the difference analysis and the TF-IDF measure. Hereby, two of these lists have been compared by counting the number of terms found in both of them divided by the number of terms in one of them. Stemming has not been applied for these tests. The number of terms in the term lists was set to be exactly 10 or 20.

Table 5 Similarity between term lists

Size of term lists	PageRank vs. TF-IDF	PageRank vs. Difference Analysis (LL)
20	0.46991515	0.3970339
10	0.6457627	0.53220326

The results show, that the term lists calculated by the extended PageRank method have a higher similarity to the ones obtained from the TF-IDF measure than to the ones from the difference analysis. When the term lists contained 10 terms only, their average overlap was at about 53% or even at almost 65% for the TF-IDF case. After applying stemming, the similarity of the term lists increased slightly in three of the four cases. However, when comparing the term lists with 10 terms obtained from the PageRank method and the TF-IDF measure, an average overlap of even 70% between them was reached. This means, that the presented PageRank method considers most of the first 10 results important that have been returned by the two other methods. Yet, it does not need to consult a reference corpus or a document corpus in order to achieve this result. In the following considerations, it will be shown that selected terms obtained from the extended PageRank algorithm can also be used as search words to automatically retrieve similar documents from the World Wide Web.

3.2 Search Word Selection for Automatic Document Retrieval

It is still an open question, which of the found characteristic terms are suitable query terms and how many of them are needed in order to obtain similar documents from corpora like the Word Wide Web? Empiric experiments have shown that at most five terms and phrases should be used for this purpose. A larger number would limit the

search results too much, while too few terms would return too many results. For an automatic retrieval of similar documents from the World Wide Web it is necessary that queries containing entries from the term lists are formulated automatically and sent to a web search engine. As it is not possible for users to change the search engines' ranking mechanisms, the only way to influence the quality of the returned documents is by choosing characteristic terms as search words. In order to receive documents that deal with the topics of the analysed document in general, proper query terms for this purpose should have been ruled out from the co-occurrence graph in the first iterations (low specifity level) and have received a high PageRank value compared to other terms in the list. Now, the documents with title and URL that have been found by Google through automatic query formulation using the characteristic terms found by the extended PageRank algorithm from the previous section are listed:

Table 6 Results for Wikipedia's text: Mobile phone

Query: wireless network standard manufacturers mobile phone
• Mobile phone - Wikipedia, the free encyclopedia *en.wikipedia.org/wiki/Mobile_phone* • Wireless Networking Links: Wireless LANs, Broadband Wireless Access *site.nissr.com/zeino/wireless-links.html* • Umts Standard-Umts Standard Manufacturers, Suppliers and Exporters *www.alibaba.com/showroom/umts-standard.html* • Mobile Phone Network Standards at Gizmo Cafe *http://www.gizmocafe.com/handhelds/parents-cell-guide/gsm-vs-cdma.aspx* • WIRELESS NETWORK STANDARDS by Timothy J. O'Neil *faculty.ed.umuc.edu/s̄dean/ProfPaps/Bowie/T4-0607/ONeil.pdf*

Table 7 Results for Wikipedia's text: Food chain

Query: energy hydrogen food webs compounds ecological
• Food chain - Wikipedia, the free encyclopedia *en.wikipedia.org/wiki/Food_chain* • Autotroph - Wikipedia, the free encyclopedia *en.wikipedia.org/wiki/Autotroph* • Biogeochemical cycle - Wikipedia, the free encyclopedia *en.wikipedia.org/wiki/Biogeochemical_cycle* • about food chain — Facebook *facebook.com/topic.php?uid=132237480175201&topic=29* • Chapter 3 – Matter, Energy, and Life *zoology.muohio.edu/oris/cunn06/cs6_03.htm*

Table 8 Results for Wikipedia's text: Earthquake

Query: seismic activity earthquake rupture magnitude
• Earthquake - Wikipedia, the free encyclopedia *en.wikipedia.org/wiki/Earthquake* • Friday focal mechanisms: Chile's persistent seismic gap *all-geo.org/highlyallochthonous/2011/02/friday-focal-mechanisms-chiles-persistent-seismic-gap/* • New Empirical Relationships among Magnitude, Rupture Length, Rupture Width, Rupture Area, and Surface Displacement *classes.engr.oregonstate.edu/cce/winter2011/ce570-001/References/02b%20PATH%20-%20Seismic%20Hazard%20Analysis/1994%20Wells%20and%20Coppersmith.pdf* • Earthquake seismology *web.ics.purdue.edu/ēcalais/teaching/eas450/seismology4.pdf* • Geotechnical Earthquake Engineering for Seismic Design *web.ics.purdue.edu/ēcalais/haiti/macrozonation/formation/1_Seismic%20Design%20Framework.pdf*

As shown in the results, the found documents by Google are highly related to the analysed documents. Moreover, the analysed documents themselves appeared as the first result in each of the tests. This finding is interesting because the presented approach obviously cannot only be used to find topically similar documents but could also be used as a basic tool for plagiarism detection. As the PageRank clustering algorithm returns not only the term weights, but also the sequence of clusters that removed terms were part of before they were ruled out from the co-occurrence graph and also their related terms in these clusters, lexical chains with such related terms can be constructed that can help to identify characteristic sections or sentences in texts e.g. for automatic text summarization and can be automatically used as specific queries to find documents in corpora with similar term sequences, too. This possibility will be examined in a later publication.

3.3 Interactive Document Retrieval

If users should be able to modify the automatically generated queries, then the term lists contain valuable information to facilitate interactive document retrieval. Besides to just recommend terms with a high PageRank for this purpose, the specifity level offers users the possibility to hierarchically browse through the terms which should be selectable as search words in an interactive search system. This option is also useful when topically different connected aspects of a document need to be explored in a horizontal or vertical manner. Especially terms that have been ruled out in the same iteration from the same cluster are interesting candidates for

horizontal exploration as they usually represent topically different aspects of a term in a higher specifity level than its own. Another possibility is its usage in automatic query expansion, similar to the approach presented in [11]. Given a query term, the search system could suggest other related terms of a higher or lower specifity level in order to broaden or narrow the search space. The relatedness of terms can be obtained from the co-occurrence significances of the used documents' co-occurrence graphs. This new query expansion approach is promising as it takes not only into account the degree of relatedness when recommending terms but also hierarchical dependencies. If a user accepts the suggested expansion terms it means that this user is satisfied with the recommended terms. It is therefore sensible to think about ways to influence the strength of the involved term relations through human interaction. This way, the PageRank of terms would depend on the structure of a document's co-occurrence graph, on the significance values of term pairs as shown in this paper and also on implicitly or explicitly gained user feedback. A personalized ranking of terms in documents would consider a user's special interest in topics and would therefore influence the selection of search words in automatic document retrieval. These possibilities will also be elaborated on in a later publication.

4   Conclusion

In this paper, a new algorithm to determine the weights of terms in text documents using extended PageRank calculations has been presented. It has also been explained which of these terms should be used as search words for the automatic retrieval of similar documents in the World Wide Web. To demonstrate the effectiveness of this approach, term lists of three analysed documents from the English Wikipedia have been given and automatically selected terms have been successfully used to find similar documents using Google. As the algorithm assigns terms a specifity level and their semantic relations are considered, this method can become helpful in building lexical chains with meaningful terms to find documents with similar term sequences, in the interactive exploration of rather general or more specific aspects of a document and in interactive document retrieval. Interactive search systems can employ this method to improve functions like automatic query expansion, as query terms could be expanded with related terms of a different specifity level, depending on user preferences. Another field of application for this method can be seen in automatic taxonomy extraction. However, it is likely that a single document is not sufficient to build an acceptable term hierarchy. Therefore, the analysis of combined co-occurrence graphs of different topically related documents could gain such hierarchies of a higher quality which contain less random and document-specific but more statistically validated dependencies that appear in a number of documents of the used text corpus. The applicability of the presented method for automatic taxonomy extraction is currently being examined.

References

1. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communications of the ACM 18(11), 613–620 (1975)
2. Heyer, G., Quasthoff, U., Wittig, T.: Text Mining - Wissensrohstoff Text. W3L Verlag, Bochum (2006)
3. Kubek, M., Unger, H.: Empiric Considerations of the PageRank's Clustering Property. In: 7th International Conference on Computing and Information Technology (IC2IT), Bangkok (2011)
4. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. In: Technical report, Stanford Digital Library Technologies Project (1998)
5. Wang, J., Liu, J., Wang, C.: Keyword extraction based on pageRank. In: Zhou, Z.-H., Li, H., Yang, Q. (eds.) PAKDD 2007. LNCS (LNAI), vol. 4426, pp. 857–864. Springer, Heidelberg (2007)
6. Mihalcea, R., Tarau, P., Figa, E.: PageRank on Semantic Networks, with application to Word Sense Disambiguation. In: Proceedings of the 20th International Conference on Computational Linguistics (2004)
7. Sodsee, S., Komkhao, M., Meesad, P., Unger, H.: An Extended PageRank Calculation Including Network Parameters. In: Computer Science Education: Innovation and Technology (CSEIT 2010) Special Track: Knowledge Discovery, KD 2010 (2010)
8. Buechler, M.: Flexibles Berechnen von Kookkurrenzen auf strukturierten und unstrukturierten Daten. Masters thesis, University of Leipzig (2006)
9. Quasthoff, U., Wolff, C.: The Poisson Collocation Measure and its Applications. In: Proc. Second International Workshop on Computational Approaches to Collocations, Wien (2002)
10. Dunning, T.: Accurate methods for the statistics of surprise and coincidence. Computational Linguistics 19(1), 61–74 (1994)
11. Kubek, M., Witschel, H.F.: Searching the Web by Using the Knowledge in Local Text Documents. In: Proceedings of Mallorca Workshop 2010 Autonomous Systems. Shaker Verlag, Aachen (2010)

Semantic Similarity-Based Web Services Access Control

Yi Zhao and Xia Wang

Abstract. As the Semantic Web has been applied in the Web Services to integrate data across different applications with the increasing development of the Semantic Web technologies, it is essential to maintain the security of the organizations involved in the Semantic Web Services. Security is a crucial concern for commercial and mission critical applications in Web-based environments. To guarantee the security of the web services, security measures must be considered to protect against unauthorized disclosure, transfer, modification, or destruction, whether accidental or intentional. Access control is a kind of security measurements to guarantee the service processes, which is defined to allow resource owners to define, manage, and enforce the access conditions for each resource. In this paper, an ontological concept similarity algorithm is first proposed taking multiple concept relations into consideration. Then, an attribute based access control model based on the semantic similarity (SABAC, for short) is proposed to specify access control over attributes defined in domain ontologies. An experimental prototype and detailed empirical discussions are presented, and the method is validated in the framework of web service selection.

1 Introduction

With the rapid development of the Semantic Web technologies, the Semantic Web [1] has been introduced to apply in the Web Services to help integrate data across different applications. With the increasing interest in Semantic Web Services 2, the relative security control methods, like access control method, have to be developed to secure the whole Web Service process.

Security is a crucial concern for commercial and mission critical applications in Web-based environments. To guarantee the security of the Web Services, security measures must be considered to protect against unauthorized disclosure, transfer, modification, or destruction, whether accidental or intentional. Access control is a

Yi Zhao · Xia Wang
Fernuniversität Hagen, Faculty of Mathematics and Computer Sciences
e-mail: `yi.zhao@fernuni-hagen.de`
Jacobs University, Large-Scale Scientific Information Services Research Group
e-mail: `xia.wang@jacobs-university.de`

kind of security measurements to guarantee the service processes, which is defined as the mechanism that allows resource owners to define, manage, and enforce the access conditions for each resource 6. Up to now, there are a lot of access control models have been proposed such as the mandatory access control (MAC), the discretionary access control (DAC), role-based access control (RBAC) 7, attribute-based access control (ABAC) 3, and context-based access control (CBAC) 8. A major drawback of the approaches mentioned above does not exploit the rich semantic interrelationships in the data model. The relative complement is the semantic-aware access control model which contains semantic-based access control (SBAC) 4, and semantic context-aware access control (SCAC) 5. These two models support making more precise decisions regarding authorization and inference rules. They fetch users' context and ontology from middleware, with which context hierarchies are built. However, the semantic relationships between the contexts, authorizations and inference rules are not considered.

In the context of semantic web services (SWS), it is a key functionality to enable service discovery, selection, and composition under the assumption that services are described using heterogeneous ontologies. In particular, concept similarity is important when the selection of services is based on the similarity of descriptions of semantic services.

In this paper, an attribute based access control model with semantic interoperation (SABAC, for short) is proposed to specify access control over concepts defined in ontologies. The model is built on the basis of XACML [14] policy language with the application of semantic interoperation. The semantic interoperation is realized between the attributes of the service requester and the service provider. The mapping result can be kept in a mapping base for reuse, and similarly, the generated access control policies can be saved for future reuse. All of these can make the access to the semantic web services secure.

The rest of the paper is organized as follows. The semantic similarity is introduced in the following section. Followed by the motivations of the section of the motivations of this paper, the measurement of the semantic web services similarity is then given. After that, the architecture of the proposed semantic interoperation based access control model SABAC is introduced. Finally are the conclusions.

2   Semantic Similarity

Similarity of semantic services is measured by the conceptual overlap of service capabilities, including non-functional information such as service name, service categories, and service qualities, and functional properties, i.e., service operations. In order to illustrate this, Fig. 1 shows an example of five services related to postal codes. The services are used to look up zip codes or to calculate distances between two places given their zip codes.

Taking *sws4* and *sws5* of Fig. 1 as examples, we assume that *zip* and *code* are similar to a certain extent in a specific application. By comparing service names and service operation names, services can be regarded as similar from the

signature level. In addition, by matching their operations and inputs/outputs we can conclude that the services *sws4* and *sws5* are similar - both services can provide information on a city according to a given zip code. Furthermore, if we assume that a machine can understand that there is some similarity between {*zip*, *ZipCode*, *code*, *Zip_Code_*1, *code*1} or {*CalcDisTwoZipsKm*, *findZipCodeDistance*}, then we can derive that *sws1:operation* is similar to *sws5:operation1*, *sws2:operation* is similar to *sws5:operation2* and *sws4:operation2*, and *sws3:operation* is similar to *sws4: operation1*. As it becomes obvious from this example, ontological concept similarity based on both syntactic and semantic approaches is the basis for service similarity.

```
sws1: ServiceName: Zip_Code_Lookup              sws2: ServiceName: CityStateByZip
        operation: GetZipByCityState                    operation: GetCityStateByZip
            input: StateCode, CityName                      input: ZipCode
            output: Body                                    output: Body

sws3:ServiceName: Distance_between_two_zip_codes  sws4: ServiceBane: ZipCodeService
        operation: CalcDistTwoZipsKm                    operation1: findZipCodeDistance
            input: Zip_Code_1,Zip_Code_2                    input:code1,code2
            output: Body                                    output: Body
                                                        operation2: findZipDetails
sws5: ServiceName: Zip4                                     input: code1
        operation1: FindZipPlus4                             output: Body
            input: Address, City, State
            output: Body
        operation2: FindCityState
            input: Zip
            output: Body
```

Fig. 1 Examples for descriptions of semantic web services

In the above service descriptions, there are some special concept features appearing in semantic services. They cause the following problems. Concepts are of two types: some are formal ones and the others are composite ones combined of several concepts, which are not formal vocabularies indexed by dictionaries such as WordNet. For such composite concepts, e.g., *FindZipCodeDistance* and *CalcDistTwoZipsKm*, the traditional methods (e.g., *Edit Distance* of strings) are not applicable to measure their semantic similarity. Therefore, the similarity of composite concepts constitutes a considerable problem.

1. A concept has a specific sense in a given service application description, e.g., *code* in our above service examples does not mean *cipher* nor *computer code*, but is actually used as synonym for *zip code*. Therefore, word disambiguation is required before considering concept similarity.

2. Information among concepts should be more than what is represented by small pieces of service ontologies, which are currently used by semantic service descriptions. And there should be more concepts relations than just the *isA* relation. Therefore, mining implied information and concept relations is necessary and contributes to the assessment of service similarity.

3   Semantic Similarity

Most of current access control schemes base their authorization approaches on lo-cally-issued credentials that are based on user identities, but the drawback is that they are not interoperable, which is the reason that Semantic Web based ap-proaches to access control policies have been developed in recent years 9101112.

Pan et al. presented a novel middle-ware based system 10 to use semantics in access control. It is based on RBAC model 9 with a mediator to translate the access request between organizations by replacing roles and objects with matched roles and matched objects. Semantic mapping is used for roles interoperation in order to find the similarity or separation of duties between roles in two ontologies.

Current tendency of the research of ABAC is around the application of semantic or ontologies. In 1112, the authors have tried to use ontologies in the attribute-based access control. In 11 an ontology-based inference engine is pro-posed to obtain the policies. The authors claim that it enables policy administra-tors to concentrate on the properties they deem necessary from their point of view; and they do not need to determine in advance which attributes a subject may use to prove these properties. While with the popularity of the Semantic Web tech-niques, the application of domain ontologies is becoming more and more com-mon. The approach in 12 is based on determining the set of required user attributes to access a specific object in order to facilitate coalition based access control. But this approach relies on identifying the necessary attributes required by external users to gain access to a specific organizational object (or service).

Semantic similarity can disclose the latent relationships between subjects and objects though they are syntactically different, and the current access control me-thods are either dynamically not interoperable, or the semantic based ABAC me-thods have also shortcomings. As RBAC is not flexible to be used in large open systems, we are highly motivated by the aforementioned problems to develop our semantic similarity based ABAC methods, aiming to realize the agent interopera-tion dynamically and deal with the semantic attribute mapping with the support of domain ontologies both on service request side and service provider side. In this means, the access control is transferred to the comparison between domain ontologies.

4   Measuring Web Services Similarity

In this section, the method of the web services similarity will be proposed based on the ontological concept similarity algorithm.

4.1   *Ontological Concept Similarity Algorithm (OCSA)*

In the context of semantic web services, concept similarity is calculated both from the point of view of ontology structure, in which *conceptual distance* is the

shortest path between two linked concepts in the semantic net of hierarchical relations, as *Dis_s*, and from the point of view of ontology information content, i.e., the more properties two concepts have in common, the more closely related they are, as *Dis_c*. Agreeing with Rada's statement [16]: *Distance is a metric on the sets of concepts (single and compound) defined on a semantic net"*, we propose a novel algorithm as:

$$Dis(c_1, c_2) = \omega_s * Dis_s(c_1, c_2) + \omega_c * Dis_c(c_1, c_2) \tag{1}$$

where c_1 and c_1 are two concepts of semantic nets, and w_s and w_c are, respectively, the weights of structure similarity and content similarity, and $w_s + w_c = 1$.

From the structural point of view, concept similarity does not depend on the path length (l) together with concept depth (h) and density (d) of ontologies as proposed in [17]. In our semantic net, it relates to path length and ontology concept depth, which is defined as $sim(c_1; c_2) = f(l; d)$. As a metric, the function $f(x; y)$ must have properties such as (1) $f(x; x) = 0$ (reflexivity), (2) $f(x; y) = f(y; x)$ (symmetry), and (3) $f(x; y) , 0$ (non-negativity). This function does not, however, satisfy the triangular inequality as discussed in [16], because there is more than one single relation involved in our case.

When various concept relations exist in the semantic net, then different tags should be assigned to different links to indicate the importance (or strength) of the links between connected concept nodes. Then, we assume that semantic relatedness or semantic distance of concepts is measured by weight values on their paths, which is unrelated to the number of directed edges connecting concept nodes, i.e., the edge length has no semantic meaning in our context.

According to this discussion, we define *Dis_s* as

$$Dis_s(c_1, c_2) = f(f_1(l), \ f_2(d)) \tag{2}$$

where f_1 and f_2 are two functions related to path and ontology structure, respectively. The characteristics of this semantic net are that, when the path of two concepts tends to infinity, the similarity tends to 0; otherwise the similarity tends to 1. Hence, we take an exponential function to define

$$f_1(l) = e^{-\omega p} = e^{-\sum \tilde{C}_k l_k} \tag{3}$$

If only one relation is considered, then $f_{1(l)} = e^{-\tilde{C}_l l}$, which is consistent with [17].

Furthermore, we propose a simplified method to compute the shortest path. If the similarity between two concepts on a path has been evaluated to be small, we do not consider such path as valid for our approach. Therefore, we define that a concept similarity value should not be less than a threshold τ. We set $\tau = 0.2$ and then consider Eq. 3, that will be $e^{-0.8*l} = 0.2$, then $l \doteq 2.012$ when $\tilde{C}_s = 0.8$ (the similarity between a concept and its superclass is assumed to be 0.8. Therefore in our approach we assign $l = 3$ which corresponds to human intuition, i.e., if two concepts are similar, their distance is not too high.

If there is no path between nodes c_i and c_j in the semantic net, instead there is a common superconcept that subsumes the two concepts being compared, then the conceptual similarity can be measured based on the maximum depth (h) to their common parent node. This is also possible in our approach, however, we consider 4 different relations. Thus, we use the function of [17] as $f_2(h) = \dfrac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}}$.

In summary, based on Eq. 2,

$$Dis_s(c_1, c_2) := \begin{cases} e^{-\sum \tilde{c}_k l_k \cdot \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}}}, & C_1 \neq C_2 \\ 1, & otherwise \end{cases} \tag{4}$$

where $\beta \geq 0$ is a parameter used to scale the contribution of depth h in a concept hierarchy. Consistent with previous work, Eq. 4 considers the shortest path and concept depth of their common superconcept.

From the point view of information content held by concepts, if concepts have rich information as concept attributes, instances and given synonyms, then the concept distance is measured as

$$Dis_c(c_1, c_2) = \frac{|c_1 \cap c_2|}{|c_1 \cap c_2| + \gamma |c_1 / c_2| + (1 - \gamma)|c_2 / c_1|} \tag{5}$$

where $|c_1 \cap c_2|$ is the intersection of the two concepts indicating the concepts' common characteristics and $|c_1 / c_2|$ is their difference; "| |" is the cardinality of a concept information set, and γ ($0 \leq \gamma \leq 1$) is a weight that defines the relative importance of concepts' non-common characteristics.

4.2 Web Services Similarity

We calculate the web services similarity based on the ontological concept similarities as,

1. If concepts c_i, c_j are single concept terms, then

$$Dis(c_i, c_j) = \text{the shortest distance between } c_i, c_j \tag{6}$$

2. If concepts c_i, c_j are compound or conjunctive concepts, then

$$dis(x_1 \wedge \ldots \wedge x_k, y_1 \wedge \ldots \wedge y_m) = \frac{1}{km} \sum_{i=1}^{k} \sum_{j=1}^{m} dis(x_i, y_j) \tag{7}$$

where $\{x_1, x_2, \cdots, x_k\}$ are sub-concepts of c_i and $\{y_1, y_2, \cdots, y_m\}$ are sub-concepts of c_j.

    3. If only one concept c_i or c_j is conjunctive, then

$$dis(x_1 \wedge \ldots \wedge x_k, c_j) = \frac{1}{k}\sum_{i=1}^{k} dis(x_m, c_j) \qquad (8)$$

Based on these, we propose an ontology-based similarity method of semantic web services as,

$$sim_{Service} = \sum sim_{Concept} + \sum sim_{Operation} \qquad (9)$$

where $sim_{Concept}$ is the sum similarities over all service concepts, and $sim_{Operation}$ is the sum similarities over all of the operation parameters with their data types.

5 The SABAC Model

Our SABAC architecture (see Fig. 2) of the semantic interoperation based approach is modeled based on XACML [14]. XACML stands for eXtensible Access Control Markup Language. It is recognized as an effective ABAC policy description method that can exactly describe the semantics of a policy [13]. XACML is a declarative access control policy language implemented in XML and a processing model, describing how to interpret the policies.

    The main components of the XACML architecture are PDP (Policy Decision Point), PEP (Policy Enforcement Point), PAP (Policy Administration Point), and PIP (Policy Information Point), where the PDP receives an XACML request, fetches the applicable policy(s) from the policy administration point, retrieves the attribute values from the policy enforcement point, evaluates the request against the applicable access control policies and returns an authorization decision to the PEP; The PEP receives an access request, extracts the attributes in the request, generates an XACML request and sends it to the PDP for evaluation; The PAP creates an XACML access control policy(s) and stores it in a policy database server. It addition to this, it sets a restriction in order to prevent unauthorized access to the access control policies. Beside this, it conducts a regular check in order to maintain the uniqueness of policy identifiers; The PIP is a component that acts as a directory server that stores the attribute values and makes it available to the PDP. Attribute values are the data that describe the characteristics of a subject, resource, action and environment.

    Similar to XACML, SABAC includes the components of Application (requester), Provider, PAP, PEP, PDP, and PIP. Some other components are added to the original XACML model, which are application domain ontology, provider domain ontology, semantic interoperation, Mapping Base, and Rule Generation. With the popularity of the use of the Semantic Web techniques, the service requester and service provider have to be strengthened with the domain ontologies. The semantic interoperation component can therefore realize the mapping work with the support of the two domain ontologies.
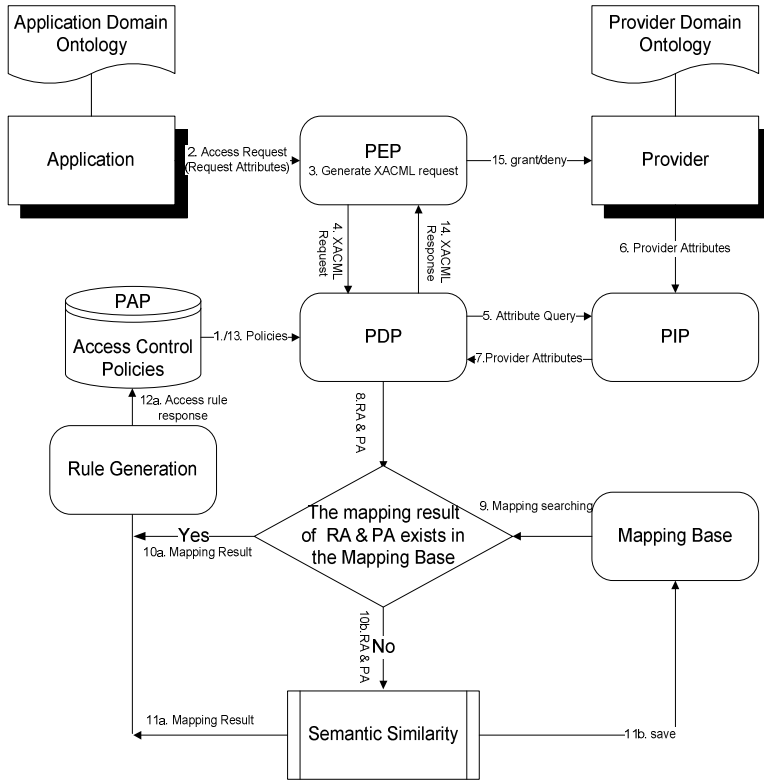
Fig. 2 Architecture of SABAC

With the help of these additional components (compared to the traditional XACML model), the proposed SABAC model is more semantic and efficient. Its performance analysis will be given in the following subsection.

5.1 The Performance Principle of the Architecture

Our model is built under the condition that both the application side and provider side are deployed with domain ontologies, which are designed by domain experts. The access control decision of our SABAC model is performed as follows. A PAP creates an XACML policy and provides it to the PDP. The application sends an attribute request to the PEP, the PEP forms the request as an XACML request, and forwards the request attributes (RA) to the PDP. The PDP requests those attributes from a PIP. The PIP collects the provider attributes (PA) from provider side, and delivers the attributes back to the PDP. The PDP then sends the RA and PA first to the Rule Base to check whether the mapping result has already exists. If the rule

does exist, it is sent back to the PDP; otherwise, the both attributes are sent to the mapping component, where the semantic similarity between RA and PA is calculated. The generated mapping result is, on one side, sent back to the PAP to form a policy, on another side, saved in the Mapping Base for the future use. The PDP evaluates the policy and sends the response back to the native format of the PEP and forwards it. The PEP makes the decision: grant the access or deny it. If access is granted, the PEP allows access to the resource. Otherwise, access is refused.

Ontological Concept Similarity Algorithm explained in last Section is used to calculate the services similarity in semantic mapping part.

5.2 Rule Generation

If the similarity distance between two attributes from request side and provider side is over a pre-defined threshold, they are considered similar. Therefore, the requester can get the access permission from the service provider. In other words, the requester can have access to the provider resources. Furthermore, the access control rule can be automatically generated according to the mapping result of the related attributes from both sides.

During the interoperation process, the source attributes (request attributes and provider attributes) are first browsed in the Mapping Base to check if there already exist available mapping results. Once the interoperation result does exist, it can be immediately obtained from the Mapping Base. Otherwise, the similarity measure must be calculated to put in the Mapping Base. If the similarity value between two source attributes is above a pre-defined threshold, the two attributes are considered similar.

The access control rules are then generated in the Rule Generation component according to the interoperation result, with the following format:

Attributes RA and PA are similar ==> the service request is granted;
Or,
Attributes RA and PA are not similar ==> the service request is denied.

Here, "similar" means syntactically, semantically, or structurally resemble with the up mentioned methods. If the two source attributes are considered similar, the requester is granted to access the service provider and get the response; and if two source attributes are not considered similar, the service requester is then denied to get access to the service provider.

The access control policy is formed according to the generated rules, which consists of the service request attributes, action group (such as grant and deny), and the service provider attributes. If the number of request attributes and provider attributes gets rising, the management of the access control policy is worth discussing. The mapping result is then saved in the Mapping Base for the future application, while at the same time, the access control rule which is formed with respect to the mapping result is then used to get the access control policy to get saved in the PAP for future reuse.

Table 1 Relationships between attributes

No	Characteristics	Rules: Given two attributes *sa* and *sb*
R1	Properties	If properties (data type property/object property $\notin$ null) of *sa* and *sb* are similar, *sa* and *sb* are also similar.
R2	Child classes	If all child classes of *sa* and *sb* are similar, *sa* and *sb* are also similar.
R3	Parent- & child-classes	If parent class & one of their child classes of *sa* and *sb* are similar, *sa* and *sb* are also similar.

5.3 Conclusions

In this paper, an attribute based access control model with semantic interoperation is proposed to specify access control over request attributes and provider attributes defined in their domain ontologies. The architecture of the model SABAC is built on XACML policy language, where the modification lies in: 1) both the Requester and Provider are strengthened with their domain ontologies; 2) the semantic interoperation process between attributes is based on the ontological concept similarity method; 3) mapping Base is designed to save the semantic interoperation result. When the mapping result between two source attributes are already existent in the Mapping Base, the efficiency of the mapping interoperation is greatly improved; 4) the Policies obtained from the access control rules are saved in the PAP for future use, similar to the Mapping Base.

Semantic interoperation for attribute based access control between the service requester and service provider can make the access to the semantic web services secure. The performance analysis of the SABAC model shows that the proposed model is very promising. Our further work will be around a sample implantation and the semantic representation of the access control policies.

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American, pp. 28–37 (2001)
2. Payne, T.R., Lassila, O.: Guest Editors' Introduction: Semantic Web Services. IEEE Trans. Intelligent Systems 19(4), 14–15 (2004)
3. Priebe, T., Fernandez, E.B., Mehlau, J.I., Pernul, G.: A Pattern System for Access Control. In: Proceedings of the 18th Annual IFIP WG 11.3 Working Conference on Data and Application Security, Sitges, Spain (2004)
4. Javanmardi, S., et al.: SBAC: A Semantic-based AccessControl Model. In: SBAC: A Semantic-based AccessControl Model, The 11th Nordic Workshop on Secure IT-systems, Linköping, Sweden, pp. 19–20 (2006)
5. Ko, H.J., Kang, W.: Enhanced Access Control with Semantic Context Hierarchy Tree for Ubiquitous Computing. IJCSNS International Journal of Computer Science and Network Security 8(10), 114–120 (2008)

6. Samarati, P., di Vimercati, S.d.C.: Access Control: Policies, Models, and Mechanisms. In: Focardi, R., Gorrieri, R. (eds.) FOSAD 2000. LNCS, vol. 2171, pp. 137–196. Springer, Heidelberg (2001)
7. Sandhu, R., Coyne, E., Feinstein, H., Youman, C.: Role-based Access Control Models. IEEE Trans. Computers 29(2), 38–47 (1996)
8. Corradi, A., Montanari, R., Tibaldi, D.: Context-based Access Control for Ubiquitous Service Provisioning. In: Proceedings of the 28th International Computer Software and Applications Conference, COMPSAC 2004 (2004)
9. Ferraiolo, D.F., Sandhu, R., Gavrila, S.I., Kuhn, D.R., Chandramouli, R.: Proposed NIST Standard for Role based Access Control. Information and System Security 4(3), 224–274 (2001)
10. Pan, C.C., Mitra, P., Liu, P.: Semantic Access Control for Information Interoperation. In: Proceedings of the 11th ACM Symposium on Access Control Models and Technologies (SACMAT 2006), pp. 237–246. ACM, New York (2006)
11. Priebe, T., Dobmeier, W., Kamprath, N.: Supporting attribute-based Access Control with Ontologies. In: First International Conference on Availability, Reliability and Security (ARES 2006), pp. 465–472. IEEE Computer Society Press (2006); ISBN 0-7695-2567-9
12. Warner, J., Atluri, V., Mukkamala, R., Vaidya, J.: Using Semantics for Automatic Enforcement of Access Control Policies among Dynamic Coalitions. In: Proceedings of ACM Symposium on Access Control Models and Technologies (SACMAT), Sophia Antipolis, France (2007)
13. Lang, B., Zhao, N., Ge, K., Chen, K.: An XACML Policy Generating Method based on Policy View. In: 3rd International Conference on Pervasive Computing and Applications, vol. 1, pp. 978–971 (2008); ISBN: 978-1-4244-2020-9
14. Moses, T.: eXtensible Access Control Markup Language(XACML) Version 2.0. OASIS Standard (2005)
15. Ehrig, P., Haase, M.: Hefke, and N. Stojanovic, Similarity for Ontologies – A Comprehensive Framework. In: ECIS (2005)
16. Wang, X., Ding, Y.H., Zhao, Y.: Similarity Measurement about Ontology-based Semantic Web Services. In: Shafiq, O. (ed.) Proc. Workshop on Semantics for Web Services, Zuerich, pp. 25–30 (2006)
17. Bouquet, P., Serani, L., Zanobini, S.: Semantic Coordination: a new approach and an application. In: 2nd International Semantic Web Conference (ISWC 2003), Florida, USA (2003)

Evolution of Verification Techniques by Increasing Autonomy of Cooperating Agents

Francesca Saglietti, Sven Söhnlein, and Raimar Lill

Abstract. As system parts are becoming increasingly decoupled, gaining at the same time in terms of local autonomy, this article elaborates on the effects this trend has on verification and validation techniques. Both qualitative approaches to fault detection and quantitative approaches to reliability assessment are analyzed in the light of their evolution to adapt to the increasing decentralization and autonomy of modern 'systems of systems'.

1 Introduction

The responsibility role assumed by software-based applications when controlling a variety of critical tasks is continuously increasing. While in previous times the limited functionality of such systems allowed for monolithic designs, their increasing complexity has led - over the past decades – to growing levels of modularity, ranging from simple well-structured programs to sophisticated component-based systems including pre-developed packages.

Evidently, the structure introduced by modularity strongly supports transparency and understandability by endorsing the principles of abstraction and of separation of concerns, in particular contributing to lean maintenance by favoring change management via highly cohesive modules (easing fault localization) and loosely coupled modules (reducing potential side effects arising during fault removal). On the other hand, the insertion of pre-developed components was shown to introduce new fault sources in case the original application context in which the components originate slightly differs from the future context addressed by the new system to be developed [5].

This trend towards decoupled software parts is further increasing, typically for systems whose functional scope is dynamically evolving with time, like

- critical services provided on internet platforms (*cloud computing*);
- controllers protecting different assets considered as essential for the functioning of a society (*critical infrastructures*), like energy generation, transmission

Francesca Saglietti · Sven Söhnlein · Raimar Lill
Chair of Software Engineering
University of Erlangen-Nuremberg, Erlangen (Germany)
e-mail: {saglietti,soehnlein}@informatik.uni-erlangen.de

and distribution, telecommunication, water supply, food production and distri-
bution, public health, transportation, financial and security services;
- mechanical, intelligent agents performing individual or common tasks (*coop-
   erative robots*).

While the motivation leading to such decentralized, highly decoupled and auto-
nomous sub-systems – namely, ease of system evolution over time as well as
growth of time efficiency and service flexibility – clearly strengthens their attrac-
tiveness, the high dependability demands posed on such software applications
render their verification even harder than is already the case for more modest
functional scopes.

In fact, as the individual system parts evolve with time and behave at a high
degree of autonomy, the multiplicity of their interplay increases at rapid pace and
is extremely difficult to be systematically captured by testing. On the other hand,
in order to verify acceptable behavior, an accurate preliminary analysis should
identify the variety of potential scenarios involving the interaction of autonomous
parts and to assess their adequacy by representative test cases.

The intention of the present contribution is to elaborate on this novel challenge
posed to software reliability engineering by illustrating recent and ongoing work
addressing the evolution of verification activities with increasing autonomy of sys-
tem parts.

2   Autonomy

The ancient Greek root of the term '*autonomy*' (*auto* = self + *nomos* = law) re-
veals that its original meaning generally referred to entities providing themselves
with their own laws. Depending on the underlying political or ethical context, this
definition may allow for slightly diverse interpretations.

- within a *political context*, it refers to the self-government of human popula-
   tions, while
- in terms of *moral philosophy*, it refers to the moral responsibility of an individ-
   ual for his / her actions.

In both cases, it involves the capacity of an individual entity (human, population,
or technical system) to make a rational and informed decision. In spite of the idea-
listic content hidden behind this assumption, it is well-known to software engi-
neers – as it is to politicians and philosophers – that full autonomy (i.e. completely
decoupled individual decisions) cannot help achieving an overall prioritized target
of a society or an application.

The opposite, namely central controllers fully dictating actions to agents, evi-
dently severely limit the potential capabilities provided at local level, restricting
both performance and flexibility.

Between both extremes, the appropriate degree of autonomy is determined by optimizing the trade-off between individual freedom and essential rules of co-existence and cooperation; in case of software-based systems the latter are mainly characterized by

- *synchronization* activities required to achieve a common target, as well as
- *coordination* activities required to avoid resp. resolve conflicts.

Therefore, the degree of autonomy may be considered as higher

- the less *synchronization* constraints are required for cooperative tasks, and
- the more *coordination* tasks (including conflict resolution) are carried out in a decentralized way.

Formally, a community Com of cooperating units may be represented (as inspired by [6]) as

$$\textbf{Com = (Aut, Init, Goal, Control)}$$

where
- **Aut** denotes a set of units;
- **Init** denotes an initial global environment;
- **Goal** denotes an overall goal or a set of terminal global environments;
- **Control** denotes a set of rules concerning future behavior of units assuming knowledge of its present global environment.

Each unit aut $\in$ Aut is described as

$$\textbf{aut = (init, goal, control)}$$

where
- **init** denotes an initial local environment;
- **goal** denotes an individual sub-goal (the task of the individual), or a set of local environments;
- **control** denotes a set of individual rules concerning its future behavior assuming knowledge of relevant portions of its local environment.

The degree of autonomy characterizing each individual aut within a community Aut is then reflected by the relation between the scope of central functionality taken over by the *central intelligence* Control(Aut) and the scope of functionalities controlled by the *local intelligence* control(aut) of the individual considered:

$$\textbf{Autonomy (aut) = funct[control(aut)] / funct[Control(Aut)]}$$

In quantitative terms, this relation may be evaluated by

- the ratio of *logical complexities*, or by
- the ratio of *frequencies of occurrence*.

3   Verification

From the perspective of verification, the major challenge posed by autonomous systems relates to their inherent low compositionality. As the behavior of an individual may depend and may have an impact on the behavior of its operative environment, understanding each single part of a system (like in case of a conventional controller and of its controlled units) is not sufficient any longer. The central principle of '*separation of concerns'* which supports classical verification approaches, is hardly applicable here. On the contrary, at system level new global properties (so-called '*emergent behavior'*) may arise, not easily deducible by logical inference, nor by composition of local properties. Emergent behavior may involve

- *intended*, expected or desired properties, implicitly targeted by the overall goal to be achieved by cooperation, but also
- *unintended,* surprising or undesired behavior (e.g. cascading) due to unpredicted side effects (e.g. deadlocks, or even unsafe behavior).

3.1   *Qualitative Verification by Structural Analysis*

A major approach to verification is based on (static or dynamic) analysis of system structure (so-called white-box or grey-box verification). In case of *monolithic units* of moderate size this is achieved by modeling the control flow (possibly with data flow annotations) and to analyze the model for the purpose

- of identifying *anomalies* (e.g. undefined or unused variables, dead code), and
- of selecting test data capable of achieving pre-defined *code coverage* criteria (unit testing [8]).

Component-based systems are usually implemented by object-oriented components exchanging messages via method calls. Message passing usually takes place by synchronous communication (hand-shake); this reduces the possibilities of race conditions and of further side effects. An approach to capture component interaction by abstraction was developed in [9] by means of communicating state machines invoking each other by parameterized message passing, hereby triggering potential state transitions both in invoking and invoked components.

In such cases structural analysis will require (in addition to the already mentioned unit verification approaches at component level) particularly accurate analysis of the intra-modular structure for the purpose

- of identifying *interface inconsistencies* (e.g. units referring to deviating physical systems of reference [5]), and
- of verifying resp. falsifying relevant *system properties* (like safeness, or liveness) expressed in temporal logic by model checking;
- of selecting test data capable of achieving pre-defined *interaction coverage* criteria (e.g. state-based interaction coverage [9], or coupling coverage [1], [4]).

As already indicated above, the main challenge of truly autonomous systems lies in their lack of synchronicity, which is reduced to a minimum degree. Due to this fundamentally asynchronous behavior, the behavior of autonomous cooperative systems cannot be captured by mere communication of state machines exchanging messages for the purpose of triggering particular specific events.

As mentioned above, autonomy should involve as little communication as possible; in particular, it should not assume the existence of common external events aimed at synchronizing their parallelism. In order to allow for as much free concurrency as ideally possible, more powerful notations are required, as offered by Petri Nets. Existing verification activities include

- *static reachability analysis* techniques, aimed a.o. at the early identification of deadlocks or unsafe behavior;
- *dynamic analysis* techniques based on the simulation of different scenarios.

As mentioned in the introduction, an increasing trend towards systems-of-systems is observable, consisting of co-existing and co-operating, de-centralized, autonomous sub-systems originating from different contexts. Here, the verification of their appropriate interplay does not require a mere *integration* perspective, but rather a view of their *interoperability* ensuring

- fulfillment of *consistency* constraints on relations between decentralized values,
- constraints on reading and writing operations between sub-systems required for information *confidentiality* or data *integrity*,
- provision resp. release of *common resources* required for *availability* targets.

While integration testing of component-based systems could progress thanks to a number of novel and automatic test data generation techniques, interoperability testing of autonomous systems still poses crucial challenges to the systems engineering community. Some ongoing work aimed at providing a systematic procedure for capturing the multiplicity of interoperation scenarios to be covered by tests will be presented in chapter 4. The considerations presented above about verification approaches evolving with the degree of autonomy are summarized in Table 1.

Table 1 Evolution of structural analysis techniques

evolution stage	static analysis	dynamic analysis
monolithic units	data flow, anomalies	unit testing, code coverage
component-based system	invariants, inconsistencies	integration testing, interaction coverage
autonomous systems	reachability, deadlocks	interoperability testing, interoperation coverage

3.2 *Quantitative Verification by Statistical Evidence*

Similarly to

- *qualitative approaches* addressing fault detection, also
- *quantitative approaches* addressing reliability assessment

are evolving together with the increasing role of interplaying system parts.

For monolithic systems, reliability may be conservatively assessed by means of statistical tests based on

- a preliminary *hypothesis* claiming an upper bound on the unknown failure probability;
- *operationally representative and independent tests* carried out to such a length as to enable the acceptance or the rejection of the hypothesis at any given confidence level.

If too demanding, the required tests may be replaced by a collection of operational evidence. In such a case, operative experience is to be filtered such as to fulfill all assumptions concerning independent and operational representative test data and test runs, as required by statistical sampling theory. The application of this technique to a software-based gearbox controller is illustrated in [6].

As soon as the logical complexity and execution frequency of component interactions tend to dominate over system behavior, black-box statistical testing approaches as the one proposed above may suffer from bias in case the reliability estimation is based on one-sided evidence missing to capture fundamental interactions. In order to prevent this from occurring, the originally monolithic, black-box approach was extended to include a grey-box perspective by granting – in addition to the demands of statistical sampling – also coverage w.r.t. pre-defined coupling criteria as proposed in [1] and [4], giving raise to an interaction-driven variant of statistical testing [7].

Finally, in case of autonomous systems, an analogous extension is due, aimed at capturing a systematic and adequate choice of scenarios involving asynchronous behavior.

The evolution of quantitative techniques is summarized in Table 2.

Table 2 Evolution of statistical evidence techniques

evolution stage	statistical testing stategy
monolithic units	black-box statistical testing
component-based system	interaction-driven statistical testing
system of autonomous systems	scenario-driven statistical testing

4   Modeling Autonomous Systems by Colored Petri Nets

Several formal notations may be considered as modeling languages for cooperating autonomous systems. A comparative analysis in the light of their expressive power, scalability, analyzability and tool support revealed Colored Petri Nets ([2], [3]) as a notation among the most promising ones; therefore, it was selected for further investigations.

A Colored Petri Net CPN is a 9-Tuple ($P, T, A, \Sigma, V, C, G, E, I$), where

- P denotes a finite set of so-called *places* $p \in P$;
- T denotes a finite set of so-called *transitions* $t \in T$, with $P \cap T = \varnothing$;
- $A \subseteq P \times T \cup T \times P$ denotes a set of directed *arcs* connecting either places with transitions or transitions with places;
- Σ denotes a finite set of non-empty, so-called *color sets*;
- V denotes a finite set of variables v with type$[v] \in \Sigma$ for all $v \in V$;
- C: $P \to \Sigma$ denotes the so-called *color function* attaching to each place $p \in P$ a color set C(p) consisting of so-called *colors*; $C(p)_{MS}$ denotes the multi-set over C(p) consisting of sets of colors, where each color member may occur more than once;
- G: $T \to EXP_V$ is a function attaching to each transition $t \in T$ a so-called *transition guard,* i.e. an expression G(t) over variables $v \in V$ with type$[G(t)] = $ Bool;
- E: $A \to EXP_V$ is a function attaching to each arc $a \in A$ a so-called *arc expression* E(a) over variables $v \in V$, with type$[E(a)] \in C(p)_{MS}$, where p is a place connected with arc a; variables in outgoing arc expressions of a transition are also in ingoing arc expressions of the same transition;
- I: $P \to C(p)_{MS}$ denotes a function attaching to each place $p \in P$ a so-called *initial marking* M(p) of type $C(p)_{MS}$.

A transition $t \in T$ with input places p_i and input arcs a_i: $p_i \to t$, $i \in \{1,\dots,k\}$ is enabled if and only if – after binding each variable $v_{ij} \in V$ in an input arc expression E(a_i) with a corresponding color $c_{ij} \in C(p_i)$, $1 \leq j \leq r(i)$, $r(i) \leq$ Card(C(p_i)):

- the value of G(t) w.r.t. the underlying variable binding is true and
- for each color c_{ij} the value of E(a_i) is not higher than the number of tokens of same color in p_i.

After firing of transition t w.r.t. colors c_{ij}, a new marking is obtained from the previous marking by

- removing from each input place p_i as many tokens of color c_{ij} as resulting by evaluating the input expression E(a_i) w.r.t. the firing-specific variable binding;
- adding to each output place p_o connected to t by an output arc a_o: $t \to p_o$ tokens in number and color as resulting by evaluating the output expression E(a_o) w.r.t. the firing-specific variable binding.

4.1 Example: Robot Factory

The following example concerns the movement of an arbitrary number of robots (modeled by a color set RB) through a narrow lane consisting of an arbitrary number of segments (modeled by a color set SEGMENT). A central controller sends orders (r, s, scurrent) to a robot r providing it with a mission by indicating its target segment s and its initial position 'scurrent' ≤ s. Robots must achieve this target as autonomously as possible by

- using their optical sensors to decide whether the next segment is free; i.e. segments where no other robot or further obstacles are visible);
- if free, moving to the next segment;
- if not, trying again to look ahead;
- enabling an alert after a pre-defined number of cycles.

Though having been intentionally restricted to a relatively simple functionality (namely exclusively forward movement, no workaround of obstacles, no deterministic time-out), it is felt that the simple example shown in Figure 1 (edited by CPN Tools [3]) already provides valuable insight into the problem complexity by supporting a compact and scalable representation of a high variety of scenarios.
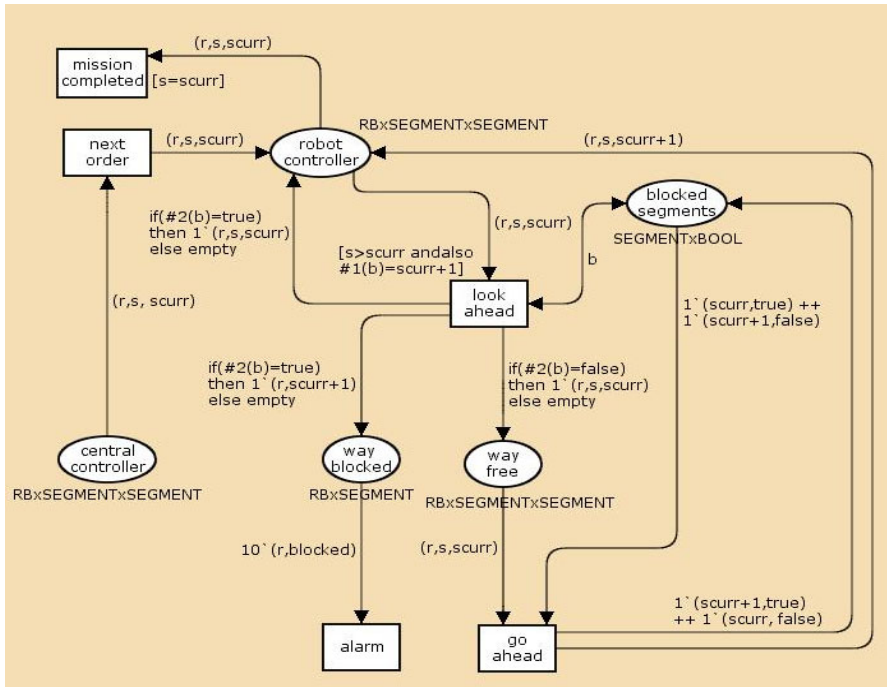

Fig. 1 CPN Model of Robot Factory

In fact, the modeling notation chosen supports the following aspects:

- a compact representation of complex systems, consisting of cooperating autonomous entities, by providing the expressive power offered by token colors and transition constraints;
- high scalability, thanks to the possibility of extending the number of robots and / or segments by adding token colors, but without changing the model structure;
- apart from the initialization phase, robots do not need any global knowledge; on the contrary, each of them contributes to update the local knowledge stored in the marking of place 'blocked segments' by indicating the effects of its own movement.

For the purpose of testing CPNs, it is felt that the few existing approaches ([11], [12]) on testing coverage criteria must be extended in order to capture global reachability and scalability properties. Once novel, appropriate coverage metrics are defined on this basis, it is planned to apply multi-objective evolutionary techniques for the purpose of supporting the automatic test data generation tools for autonomous systems, as already successfully done for architectures involving tighter component coupling ([8], [9]).

5   Conclusion

This article proposed an analysis of classical and novel verification techniques evaluating their evolution in the light of the increasing level of autonomy of system parts. Both qualitative fault detection approaches and quantitative reliability assessment approaches require new extensions and adaptations to fit the growing degree of decentralization and autonomy of modern 'systems of systems'.

References

[1]   Alexander, R.T., Offutt, A.J.: Coupling-based Testing of O-O Programs. Journal of Universal Computer Science 10(4) (2004)
[2]   Jensen, K.: Coloured Petri Nets: Basic Contents. Analysis and Practical Use. Springer, Heidelberg (1997)
[3]   Jensen, K., Kristensen, L.M.: Coloured Petri Nets - Modelling and Validation of Concurrent Systems. Springer, Heidelberg (2009)
[4]   Jin, Z., Offutt, A.J.: Coupling-based Criteria for Integration Testing; Software Testing. Verification & Reliability 8(3) (1998)

[5] Jung, M., Saglietti, F.: Supporting Component and Architectural Re-usage by Detection and Tolerance of Integration Faults. In: 9th IEEE International Symposium on High Assurance Systems Engineering (HASE 2005). IEEE Computer Society Press (2005)

[6] Kreowski, H.-J.: Modeling Iteracting Logistic Processes by Communities of Autonomous Systems, talk on activities of DFG-SFB 637 (Selbststeuerung logistischer Prozesse). In: Chinese-German Symposion, Braunschweig (2010)

[7] Meitner, M., Saglietti, F.: Software Reliability Assessment based on Operational Representativeness and Interaction Coverage. In: 24th International Conference on Architecture of Computing Systems, ARCS 2011. VDE Verlag (2011)

[8] Oster, N., Saglietti, F.: Automatic test data generation by multi-objective optimisation. In: Górski, J. (ed.) SAFECOMP 2006. LNCS, vol. 4166, pp. 426–438. Springer, Heidelberg (2006)

[9] Saglietti, F., Pinte, F.: Automated Unit and Integration Testing for Component-based Software Systems; Dependability and Security for Ressource Constrained Embedded Systems (D&S4RCES). ACM Digital Library (2010)

[10] Söhnlein, S., Saglietti, F., Bitzer, F., Meitner, M., Baryschew, S.: Software Reliability Assessment Based on the Evaluation of Operational Experience. In: Müller-Clostermann, B., Echtle, K., Rathgeb, E.P. (eds.) MMB&DFT 2010. LNCS, vol. 5987, pp. 24–38. Springer, Heidelberg (2010)

[11] Zhu, H., He, X.: A Theory of Testing High-Level Petri Nets. In: 16th IFIP World Computer Congress (2000)

[12] Zhu, H., He, X.: A methodology of testing high-Level Petri nets. Information and Software Technology 44 (2002)

Administration of P2P Systems Based on Application Contexts

Maytiyanin Komkhao, Sunantha Sodsee, Nanthachai Poolketgij,
and Herwig Unger

Abstract. Data derived from application contexts may be used to determine the positions of peers in 2- or 3-dimensional space. The coordinates thus obtained as well as corresponding direction and distance information may help, in turn, to improve routing of and searching for data. Since peers are present at only few positions in a given space, a new, effective algorithm is introduced to establish correct neighbourhood connectivity and a tessalation of the full space by the existing peers. For searching and routing it uses considerably more alternative paths through a structure than known algorithms. Simulations revealed its capability to cope faster and in approximately constant time, independent on network size, with changes in P2P systems brought about by, e.g., newly joining, leaving or failing nodes.

1 Introduction

1.1 From Client-Server to P2P

P2P overlay networks are well-known for decades and applied in large-scale file sharing applications. P2P also goes beyond the centralized organization like client-server systems. In client-server architecture, the problem takes place when clients send many requests to a server, which is the exclusive provider in the system. A huge amount of requests in such a system leads to communication congestion. In

Maytiyanin Komkhao · Sunantha Sodsee · Herwig Unger
Fernuniversität in Hagen, Germany
e-mail: {maytiyanin.komkhao,sunantha.sodsee,herwig.unger}@
    fernuni-hagen.de

Nanthachai Poolketgij
Nakhon Sawan Rajabhat University, Thailand
e-mail: nanthachai@student.nsru.ac.th

P2P overlay networks, each peer can act as a server and client in order to provide services and use services simultaneously. There are two types of P2P structures which can be distinguished: unstructured P2P and structured P2P [1, 2, 3].

First, unstructured P2P; Napster is a well-known example. It is based on a decentralized communication model for searching files and a centralized model for locating shared files. To request a file, the name and IP address of the stored target peer are needed to know. This information is retrieved by the peers from an indexing server, which presents a single point of failure and is therefore one of the main issues of this model. Gnutella is the next example, it is based on both a decentralized communication model for searching files and a decentralized model for locating files. A file request will trigger a flooding mechanism to search the network. As a consequence of this, these messages lead to traffic overhead when the scale of the network is increased to large networks [1, 2, 3]. The one main issue in designing P2P overlay network is how to deal with the desired potential of a file sharing system to support a huge amount of contents. In large-scale networks, it is often hard to retrieve a content from a particular peer. To overcome this problem, Content Addressable Networks (CAN) employ indexing schemes in order to map file names to peers to provide a more reliable method for retrieving shared contents.

Secondly, structured P2P; CAN uses a Distributed Hash Table (DHT) in which communication is based an overlay network architectures. It is designed for scalability, fault tolerance and self-organizing function. Many decentralized look-up protocols use the functionality of hash tables for instance Chord, Pastry, and Tapestry. CAN works on a virtual multi-dimensional Cartesian coordination space. For example, a data object is hashed to pointer P (identifier in the zone) into x-y coordinates space , then the pointer P will be indexed into the peer that responsible for that zone. Herein, pointer P is a new peer for the following description. To create CAN, three main phases are considered as follows: bootstrap, zone management, and peer joining. In the bootstrap phase, a new peer which is hashed into the network which is the first peer occupying the whole area of the space. In the zone management phase, firstly, a new peer which is hashed into the network randomly sends a request message to the existing peer. Secondly, the existing peer receives the request and forwards the message to its neighbour until reaching to the new peer (the pointer P). Thirdly, the existing node which is responsible for a new peer (pointer P), subsequently separates its zone in half-half with the new peer. In the peer joining phase, first, the new peer discovers IP addresses of itself and its neighbours. Secondly, the existing peer which a new peer (pointer P) lies on, subsequently updates neighbours and zones by itself and its neighbours. Finally, the new peer (pointer P) and the neighbours update neighbours and zones by themselves.

One main issue of CAN is how to deal with the time consuming reorganization process of the zone. Due to binary partition trees-like for merging zone, for example, when a failure of a peer is detected; its zone reorganization process is started - a peer failure will be compensated by another peer whose is the sibling. At that time, its

former neighbours become the neighbours of the sibling peer. A routing table which is handled in each peer and stores information about its neighbours, then it has to be recomputed.

Due to the complexity of peer failure organization and a single point of failure in CAN, RecZONE (a decentralized network for routing messages) is introduced. Its objectives are the following:

- Obtaining a complete grid.
- Avoiding broadcasting.
- Avoiding single point of failures.

1.2   Practical Example

In real-life applications, global data storage solutions which rely on centralized frameworks need to be optimized. Overwhelming databases of RFID information that are recorded from a lots of product data, are considered. Radio Frequency Identification (RFID) and Electronic Product Code (EPC) have been used for optimizing performances in supply chain management and decreased cost in recent years instead of using bar coding [6, 7]. The RFID technology is used in commercial products since 1980. Many firms and supermarkets in US, such as Walmart, have been using RFID for reducing the logistic costs [6, 7] with its benefits, such as reducing cashier crew levels and tracking in warehouse management. RFID tags are placed on products, transmitting the identity data of them, and sharing the information of products to supply chain using radio waves for the purpose of tracking of products.

The EPC is used for mapping item and product by identifying unique numbers. RFID readers read and keep the data of products and transmit the data via air interface to the Application Level Event Middleware (ALE) for filtering data to be sent to other systems. EPC contain information such as the manufacturer location, product number, product lot, and serial number to facilitate sharing data in real time [6]. EPCglobal network for RFID infrastructure has many processes as follows: first, EPC is an identifier of the data in the network. Second, the EPC code which is embedded in a RFID chip is attached to products to provide product's information. Third, RFID reads information of EPC code. Forth, RFID reader forwards the EPC information to the server. Fifth, a middleware system manages and extracts the information in real-time and then forwards information of EPC to an Electronic Product Information System (EPCIS). EPCIS provides a data communication interface for sharing and delivering the information between databases and enterprise applications [6, 8, 9].

In a centralized framework as EPCglobal Network, EPCIS act as central lookup services which also provide relations between EPC and Object Name Service (ONS) – ONS generates logical IP address for identifying EPC information on the products in supply chain. However, EPCglobal Network has to manage amounts

of shared data from multiple readers in real time. Drawbacks of this centralized framework are as follows: single point of failure problem and it does not scale for large database [8, 10].

A distributed platform, as proposed in [8], can get rid of drawbacks present in EPCglobal Network framework. [8] presented an algorithm for RFID information routing in a P2P-based structure, which is called UTH. This platform uses a P2P-based middleware. It also does not need IP addresses from ONS generation anymore. The decentralized UTH algorithm for structure building is aimed at building mesh-like structures. The UTH Algorithm assigns EPC values, which contain 28 bits from manufacturer data and separate EPC domain manager values, in a two-dimensional space which acts as a virtual coordinate space in the Peer-to-Peer overlay network (see further details in [8]). For the assignment of neighbours, eight directions are introduced in Fig. 1.
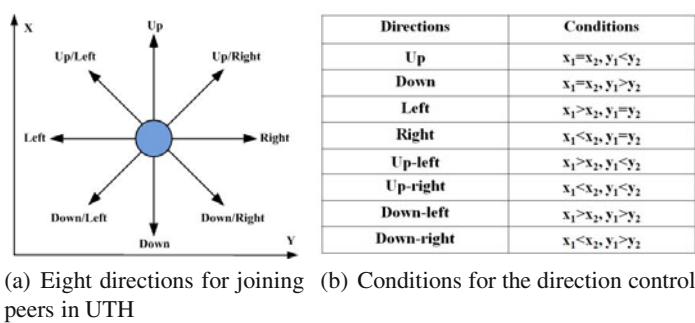

Directions	Conditions
Up	$x_1=x_2, y_1<y_2$
Down	$x_1=x_2, y_1>y_2$
Left	$x_1>x_2, y_1=y_2$
Right	$x_1<x_2, y_1=y_2$
Up-left	$x_1>x_2, y_1<y_2$
Up-right	$x_1<x_2, y_1<y_2$
Down-left	$x_1>x_2, y_1>y_2$
Down-right	$x_1<x_2, y_1>y_2$

(a) Eight directions for joining   (b) Conditions for the direction control
peers in UTH

Fig. 1 Direction control of UTH

In UTH, a mesh-like structure is used, and therefore redundant links will occur. Consequently, a contribution for routing performance might not be very efficient. In addition, the structure created by UTH is an incomplete grid structure. This means, that links of peers still cross each other - this mean that unsuitable connections cause a reduction of search efficiency (see details in Sec. 3.1.1).

From the motivation above, to find a decentralized structure, in which each peer occupies its own zone across the area, this structure will have useful properties for routing and searching. In this work, the block-building shape construction of CAN is considered, and a CAN-like construction algorithm is also applied for building the structure in rectangular-shape while relying on the separation of the x-axis (separating in both x-y axis take too much time for computing the structure) in a distributed manner. By doing this, a complete grid structures will be gained. Additionally, Rec-ZONE provides multiple available paths, eliminates redundant links by creating a complete grid structure, and reduces traffic.

2 RecZONE Algorithm

2.1 Basic Idea

This paper proposes a decentralized network called "RecZONE", which is motivated by CAN as described in Sec. 1.1. The basic idea of RecZONE's design is described as follows: A new peer joins the network by partitioning a zone which is occupied by an existing peer and each peer is responsible for its zone. This way, it can be guaranteed that, there is no space or hole present in the network. Each peer maintains IDs and zones of itself and its neighbours. The scale of the network can be increased to large-scale network. In our approach, the zone allocation differs from CAN because RecZONE does not divide zones into only half-half between the new peer and the closest peer. Moreover, RecZONE takes into account the zone of the neighbours too. The idea comes from the wish to get an "almost fair" zone allocation for zone responsibility. Zone partitioning is being processed in the x-y-coordinate space and the core separation corresponds to the x-axis.

2.2 Details of the Algorithm

Herein, the challenging question is, how does the new peer join the network? There are thress possible cases of joining peers that can be distinguished (the bootstrap phase is not be included in the cases). Case 1: the new peer joins the network when there is just one existing peer. Case 2: the new peer joins the network when it is next to the existing peers. Case 3: the new peer joins the network in between the existing peers.

2.2.1 Possible Cases for New Peers Joining the Network

According to the above-mentioned cases, there are three possible cases for joining peers in the network. First, Fig.2(a) presents Case 1: there is an existing peer in the network. When a new peer joins the network, it sends a message randomly to an existing peer. The existing peer will check if any other peers are closer than it or not, if no one is closer than the existing peer, then, it is set to be the closest peer and defines its zone, as well as the new peer is added to the neighbour list. Later, the closest peer sends the connection message to the new peer and the new peer defines its zone and adds the closest peer to its neighbour list.

Secondly, Fig.2(b) shows Case 2: there are many existing peers and a new peer is joining next to them. At first, a new peer sends a message randomly to the existing peers. An existing peer checks if its neighbours are closer to the new peer than itself or not. If "yes", the existing peer searches for the closest one and then sends a message to it. When the closest peer receives that message, it will check the information about its neighbours. If it has no neighbour, then the closest peer defines
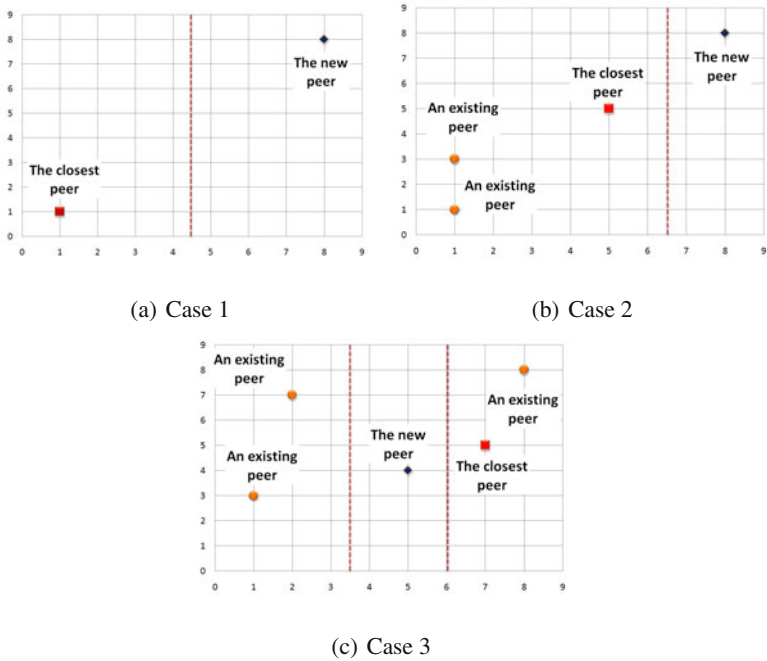
(a) Case 1   (b) Case 2

(c) Case 3

Fig. 2 Possible cases for joining peer (a) Case 1, (b) Case 2 and (c) Case 3

its zone and the new peer is added to its neighbour list. Later, the closest peer sends its neighbour list to the new peer and then creates a link connecting to the new peer. After receiving that information, the new peer defines its zone and adds the closest peer to its neighbour list as well.

Thirdly, Fig.2(c) illustrates Case 3: there are many existing peers and a new peer joins between them. At first, the new peer sends a message randomly to an existing peer. When the existing peer receives the message, it checks if there are any peers closer than itself, then the existing peer calculates the minimum value of distances between itself and its neighbours, after that it forwards the message to the closest one.

When the existing peer receives the message, it checks for the closest peer information in its neighbour. If there is no neighbour which is closer than itself, then it is the closest peer and defines its zone and the new peer be added to its neighbour list. Later, the closest peer sends its neighbour list to the new peer, then the closest peer creates a link connecting to the new peer. When the new peer receives the neighbour lists, it sends a message to the peer within the neighbour lists received from the closest peer. The new peer receives the connection messages and defines its zone and it adds the closest peer to its neighbour list.

The closest peer disconnects from the existing peers which reside on another side of the network and then connects to the new peer again for updating its zone and its neighbours, which is called neighbour list updating. The new peer then receives a connection from the closest peer and defines its zone and then adds the existing peers which disconnected from the closest peer to be the neighbours. Then, the new peer updates both its zone and its neighbours again. Finally, the existing peers connect to the new node and update their zone and their neighbours.

2.2.2   Zone-Construction

Zone-construction in RecZONE is described in the following three main steps:

1. New peers contact and forward the message to the existing peers: check for the closest peer (see Fig. 2(c)). After the new peer $(ID(x_{new}, y_{new}))$ sends the request to any existing peer $(ID(x_{existing}, y_{existing}))$ in the network, the existing peer which is chosen with uniform random distribution value finds the minimum value of the network by calculating the values of the distances of related neighbours from existing peers. The minimum value is gained by calculating the following distances $x_{new} - x_{existing}$ and $y_{new} - y_{existing}$. In Fig. 2(c)), it is shown that the closest peer has the coordinate $(7,5)$.
2. New peers check their related zone: check the value of $x_{min}, y_{min}, x_{max}, y_{max}$ of the closest peer and neighbour peers logically.
3. New peers are being updated and connected with their neighbours: check the member status of neighbour list, which updated accordingly.

To construct such a decentralized network, Fig.3 illustrates the pseudo code for zone-construction in RecZone.

3   Performance Evaluations

In this section, the zone-construction time and routing performances are considered. RecZONE is implemented in Java. P2PNetSim is the tool used to simulate a distributed architecture [11]. The following parameters of network sizes are varied from small-scale to large-scale as follows: 25; 36; 49; 64; 100; 256; 1,000; 4,096 and 10,000 respectively in a two-dimensional coordinate system. The evaluations are conducted after peers have joined the network, in Sec.3.1.2.

For the routing evaluation, the comparison between RecZONE and grid is presented. The percentage of joining peers is significant in this test; so this parameter is varied as follows: 25, 50, 75, and 100 respectively. For the network sizes, our work is simulated for 25, 50 and 100 peers respectively.

The objectives in Sec. 1.1 have been achieved as follows:

- Zones occupancy is represented as rectangular-shapes by peers in order to fully cover the network, therefore a complete grid is obtained.

```
If (Peer was visible)
   If (command is "Hello")
      If it is the closest peer
         Define zone with peer (ID(x, y))
         Add (peer) ID(x, y) to the neighbour list
         Send (Message) connect_message to peer (ID(x, y))
         Send (Message) introduce_message neighbour list to peer (ID(x, y))
         Remove non-neighbour peer(s)
         Send (Message) disconnect_message to non-neighbour peer(s)
         Send (Message) new_zone_message to neighbour peer(s)
      Else forward (Message) message
   If (command is "connect")
      Define zone with peer (ID(x, y))
      Add (peer) ID(x, y) to the neighbour list
   If (command is "introduce")
      Check (peer) ID(x, y) is not in the neighbour list
      Send (Message) hi_message to peer (ID(x, y))
   If (command is "hi")
      Define zone with peer (ID(x, y))
      Add (peer) ID(x, y) to neighbour list
      Send (Message) connect_message to peer (ID(x, y))
      Send (Message) introduce_message the neighbour list to peer (ID(x, y))
      Remove non neighbour peer(s)
      Send (Message) disconnect_message to non-neighbour peer(s)
      Send (Message) new_zone_message to neighbour peer(s)
   If (command is "disconnect")
      Remove (peer) ID(x, y) form the neighbour list
   If (command is "zone")
      Set (zone) new zone of neighbours
      Remove non-neighbour peer(s)
      Send (Message) disconnect_message to non-neighbour peer(s)
   Else forward (Message) message
Else forward (Message) message
```

Fig. 3 Pseudo code for zone-construction in RecZone

- A communication message is sent from peer to peer which avoid broadcasting messages, so the network traffic will be decreased and it also supports large-scale networks.
- Multiple paths are available for efficient routing (more than one path can be selected for routing from the origin to the destination) and to avoid the single point of failure problem, which is visualized in Sec. 3.1.4 (GUI) for creating links from zone-constructions.

3.1 Simulation Results

3.1.1 Comparison between UTH and RecZONE

As mentioned above, the idea of CAN network is employed in RecZONE network. The comparison between UTH and RecZONE is presented in Fig. 4. This comparison is conducted on the network size of 9×9. Figure 4(a) shows the result from the UTH algorithm. For an example, a cross-link from a connection of peer 2 and peer 7 is created. It crosses over a link between peer 3 and peer 4. This cross-link results from the incomplete grid network. Figure 4(b) presents the result from RecZONE, each zone (rectangular-shape) represents an area which is covered by a peer to obtain a complete grid in order to provide multiple paths and routing via adjacent neighbours. The difference between UTH and RecZONE is the connections of peers, which is established to other peers via adjacent bourders in RecZONE. Therefore, unsuitable connections result in redundant links and are eliminated in RecZONE.
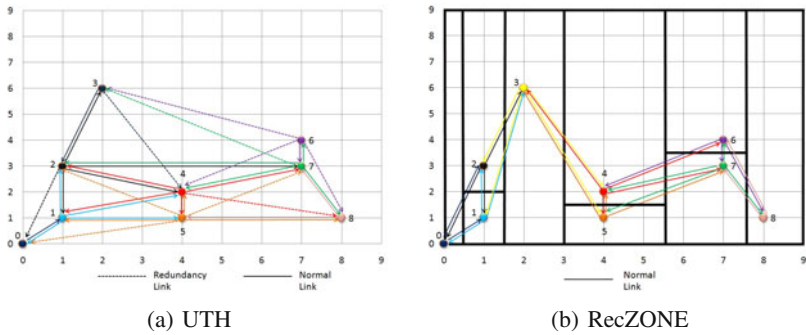

(a) UTH   (b) RecZONE

Fig. 4 Comparison of connection between (a) UTH and (b) RecZONE

3.1.2 Zone-Construction Time

Herein, the simulations are conducted on different network sizes, from small to large network sizes. Figure 5 visualizes RecZONE's constructing time on the network sizes of 25; 36; 49; 64; 100; 1,024; 4,096; and 10,000 when peers have completely joined. It shows the zone-construction times for each network size when generating peers in the networks.

First, zone-construction time is the total time that each peer uses to create its zone. The results show that the zone-construction time of big networks is higher than small networks. This significance comes from numerous hops by which the message is being forwarded from the existing peer until the closest peer is found. Compared to the flooding mechanism in UTH, the number of hops can not exceed the network size. Messages from the flooding mechanism, however, cause traffic overhead in UTH.
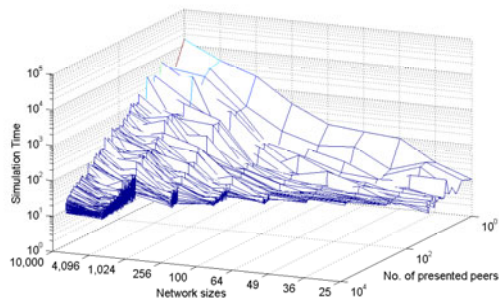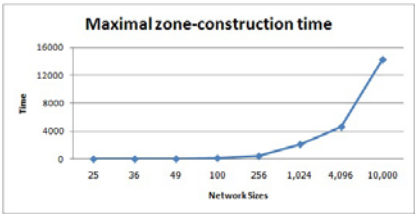
Fig. 5 Zone-construction time of presented peers
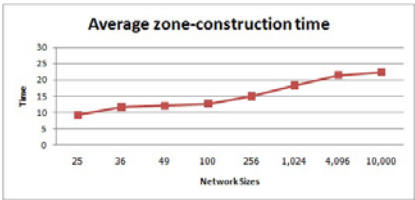

Fig. 6 Maximum zone-construction time

Fig. 7 Average zone-construction time

Secondly, the maximum zone-construction time is considered, Fig. 6 presents the maximum zone-construction time of different network sizes. This figure is obtained from the peak-hop value of each network size. It shows that the network size of 10,000 takes much more time than others and it will take around 14,000 steps. In contrast to that, a small network size, for instance 25, has the zone-construction time 20, only.

Thirdly, Fig. 7 illustrates the average zone-construction time in different network sizes. This figure presents the hop count for each joining peer (the new peer randomly sends the message to the existing one, then the message is being forwarded to the neighbour whose has the minimum distance, which is also the closest peer who

occupies the location that the new peer wants to join). The average zone-construction time is calculated by the sum of the number of hops (hop count) divided by the network size.

When increasing the network size, the consumed time should increase, too. The result shows that the average time for the network size of 10,000 is also higher than for other network sizes and the average time is around 18. On the other hand, for the small network size, for example 25, the average time is approximately 8, only. It can be concluded that the time for both zone-construction time and average-construction time are increased depending on network sizes.

3.1.3 Routing Evaluations

In this section, the objective is to compare the routing performance between Rec-ZONE and the regular grid network. RecZONE takes less time for routing than the grid network because RecZONE creates links between adjacency zones, in which each peer is an identifier for each zone in the network. An assumption in this section is that the routing time in RecZONE should be less than the grid network because RecZONE provides a complete grid structure and multiple paths for routing.

To conduct the simulations, the networks are simulated in different sizes. A peer is added to the network randomly and will define its zone and its neighbours using the zone construction method described. Each peer uses messages for communicating with each other. A message, however, will be sent (from peer to peer) within a time step. A next peer will be added to the network when the previously added peer creates connections between itself and its neighbours completely.

The routing evaluation is established by counting the number of hops while forwarding messages, which are sent from an origin peer to the destination peer. Both origin and destination peers are also chosen randomly from x-y-coordinate space in the network.
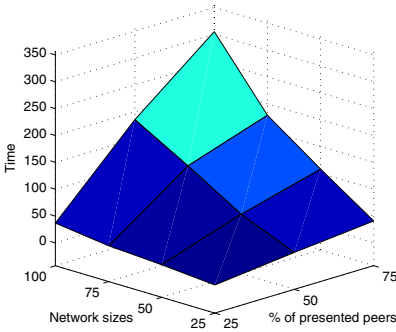

Fig. 8 Routing time comparison

Figure 8 shows the routing time comparison between RecZONE and the grid network. According to RecZONE, the number of fully joined peers is equal to the size of grid network. It means that each peer has not more than four neighbours. The simulations are focused on 25, 50 and 75 percentage of joining peers. The results show that the routing time for a small number of joining peers is much lower than for a large number. It can be seen that RecZONE provides a complete network which it does not waste time for routing via peers whose has not adjacent zones.

3.1.4 GUI of the Algorithm

Figure 9 illustrates the graphical user interface (GUI) of RecZONE. In the example, a network size 5×5 is presented.

(a) peer 2 has joined   (b) peer 3 has joined

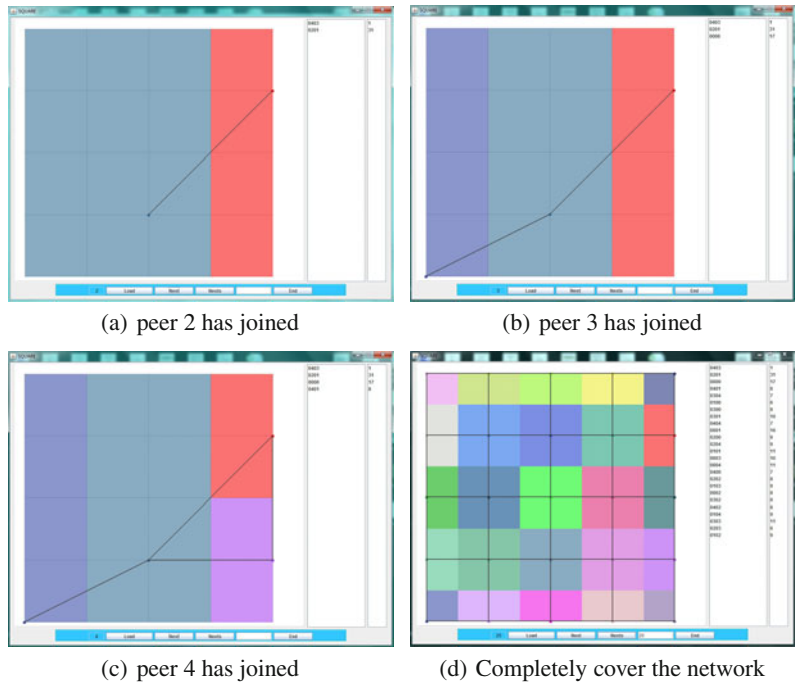(c) peer 4 has joined   (d) Completely cover the network

Fig. 9 GUI shows joining peers and peer IDs as sequences: (a) Peer2 – ID21, (b) Peer3 – ID00, (c) Peer4 – ID41 and (d) Completely joined peers (25 peers)

To visualize network creations from RecZONE, Fig 9(d) shows how RecZONE works for both link-creation and zone-construction. For example, Fig 9(a) shows that the second peer (ID21) joins the network. Then, the third peer (ID00) (See Fig. 9(b)) and forth peer (ID41) (See Fig. 9(c)) are present in the network. Both zones

and neighbour peers are being updated in each time step. A link is created from the existing peers to the newly added peer. Peers can join until the network size is reached.

4 Conclusion

Herein, a decentralized network construction has been proposed in which peers create their zones and establish connections to their neighbours individually. Peers communicate via messages which are sent hop by hop; the scalability of the network is increased as flooding mechanisms are avoided. Neighbours are defined based on zone arrangement in rectangular-shape and multiple paths are available for routing benefits in order to avoid the traffic congestion. As a consequence of this, a single point of failure is also avoided.

For the further work, the construction of balanced zones should be taken into account because each peer should have an equal responsibility for its zone which would benefit load balancing in the network.

References

1. Popescu, A., Ilie, D., Kouvatsos, D.: On the Implementation of a Content-Addressable Network. In: 5th International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks, Karlskrona, Sweden (2008)
2. Lua, E.K., Crowcroft, J., Pias, M., Sharma, R., Lim, S.: A survey and comparison of peer-to-peer overlay network schemes. In: Proceedings of IEEE Communications Surveys and Tutorials, pp. 72–93 (2005)
3. Cholvi, V., Felber, P., Biersack, E.W.: Efficient search in unstructured peer-to-peer networks. In: Proceedings of SPAA, pp. 271–272 (2004)
4. Ratnasamy, S., Francis, P., Handley, M., Karp, R.M., Shenker, S.: A scalable content-addressable network. In: Proceedings of SIGCOMM, pp. 161–172 (2001)
5. Risson, J., Moors, T.: Survey of research towards robust peer-to-peer networks: Search methods. Computer Networks 50, 3485–3521 (2006)
6. Fosso Wamba, S., Boeck, H.: Enhancing Information Flow in a Retail Supply Chain using RFID and the EPC Network: a Proof-of-Concept Approach. Journal of Theoretical and Applied Electronic Commerce Research, Special Issue on RFID and Supply Chain Management 3, 92–105 (2008)
7. Michael, K., McCathie, L.: The Pros and Cons of RFID in Supply Chain Management. In: Proceedings of ICMB, pp. 623–629 (2005)
8. Berg, D., Coltzau, H., Sukjit, P., Unger, H., Nicolaysen, J.G.: Passive RFID tag Processing using a P2P architecture. In: Malaysian Software Engineering Conference (2007)
9. Weinstein, R.: RFID: A Technical Overview and Its Application to the Enterprise. In: Proceedings of IT Professional, pp. 27–33 (2005)
10. Huang, D., Verma, M., Ramachandran, A., Zhou, Z.: EPCA Distributed ePedigree Architecture. In: Accepted by the 11th International Workshop on Future Trends of Distributed Computing Systems (2007)
11. P2PNetSim, User's manual. JNC, Ahrensburg

LaBeeB: Systematic Peer Clustering for Building a Semantic Peer-to-Peer Web Search Engine

Motasem Al Amour

Abstract. Search engines (SE) were only considered in presenting the results of a search query ranked according to a certain algorithm. This means all users will see the same results for the same query (at the same time). Although these users differ in many aspects, this fact was not considered in modern search engines. Peer-to-Peer SEs have tried to imitate centralized ones, with little success. They were faced with massive amount of data, very dynamic structure of the environment and a big number of peers. The real power of a p2p network was not utilized sufficiently, which is the peers themselves. Peers are a representation of human identities in the internet. A peer (or a human) can be categorized by the following criteria: the language it speaks, the country it comes from, the age group it belongs to and the human character it falls under. This very peer in turn has many interests, and so it will visit web pages that match these interests. This information for this peer and million others will be then saved in the p2p network. With proper interpretation of this information a semantic web search engine can be built and a systematic method can be used to rank the result of a query according to the number of visited web pages visited by peers that have the same criteria as the query initiator.

In this paper we present LaBeeB[1], an innovative p2p web search engine that can resolve user queries effectively in a semantic fashion and can then rank the results based on human factors.

1 Introduction

To enable LaBeeB to rate web pages to the number of visits of peers similar to the query initiator, one should be able to measure the *similarity* between peers or

Motasem Al Amour
Faculty of Mathematics and Computer Science Chair of Communication
Networks Fernuniversität Hagen Hagen, Germany
e-mail: almotasem@hotmail.com

[1] An Arabic word means knowledgeable and intelligent.

humans in general. We define similarity as a combination of *language, country, age and personality or character* which are known for each peer and are called *peer's attributes*. We assume peers who share these attributes are very alike. And if these peers also share the same interest in politics for example, then we argue when one of these peers visits a page with a political topic, most probably other *similar* peers will be interested in that page too. The answer is to be found in [19] and [20] where the term *limbic character* is defined. It focuses on our unconscious brain (the limbic system) and how it is affecting our decisions, behavior and all day activities. That part of the brain is responsible for 75% of our decisions [19]. Three basic emotions *balance*, *dominance* and *stimulation* are defined; the combination of these emotions gives one of six limibic characters for every person.

Most approaches for a p2p content search have suggested a Distributed Hash Table (DHT) that holds a global inverted index of (keyword,web address) with some improvements like [**?**] [23] [24] [25] [21]. This design will deliver poor performance for a p2p environment with millions of users, billions of web pages and terabytes of data. Others proposed a semantic approach [8] [7], by adopting a hybrid p2p network where some tasks are assigned to special peers. LaBeeB will not use any dedicated peers because first it is not a wise choice to put the load of a p2p SE on some *super peers* that they could easily be overloaded and second we will use a true p2p network where the load is distributed across all peers.

Although a DHT provides an effective and guaranteed data retrieval, it has the disadvantage of a large maintenance cost to keep it consistent, especially in a highly fluctuating dynamic environment. That is why [18] argues that the cost for maintaining the DHT outweighs the advantage of having one in the first place as most time is spent on updating DHT indices. LaBeeB will overcome this disadvantage by first reducing the amount of data to be stored in the DHT by adopting another form of the global (keyword,web address) inverted index, and second by clustering peers and so having not one but several DHTs. With this design a DHT size is reduced drastically and thus the maintenance cost is minimized.

Text Categorization or Text Classification (TC) [14] [15] [16] is the task of automatically sort a set of documents (in our case a web page) into one or more categories based on the document's content. Thus TC is a method to categorize texts semantically. TC is formalized by the function $\phi : D \times C \rightarrow True, False$ where D is a set of documents and C is a predefined set of categories.
LaBeeB categorizes web pages using TC techniques, by using this principle a web page is indexed by its category rather than by all keywords it contains. This means we can replace a global huge (keyword, web address) inverted index with a smaller semantic (category, web address) one.

The rest of the paper is organized as follows. Sections 2 presents the p2p search engine architecture. Section 3 illustrates peer clustering. We conclude with a discussion of what we have accomplished.

2   The P2P Search Engine

Search engines are the most visited websites in the internet. Only in the United States -which has about 227M internet users [11]- around 80M and 75M users have visited Google and Yahoo! daily. Applying these numbers to all internet users of 1700M worldwide [11] one can estimate 780M visits to Google each day. In addition to that the number of web pages indexed by Google can now only be estimated to 15 billion [10] (Feb 2010). For 1000 word in each web page we can estimate the number of document identifiers of Google's inverted index to be 15×10^9, with millions of web pages added to the index daily[2]. Assuming a 2% of internet users participate in LaBeeB. The overlay network will have up to 34M users, from those we assume one tenth are active peers in the p2p network at any time. We also expect a high rate of peers join/leave the network and about 60M search requests each day or around 700 search requests/second.

An important requirement for a search engine is an effective search performance and guaranteed search results. For unstructured protocols like Gnutella [3] it is not the case, ([4] and [5] enhance the search, but with the additional cost of replicating the data). Structured protocols like Chord[1] and CAN[2] overcome this downside by using a DHT, with a guarantee to find any data stored in the DHT with $log(N)$ hops where N is the number of peers in the network. But this is only guaranteed with a more complex structure and a large maintenance cost to keep the DHT consistent [18]. LaBeeB will use a DHT to store the required data in the p2p network. But we will present in this and next section methods that reduce the maintenance cost of the DHT and the p2p network. The DHT can mainly do two operations store a [key,value] pair referenced with the function η, and retrieve a value of a certain key referenced as function θ.

LaBeeB will use a structured protocol with the assumption that data replication is in place, so when a peer fails, the key,value pairs it holds are not lost that they are replicated on some other node(s). An Example of Chord with data replication can be found out in [6].

A peer p who can speak languages $L = \{l_1, l_2...l_n\} \subset \Lambda$ where Λ is a set of all languages, and comes from country $y \in Y$ where Y is a set of all countries, and belongs to age group $a \in A$ where $A=\{$young age,college age,early work age,Late work age,elderly age$\}$ and has a limbic character $i \in I$ where $I=\{$adventurer, performer, discipliner, hedonist, epicure, traditionalist, harmonizer$\}$ has a set of attribute α defined as the following:

$$\alpha = \{L, y, a, i\} \tag{1}$$

Throughout this document we will refer to a peer with attribute α as p_α and the attributes of peer p as α_p.

p_α also have several interests that fall under category $c_p := \{c_1, c_2..., c_n\} \subset C$ where C is the set of all categories.

---

[2] Numbers are taken in February 2010.

Each web page $w \subset \Psi$ where Ψ is a set of all web pages can be categorized in categories $c_w = \{c_1, c_2 ..., c_m\}$. p_α visits a set of web pages w_p where most probably those web pages falls under the interest of p, in other words $c_{p_\alpha} \subset c_{w_p}$. The term interests and category will be used interchangeably.

2.1 Crawling and Indexing

Each peer holds a local crawler who has three tasks. First as p_α visits page w the TC calculates its all possible categories $c_w \subset C$. The crawler can decide if p_α has interest $\chi \in c_w$ if p visits pages W within a time period t such $\forall \omega \in W : \chi \in c_\omega$, and so can the crawler detect the interests of p_α or c_{p_α}. This task can also be extended to remove interest χ from c_{p_α} if in W visited pages $\forall \omega \in W : \chi \notin c_\omega$. The second task is to index each page visited by [$keyword, web\ address$] localy by building the inverted index $I_{[k,w]}$. The last task of the crawler is to inform the p2p network that this page has been visited by a peer p with attributes α.

The information delivered by the crawler in its first and third tasks are to be indexed in the p2p network. LaBeeB must be capable to determine peers that have a certain category and the number of visits for each website visited by peers with a certain attribute α. That is accomplished by building two global inverted indices shared over a DHT. $I_{[c,p]}$ holds [$category, peer$] pair where the key is category and the value is the peer (peerID,α_p) and $I_{[w,v]}$ holds [$webPage\ address, visits$] where the web page is the key and the value is the number of visits for every α. When p_α visits a new page, $I_{[c,p]}$ is not changed except in the case if the crawler decides that a category c is to be added to (or removed from) c_{p_α}.

$$\eta_{I_{[c,p]}}(c, p_\alpha) \tag{2}$$

While $I_{[w,v]}$ is always changed whenever a peer with attribute α visits page w, where the peer p_w holding the key w increments the number of visits for peers with attribute α by one.

$$\eta_{I_{[w,v]}}(w, \alpha) : v_{w,\alpha} = v_{w,\alpha} + 1 \tag{3}$$

But it is still to be noticed the size of $I_{[c,p]} + I_{[w,v]}$ is much smaller than the size of a global [$keyword, web\ address$] inverted index. We will see in the next section how we can reduce the size of the indices stored over the DHT and how to minimize the load on $I_{[w,v]}$. Figure 1 summarizes the indices used by LaBeeB.

2.2 Searching and Ranking

When peer r with attributes α initiates a query $q = \{k_1, k_2, ..., k_q\}$ of keywords k, the local TC will compute the category of that query c_q. Then a DHT lookup to the index $I_{[c,p]}$ will be issued to retrieve a set of peers p_c as Equation 4

$$p_{c,\alpha} = \{p_1, ..., p_n, \} = \theta_{I_{[c,p]}}(c_q) : \forall p \in p_{c,\alpha}, \ \alpha_r = \alpha_p \tag{4}$$
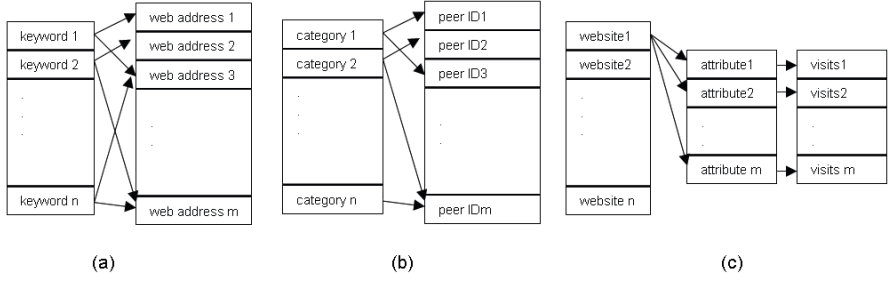
Fig. 1 General overview of LaBeeB's Indices (a) $I_{[k,w]}$ where the keyword k is the key for the local index (b) $I_{[c,p]}$ where the category c is the key in the DHT (c) $I_{[w,v]}$ where the website address w is the key in the DHT

Where n is a predefined number of active peers having the *same* attribute α as the query initiator that must be returned. Each $p \in p_{c,\alpha}$ will lookup the query in its local $I_{[k,w]}$ where a list of web addresses $d_p = \{\delta_1, \delta_2, ..., \delta_m\}$ are attained as Equation 5

$$\forall \delta \in d_p \ \delta = \theta_{I_{[k,w]}}(k_1) \wedge \theta I_{[k,w]}(k_2) \wedge ... \wedge \theta_{I_{[k,w]}}(k_q) \tag{5}$$

d_p are then sent to r. The query initiator computes $d = \sum d_{p_1}, d_{p_2}, ..., d_{p_n}$ and omits the redundancies. At that part searching for the query is complete and each address in d contains the keywords of q. The results are then ranked by the number of visits for each website carried out by peers who have the same attributes α as p_α. Therefore index $I_{[w,v]}$ is looked up as in Equation 6

$$\forall \delta \in d \ v_{\delta,\alpha} = \theta_{I_{[w,v]}}(\delta_{website\ address}) \tag{6}$$

The number of visits $v_{\delta,\alpha}$ presents the number of visits of peers with the same attribute α as the initiator. The web addresses of δ with the number of visits $v_{\delta,\alpha}$ are then provided to r_α sorted in a descending order according to $v_{\delta,\alpha}$.

Algorithm 1 summarizes the searching and ranking of LaBeeB.

It should be noticed that LaBeeB can deliver results even if the DHT is not consistent, for example when the churn rate (peer join/leave) is high. The $p_{c,\alpha}$ returned in Eq.4 is only a subset of peers with attribute α and have the key c of the set $P_{c,\alpha}$ or $p_{c,\alpha} \subset P_{c,\alpha}$. Thus if $P_{c,\alpha}$ is not fully consistent we still able to get a set of peers $p_{c,\alpha}$. The same can be applied to $v_{\delta,\alpha}$ where this number servers for the orientation of how popular a page is and a variation of $\pm 10\%$ is acceptable.

3   Peer Clustering

A study in [26] showed that an average American internet user spent 68 hours online per month and visits nearly 2700 websites and spent 57 seconds in average per page.

Algorithm 1. The search and ranking algorithm of LaBeeB

1: peer r initiates query q {Carried out on r}
2: $c_q = TC(q) = \{c_{q_1}, c_{q_2}, ..., c_{q_k}\}$
3: $p_c = lookup_{I_{[c,p]}}(c_q)$ {Eq. 4}
Require: $|p_c| = n : n > 0$
Require: $\forall p \in p_c,\ \alpha_r = \alpha_p$ {Carried out on every $p \in p_c$}
4: **for all** $p \in p_c$ **do**
5:     $d_p = lookup_{I_{[k,w]}}(q)$ {Eq. 5}
6:     $r \leftarrow d_p$
7: **end for**{Carried out on r}
8: $d = sum(d_{p_1}....d_{p_n})$ {ignore redundant web pages}
9: **for all** $\delta \in d$ **do**
10:     $v_\delta = lookup_{I_{[w,v]}}(q)$ {Eq. 6}
11: **end for**
12: present the pair δ and v_δ sorted descendingly by v_δ

For our assumption of 3,4 million active peers in LaBeeB's p2p network at any time with one minute spent on each page, this means more than 56,000 pages are visited per second by all peers. According to Equation 3 the index $I_{[w,v]}$ is accessed by more than 56,000 times/second. One may argue that the load is distributed equally across all peers in a DHT or each peer gets $56 \times 10^3 / 34 \times 10^5 \approx 0.017$ of the overall load, but this is not the case. The internet consists of very popular and unpopular pages. If one peer holds the key of a popular internet address with thousands of visits every second then it can be overloaded.

This is not the only deficiency of the p2p network. The maintenance cost to keep the DHT consistent is every high, especially when the churn rate is high (the rate at which peers join or leave the network). That is why [18] argues that the cost of maintaining a consistent DHT outweighs the advantages of having one. Another consideration is the number of hops required to retrieve or store data which is $O(logN)$.

As a solution we will not adopt a flat p2p model where all peers are grouped in one network. We will adopt a dynamic and hierarchal clustering method. As already noticed when describing the design of a global name system "Hierarchy is a fundamental method for accommodating growth and isolating faults" [17]. The advantage of clustering is presented in Theorem 1.

Theorem 1. *Reducing the number of peers by grouping them into clusters we reduce the maintenancecost, the load on peers holding indices of popular pages and the number of hops*

Proof. When the p2p network consists of x clusters then the churn rate is reduced by almost the same amount in each cluster so reducing the maintenance cost of the DHT in each cluster. Popular indices are held by up to x peers, so distributing the load. The number of hops in each cluster is reduced by the amount $log(x)$ where $log(\frac{N}{x}) = log(N) - log(x)$. □

Clusters in LaBeeB are made to ease the task of the p2p web search engine in in-
dexing, searching and ranking. This can be achieved by the following:

1. With high probability, each cluster should be able to answer a query issued by
   a peer belonging to it locally without the need to transfer the query to other
   clusters.
2. The DHT in each cluster is kept as small as possible to reduce the maintenance
   costs but yet able to answer local queries. We are mainly interested in optimizing
   the index $I_{[w,v]}$.

To accomplish the previous two points we group similar peers in a cluster. The
reason for this comes from the fact that similar people share a notable percentage P_V
of the web pages they visit. For LaBeeB this means indexing is reduced by the same
amount of P_V, so reducing $I_{[w,v]}$. The same concept can also be applied on searching,
a notable percentage of P_Q of the issued queries by similar peers in a cluster can be
answered with web pages already accessed by peers in the same cluster. That means
P_Q percentage of the queries are to be found in the cluster indices, without the need
to route the query to other clusters.

To summarize, clusters should be built such that $\frac{P_V}{n}$ and $\frac{P_Q}{n}$ are as great as possi-
ble where n is the total number of active peers in the cluster. So the key point is to
define the elements that plays an important role in terms of web surfing for the mea-
surement of similarity between peers. Because the clusters in a hierarchal fashion,
those elements should be prioritized; where the elements with the highest priority
sit on top of the hierarchy and those with lower priority sit beneath.

Similarity was already defined in Eq. 1, these elements of *language, country, age
group and limbic character* are also used for cluster building, the priority is defined
by the 4-tuple in Eq. 7:

$$\overline{\alpha} = (l, y, a, i) \tag{7}$$

From Eq. 7 one can conclude that there are four levels of hierarchy one for every
element in $\overline{\alpha}$. A hierarchy level v can be defined as:

$$
v = \begin{cases}
0 & \text{Cluster is built according to language} \\
1 & \text{Cluster is built according to country} \\
2 & \text{Cluster is built according to limbic character} \\
3 & \text{Cluster is built according to age group}
\end{cases}
$$

The peers in a cluster with a certain hierarchy level must have the same attribute as
the cluster and its parents (elements in $\overline{\alpha}$ preceding the current level).

$$\forall p \in u, \ \forall h : 0 \leq h \leq nu_u \ \overline{\alpha}_u(h) \in \alpha_p \tag{8}$$

Clustering by Language

Peers speaking different languages will visit web pages in different languages and
hence the indices they generate are language bound. So grouping peers according

to the language they speak will reduce the cluster size and optimize the DHT to contain only data related to that language. At the same time we are sure that the any query in that language can be answered (if the answer is indexed in the p2p network). Language is the first element in $\overline{\alpha}$ and so language clusters sit on top of the hierarchy. That means there is no one root or global cluster, instead, there are as many root clusters as the number of languages spoken by LaBeeB's active peers. We can notice that P_Q and P_A are very high in this level of hierarchy, but for certain languages the number of peers in each cluster is still grand, meaning $\frac{P_V}{n}$ should be increased by reducing the number of peers in the cluster. Therefore clustering in the next level is needed.

Clustering by Country

Languages like English, Arabic, Spanish, French and others are the official language in many countries. The idea is, for a language cluster when the number of peers for a certain country increases a new cluster based on Language-Country is formed where all peers belonging to that country leave the language cluster and enter the new one. When the number of peers for a Language-Country decreases, that cluster will be dissolved in the parent one where all its peers will join the language cluster and then the child cluster will be destroyed. Does grouping peers according to their country reflect a common behavior in surfing the internet for peers that speak the same language but are in different countries? In other words do $\frac{P_A}{n}$ and $\frac{P_Q}{n}$ increase remarkably for a Language-Country cluster. If this argument is true this will mean that the DHT in the Language-Country group is more optimized than the Language one and we have less number of peers than the parent cluster. To prove our argument we have compared the first 20 most visited web pages of three Arab countries Jordan, Syria and Morocco (sites for search engines, social networks, blogs, file sharing and porn have been excluded from the list). We choses these three countries due to similarities in language, culture and religion. We found that only 5/20 websites are the same for both Syria and Jordan and only 3/20 with Morocco.

We believe that country plays the second most important role in defining similarity between peers, therefore it is the second element in cluster building.

With country grouping we have the benefit that peers in one country sit in the same or in adjacent Autonomous Systems (AS). That means the routing and communication overhead between the peers in the same Language-Country group (and any descended group as to be seen later) is reduced. The same concept was introduced in [9] where peers with the same AS number are clustered.

Clustering by Age

Age plays an important role on our interests and the type of website we prefer to visit. According to [13] 70% of teenagers in the age 18-21 use social network sites while less than 6% of seniors use such sites. Table 1 [12] lists internets on the internet by generation in the USA.

Table 1 Activities by generation in the USA (2009)

Activity	12-17	18-32	33-44	45-54	55-63	64-72	73+
Younger are more likely to engage in the following activities							
Play games online	78	50	38	26	28	25	18
Watch videos online	57	72	57	49	30	24	14
Get info about a job	30	64	55	43	36	11	10
Send instant msgs	68	59	38	28	23	25	18
Social networking	65	67	36	20	9	11	4
Download music	59	58	46	22	21	16	5
Read blogs	49	43	34	27	25	23	15
Create blogs	28	20	10	6	7	6	6
Visit a virtual world	10	2	3	1	1	1	0
Activities where older dominate							
Get health info	28	68	82	74	81	70	67
Buy online	38	71	80	68	72	56	47
Bank online	-	57	65	53	49	45	24
Visit govt sites	-	55	64	62	63	60	31
Get religious info	26	31	38	42	30	30	26

Age is the third attribute to measure peer's similarity and the third criteria for cluster building, the reason for that can be concluded from Table 1. We will use 5 age groups: Young (≤ 18), College (19-29), Early Work (30-44), Late Work (45-59) and Elderly age (≥ 60). When the number of peers belonging to a certain age group in a Language-Country cluster increases, a new cluster based on Language-Country-Age will be created. Affected peers will leave the parent cluster and join the new one. And vice versa when the number of peers in a Language-Country-Age cluster decreases where it will be dissolved into the parent.

We expect that $\frac{P_Q}{n}$ and $\frac{P_A}{n}$ in a Language-Country-Age cluster is greater than the one of a Language-Country one.

Clustering by Limbic Character

Researches on limbic character have focused on the correlation to marketing and consumption. For instance [20] lists the following as a common behavior between persons with the same limbic character.

- Persons of type adventurer show the most interest in computer products, followed by hedonists and performer, while harmonizer and traditionalists showed no interest at all.
- Performers and Adventurers showed the highest interest in Automobile, while Harmonizers and Discipliners showed the lowest.
- Discipliners followed by sustainers are masters in buying garden products, hedonists are the worst.

- Discipliners, Harmonizers and Traditionalists are the *conservatives*. Epicures, Hedonists, Adventurers and Performers are the *moderns*
- Hedonists, Adventurers and Performers are the *actives*. Epicures, Discipliners, Harmonizers and Traditionalists are the *passives*.

Our idea is to use the limbic character as a representation of internet users' personality. It will be used as the fourth and last criteria to form clusters. The creation and dissolution procedure for a Language-Country-Age-Limbic cluster is the same for clustering by country and age. The expectation is $\frac{P_Q}{n}$ and $\frac{P_A}{n}$ in a Language-Country-Age-Limbic cluster is greater than the one of a Language-Country-Age one.

Figure 2 shows how the English speaking peers can be clustered based on our clustering method. Each circle presents a cluster, the fixed lines do not present a connection between the clusters (only a parent child relationship) and the dotted lines present a possible number of clusters in this level of hierarchy. The cluster named Performer in the figure can only be created if 1. The English group exists 2. Peers from USA and cluster English exceed a certain number to create their own cluster 3. Peers belonging to the age group 'Early work age' under the English-USA cluster exceed a certain number of peers and create their own cluster 4. Peers with the limbic character Performer under the previous cluster go over a certain number and create their own cluster.
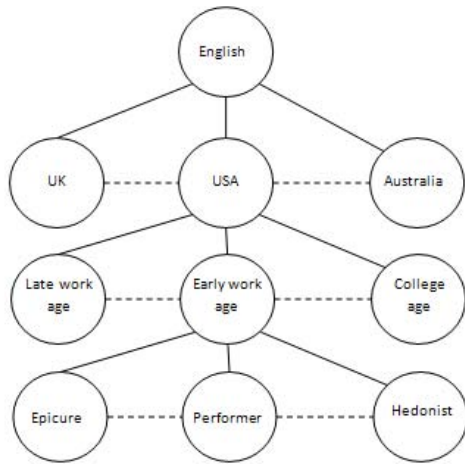

Fig. 2 An example how english speaking peers can be clustered.

4 Conclusion and Future Work

We presented a new approach for web search engine searching and ranking. Searching is performed semantically and a query is ranked based on human attributes. LaBeeB is not an attempt to imitate a centralized web search engine, on the contrary it provides new functionalities and more possibilities that can only be accomplished

in a p2p network. LaBeeB has introduced several techniques to optimize the storage/retrieval in a very dynamic peer-to-peer network by adopting a new search/rank algorithm, decrease the size of the shared indices over the network and distribute the load across semantically created clusters.

In the future we will present algorithms for dynamic cluster creation and dissolution. In addition a simulated framework of LaBeeB is to be developed to evaluate the effectiveness of searching and ranking.

References

1. Stoica, I., Morris, R., Karger, D.R., Frans Kaashoek, M., Balakrishnan, H.: Chord: A Scalable PeertoPeer Lookup Service for Internet Applications. In: SIGCOMM (2001)
2. Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A scalable content-addressable network. In: ACM SIGCOMM (August 2001)
3. Gnutella, `http://gnutella.wego.com`
4. Terpstra, W.W., Leng, C., Buchmann, A.P.: BubbleStorm: Analysis of Probabilistic Exhaustive Search in a Heterogeneous Peer-to-Peer System. In: Technical Report, No. TUD-CS-2007-2, TU Darmstadt (May 2007)
5. Terpstra, W.W., Kangasharju, J., Leng, C., Buchmann, A.P.: BubbleStorm: Resilient, Probabilistic, and Exhaustive Peer-to-Peer Search. In: Proceedings of the 2007 ACM SIGCOMM Conference (2007)
6. Al Amour, M.: Design and prototypical development of a Chord-based Peer-to-Peer Network for a Distributed Virtual Community. Master Thesis, University Dusiburg-Essen (2008)
7. Tang, C., Xu, Z., Dwarkadas, S.: Peer-to-peer information retrieval using self-organizing semantic overlay networks. In: Proceedings of ACM SIGCOMM, pp. 175–186 (August 2003)
8. Li, M., Lee, W.C., Sivasubramaniam, A.: Semantic Small World: An Overlay Network for Peer-to-Peer Search. In: Proc. 12th IEEE International Conference on Network Protocols, ICNP 2004 (2004)
9. Wang, H., Zhu, Y., Hu, Y.: An Efficient and Secure Peer-to-Peer Overlay Network. University of Cincinnati (2005)
10. `http://www.worldwidewebsize.com`
11. `http://www.internetworldstats.com/stats.htm`
12. `http://pewresearch.org/pubs/1093/generations-online`
13. `http://factfinder.census.gov/servlet/STTable?_bm=y&-geo_id=01000US&-qr_name=ACS_2008_3YR_G00_S0101&-ds_name=ACS_2008_3YR_G00_S0101&-ds_name=ACS_2008_3YR_G00`
14. Sebastiani, F.: Text categorization. In: Zanasi, A. (ed.) Text Mining and its Applications, pp. 109–129. WIT Press, Southampton (2005)
15. Joachims, T.: Making large-Scale SVM Learning Practical. In: Schölkopf, B., Burges, C., Smola, A. (eds.) Advances in Kernel Methods - Support Vector Learning. MIT Press (1999)
16. Schapire, R.E., Singer, Y.: BOOSTEXTER: a boosting-based system for text categorization. Machine Learning 39(2/3), 135–168 (2000)
17. Lampson, B.: Designing a global name service. In: Proc. 4th ACM Symposium on Principles of Distributed Computing, Minaki, Ontario, pp. 1–10 (1986)

18. Traversat, B., Abdelaziz, M., Pouyoul, E.: Project JXTA: A Loosely-Consistent DHT Rendezvous Walker. Project JXTA Sun Microsystems, Inc.
19. Häusel, H.-G.: Think limbic! Die Macht des Unbewussten verstehen und nutzen für Motivation. Marketing, Management, Haufe (2009)
20. Häusel, H.-G.: Brain View Warum Kunden kaufen, Haufe (2008)
21. Bender, M., Michel, S., Triantafillou, P., Weikum, G., Zimmer, C.: Minerva: Collaborative P2P Search. In: VLDB (2005)
22. Bender, M., Michel, S., Triantafillou, P., Weikum, G., Zimmer, C.: P2P Web Search with MINERVA: How do you want to search tomorrow?
23. Loo, B.T., Huebsch, R., Stoica, I., Hellerstein, J.M.: The Case for a Hybrid P2P Search Infrastructure. In: IPTS (2004)
24. Tang, C., Dwarkadas, S.: Hybrid global-local indexing for efficient peer-to-peer information retrieval. In: Proceedings of the Symposium on Networked Systems Design and Implementation (NSDI) (June 2004)
25. Podnar, I., Rajman, M., Luu, T., Klemm, F., Aberer, K.: Beyond Term Indexing: A P2P Framework for Web Information Retrieval. Informatica, Special Issue on Specialised Web Search (2006)
26. http://mashable.com/2009/10/14/net-usage-nielsen

On the Performance of Secret Entropy Coding: A Perspective Beyond Security

Shujun Li

Abstract. In this paper, we study the overall performance of two main forms of secret entropy coding – secret Huffman coding and secret arithmetic coding, as solutions to multimedia encryption. We consider a set of criteria, which include not only security but also other aspects of the performance. We draw the conclusion that neither can fulfill all the criteria, but secret arithmetic coding can offer a better solution. We also point out the possibility of amending existing multimedia coding standards to facilitate multimedia encryption.

1 Introduction

To fulfill increasing demands for content protection of multimedia products, multimedia encryption has been extensively studied in the past [9, 38, 10, 44]. Because simply encrypting compressed multimedia data with a textual cipher cannot fulfill requirements of some practical applications, many different encryption techniques have been proposed to design joint compression-encryption schemes. Among all the proposed techniques, secret entropy coding has attracted more attention than others, because there are various ways to integrate encryption into data compression without much additional computational load. In this context Huffman coding and arithmetic coding are the two most widely-adopted entropy coding algorithms in multimedia coding standards, and most research is done on secret Huffman coding and secret arithmetic coding. Though some cryptanalytic results on a number of specific secret entropy coding algorithms have been reported and some security problems have been identified, many aspects of the overall performance of secret entropy coding have not been well understood and a performance comparison between secret Huffman coding and secret arithmetic coding has never been done before.

Shujun Li

Department of Computer and Information Science, University of Konstanz,
Universitätsstraße 10, Mailbox 697, Germany
e-mail: `http://www.hooklee.com`

This paper studies the overall performance of secret Huffman coding and secret arithmetic coding, by considering a set of criteria. Instead of analyzing security only, we extend our focus to the capabilities to support more useful features required in multimedia encryption. As a conclusion, we point out that neither secret Huffman coding and secret arithmetic coding can fulfill all criteria, but the latter can offer a better solution to multimedia encryption as a whole.

The rest of the paper is organized as follows. In the next section, we give a brief survey of multimedia encryption and show a set of criteria about the overall performance of multimedia encryption systems. Then, we apply these criteria to secret Huffman coding and secret arithmetic coding in Section 3, trying to clarify their overall performance and make a qualitative comparison between them. Finally in the last section, we give a short summary and mention the possibility of designing security-oriented multimedia coding systems to facilitate multimedia encryption.

2   Multimedia Encryption

According to the relationship between encryption and compression, there are three possible approaches to design multimedia encryption systems: 1) encryption before compression; 2) encryption after compression; 3) joint compression-encryption.

For the first approach, completely new algorithms have to be devised to ensure efficient compression of encrypted data, because encryption generally leads to a random output with a very high information entropy that cannot be compressed effectively afterwards. A recent solution based on distributed source coding was proposed in [20], by taking the encryption key as side information available at the decoder side. However, for this solution, decoding is generally impossible without the knowledge of the decryption key, which is not desired in some applications requiring format compliance, such as postprocessing without decryption (see below for more details). In addition, this scheme puts some requirements on the encryption algorithm involved such that not all available ciphers can be freely chosen and deployed. Another problem is that the proposed solution is not compatible with existing multimedia coding standards.

The second approach is the most direct and simplest one, which is often called *naive encryption* in the literature [32]. By simply employing a textual cipher in this way, the syntax format of the compressed multimedia data will be destroyed. However, *format compliance* of encrypted multimedia data is very useful in many applications such as the following:

- *postprocessing of encrypted multimedia data without decryption*: watermark embedding, transcoding, rescrambling, bitrate control, repacketization, etc.;
- *perceptual/traparent encryption* [24]: encryption is used to degrade the quality of multimedia data rather than conceal all the information, which is useful for preview-before-pay multimedia services;

- *scalable encryption*: a multimedia product has different resolutions encrypted with different configurations;
- *ROI (region-of-interest) encryption*: only part of a multimedia product is encrypted.

To achieve format compliance, it is obvious that the multimedia data cannot be fully encrypted, i.e., the idea of *selective encryption* (also called *partial encryption*) has to be adopted to leave some syntax elements unencrypted. In addition, special aspects have to be considered in the design of the encryption part such that the encryption process is compatible with the underlying multimedia coding standard. This means that the third approach – joint compression-encryption – should be used instead of the first two ones.

Selective encryption is also very useful to reduce the encryption load, which is especially important for some applications such as video-on-demand systems that need to perform real-time encryption on a large number of videos and send the encrypted videos to a large number of users simultaneously. Selective encryption is also useful to save energy for resource-constrained devices like wireless multimedia sensor networks (WMSNs) [2]. Unfortunately, if too many syntax elements containing perceptual information are left unencrypted, the security might be compromised. A lot of research [1, 39, 31, 41, 37, 24, 8, 23, 26] has shown that some perceptual information can be recovered from various kinds of unencrypted syntax elements. This problem is due to the following fact: most multimedia coding standards are designed in such a way that many syntax elements can be decoded independently without decoding other syntax elements. To essentially overcome (or at least mitigate) this problem, the underlying multimedia coding standards have to be amended by introducing more dependence among syntax elements containing aural/visual information. But long-range dependence should be avoided, otherwise ROI encryption will be impossible.

Because many joint compression-encryption systems achieve encryption at the expense of compression efficiency, the size of ciphertext will not be the same as that of the plaintext. As a result, *size preservation* becomes another important concern in the design of multimedia encryption systems. In the ideal case, every syntax element should keep its original size after encryption. Size preservation is a stronger version of "no influence on compression efficiency". Typical applications of ideal size preservation include on-the-spot encryption[1], real-time mounting and dropping of encryption, simultaneous encryption at multiple points, and so on. In addition, if size preservation is fulfilled, bitrate re-control and re-packtization will not be necessary after encryption.

Yet another concern is about the capability to support the reuse of session key. Because of the nature of some multimedia encryption techniques, block ciphers cannot be used. In other words, only pseudo-random keystreams generated from session keys (i.e., stream ciphers) can be used. Because stream ciphers are not secure when

---

[1] On-the-spot encryption means that a file can be encrypted by simply writing the ciphertext back to the original place of the plaintext without creating a temporary copy.

any session key is reused, a key management system must be employed to determine a unique session key for each encrypted multimedia signal, which will make the whole system overcomplicated for some practical applications such as encryption of private pictures and videos on personal computers and hand-held devices.

Because there are many concerns of a multimedia encryption system, it is not sufficient to evaluate the overall performance with a single factor like security. We believe at least the following criteria (or requirements) should be considered:

1. security against various attacks;
2. ideal format compliance;
3. ideal size preservation;
4. reuse of session key (or flexibility – capability to work with both stream ciphers and block ciphers);
5. low encryption load (vs. naive encryption);
6. high energy efficiency (i.e., low energy consumption);
7. easy implementation in existing multimedia coders.

To our knowledge, the existing work has not considered the overall performance of multimedia encryption systems with respect to all of the above criteria, though some results have been reported on security, compression efficiency (weaker version of size preservation) and some implementation issues.

In the next section, we apply the above criteria to secret Huffman coding and secret arithmetic coding, trying to obtain a better understanding on the overall performance of the two main forms of secret entropy coding.

3   Secret Entropy Coding

First, we show some common issues about secret entropy coding, which are about the last three criteria.

Secret entropy coders realize encryption by keeping the statistical models and/or the behaviors of the entropy coders secret. Because an entropy coder has already been embedded in each multimedia coder, it is generally very easy to integrate encryption into the whole multimedia coding process. This is the main advantage of secret entropy coding, and also the main reason why it has attracted more attention than other multimedia encryption techniques.

There are two kinds of secret entropy coders: static and dynamic. A static secret entropy coder has a static statistical model and static coding behavior. It can be easily converted into a dynamic coder, by using a pseudo-random source (i.e., a stream cipher) to frequently update the statistical model and/or the coding behavior. In contrast, there does not seem to be a way to directly use block ciphers. At least no work has been reported on such a possibility. As a result, secret entropy coding both suffer from a common problem: the session key cannot be reused.

For static secret entropy coders, no encryption load is added to the base multimedia coding system, because there are no explicit encryption operations except for

the initial process of generating secret statistical models and/or coding behaviors. This is another major advantage of static secret entropy coding. For dynamic secret entropy coders, the condition is quite different. Since a stream cipher has to be used to frequently update the statistical model and/or the coding behavior, there is some additional encryption load. Because the update is carried out before compression, the encryption load consumed on the update will be more than that consumed on naive encryption as long as the update frequency f is larger than a critical value f_0. Assuming the computational complexity of the updating process is n times more complicated than that of the stream cipher involved, then the critical frequency will be $f_0 = r/n$, where $r = \text{Size(input)}/\text{Size(output)}$ is the compression ratio.

3.1   Secret Huffman Coding

Huffman coding is the most widely-adopted entropy coding algorithm in multimedia coding standards, such as JPEG [15], MPEG [14, 16, 17], H.264/AVC [18] and VC-1 [33]. Given a prior statistical model of the input, a Huffman tree is designed by assigning bit patterns of different sizes (i.e., different variable-length codewords – VLCs) to different nodes (i.e., different input symbols). For an input sequence of symbols, the output of a Huffman encoder is a sequence of VLCs. The Huffman tree is constructed in such a way that no VLC is the prefix of any other VLCs and thus a bitstream of VLCs can be decoded unambiguously. By assigning longer VLCs to input symbols with smaller occurrence probabilities, the effect of data compression is thus achieved. In multimedia coding standards, each Huffman tree involved are normally represented by a 1-D Huffman table, so the Huffman coding process can be performed via simple look-up-table operations. In newer multimedia coding standards like H.264/AVC, context-adaptive Huffman coding (formally named CAVLC – context-adaptive variable-length coding) is used. In this case, there are a number of candidate Huffman tables for each input symbol, and which one will be used is determined by previously-coded values (i.e., the context).

As we mentioned above, to design a secret entropy coder, one needs to keep the statistical models and/or the coding behavior secret. For secret Huffman coding, this means keeping one or more secret Huffman tables secret.

The most serious problem about secret Huffman coding is its incapability to maintain format compliance. There are three kinds of format incompliance. First, for any secret Huffman table, once there is some VLCs that are neither valid VLCs nor prefixes of valid VLCs in the original Huffman table, it will be impossible to decode these VLCs. Second, if some VLCs in a secret Huffman table are not included in the original table but are prefixes of some valid VLCs, the synchronization between encoder and decoder might be destroyed and as a result the decoding of the whole bitstream will fail at some point after any VLC is incorrectly decoded. Third, even when all the VLCs in a secret Huffman table are also valid VLCs in the original table, i.e., the secret table is obtained by permuting the original one, some semantic errors might still happen during the decoding process. For instance, the number

of DCT coefficients in a single block might exceed the theoretical upper bound (64 for an 8×8 block), since the number of zero coefficients is encoded in each VLC. Actually, the above results are common for all entropy coding algorithms in which VLCs are involved, such as Exp-Golomb coding used in H.264/AVC.

Though the integration of encryption into multimedia coding is also quite simple for secret Huffman coding, there is a potential problem about implementation. In some multimedia coding systems, the source code of the entropy coder is specially optimized for the original Huffman tables. So re-programming and re-compilation of the entropy coder might be necessary, though it is not heavy work in most cases.

In the following, we discuss the other two criteria – size preservation and security – for static and dynamic Huffman tables, respectively.

3.1.1 Static Huffman Tables

In most multimedia coding standards, especially those relatively old ones, static Huffman tables are used. Secret Huffman coding algorithm can be easily designed by replacing these static Huffman tables with secret (but also static) ones. Instead of designing secret Huffman tables from scratch, many researchers have suggested deriving them from the original ones defined in multimedia coding standards by performing some specific operations such as the following ones: 1) permuting VLCs in the original Huffman table [6, 21]; 2) tree mutation process – randomly swapping the bit patterns assigned to two branches at the same level of the Huffman tree [41]; 3) randomly flipping the last bits of some VLCs and adjusting other VLCs accordingly to ensure the validity (i.e., the prefix-rule) of the Huffman tree [21]. A stream cipher is generally used as a pseudo-random source to control the operations involved.

For static Huffman tables, there exists a conflict between size preservation and security. To achieve size preservation, the secret Huffman tables should have the same structure as the original ones, and thus each VLC has the same size as designed in [6, Algorithm I]. Unfortunately, as long as the size of each VLC does not change, the secret Huffman tables can be easily revealed in known/chosen plaintext attack, because the boundary between any two consecutive VLCs is obviously distinguishable. To solve this problem, some researchers proposed to relax the requirement on size preservation and allow some VLCs to have different sizes from the original ones [21, 41]. Unfortunately, such a relax of size preservation does not enhance security, because in plaintext attacks the size of each VLC can still be recognized by observing the context of the encrypted bitstream [25, 19]. For example, when $n \geq 2$ identical VLCs occur consecutively, it is quite easy to locate the repeated bit pattern.

Another security problem is about the number of "good" secret Huffman tables. Because of the nature of entropy coding, different input symbols have different occurrence probabilities, and thus different VLCs in a Huffman table have different levels of significance – shorter VLCs are more significant. However, the number of significant VLCs is often much smaller than the number of other less significant

VLCs. This means that the number of "good" secret Huffman tables is mainly determined by a small number of significant VLCs. This fact can explain the experimental results about MPEG-2 videos reported in [21]: a large number of candidates exist for each secret Huffman table, but only a quite small number of them are "good" for encryption. An implication of this fact is that the attacker does not need to break all VLCs. Instead, a partially-broken Huffman table might be enough to recover most information of a multimedia signal encrypted by a secret Huffman table.

To enlarge the relatively small key space of a single Huffman table, one can try to increase the number of secret Huffman tables, which is possible because normally more than one Huffman table is defined in each multimedia coding standard. For example, in MPEG-2 standard, 15 Huffman tables are defined and 5 of them are used in the secret Huffman coding algorithm proposed in [21]. Unfortunately, because these distinct Huffman tables are defined for coding different syntax elements, a divide-and-conquer (DAC) attack might be mounted to break all the secret Huffman tables separately. For instance, for the secret Huffman coding algorithm proposed in [21], the secret Huffman table B-14 can be separately broken by trying to decode a number of non-intra macroblocks, while other Huffman tables are still unknown. As a whole, static secret Huffman tables cannot offer a high level of security.

3.1.2 Dynamic Huffman Tables

To improve the security of static Huffman tables, dynamic Huffman tables can be used instead of static ones. There are three approaches to generate dynamic Huffman tables.

The first approach is so-called MHT (multiple Huffman tables) encryption [41]. That is, for each input symbol to be encoded and encrypted, a Huffman table is secretly selected from a number of (maybe public) candidate Huffman tables. In this case, a stream cipher should be used as a pseudo-random source to choose a specific Huffman table for each VLC. If the candidate Huffman tables are public and the session key is reused for distinct multimedia signals, it has been shown [19] that known/chosen plaintext attack can be launched to recover all the dynamic Huffman tables one by one. If the candidate Huffman tables are also kept secret, differential chosen-plaintext attack can be used to recognize VLCs in each secret Huffman table by observing the change of ciphertexts when only one VLC at a given position changes. So the security can be ensured only when the session key is not reused. Note that other problems of static Huffman tables cannot be overcome.

The second approach is to dynamically update the secret Huffman table after every n VLCs have been coded. This approach can be considered as a special case of the first one, where all the candidate Huffman tables are kept secret. Apparently, the analysis on the first approach remains the same for the second one.

The third approach is to use context-adaptive Huffman tables. In this case, dynamic Huffman tables are determined by previously-coded symbols (i.e., the context). Apparently, it is also a special case of the first approach.

3.2   Secret Arithmetic Coding

Arithmetic coding is a quite different entropy coding algorithm from Huffman coding, which represents each input symbol s as a subinterval $I(s)$ in the range $[0, 1)$ rather than a VLC in Huffman coding. The length of the subinterval is determined by the occurrence probability of the corresponding input symbol. The encoding process starts from the unit interval $I_0 = [0, 1)$, and a smaller subinterval I_i is obtained once a new symbol s_i is fed in, where the position and length of I_i is determined by $I(s_i)$, i.e., by the occurrence probability of s_i. After all n symbols have been processed, any fraction in the final subinterval I_n can be taken as the output of the encoder. Generally speaking, arithmetic coding is an optimal entropy coding and has a better compression performance than Huffman coding. In addition, it is more natural to introduce context adaptiveness in arithmetic coding than in Huffman coding, thanks to the separation of the statistical model and the coding process. In fact, most arithmetic coders available are context-adaptive.

Though arithmetic coding has not been adopted in multimedia coding standards until recently, the possibility of adding encryption into compression has attracted much attention since the very beginning of the development of arithmetic coding technique. After the appearance of the first proposal in 1988 [40], a lot of follow-up research have been reported [4,5,12,3,27,29,30,28,34,36,35,13,43,41,11,7,22,19]. Basically, there are three classes of secret arithmetic coding algorithms:

- *secret initial statistical model* – the initial statistical model is taken as the key, and the coder works as usual;
- *secret initialization process* – the initial statistical model is public, and a secret initialization process is carried out before encoding/decoding starts;
- *randomized coding* – a stream cipher is used to randomize the arithmetic coding process.

It has been known that the first two classes are not secure against chosen-plaintext attacks [4, 5, 27, 35]. A combination of the two classes of secret arithmetic coding algorithms is also insecure against chosen-plaintext attack, as shown in [34]. The third class of secret arithmetic coders are secure, but only when the session key is not reused [19, Sec. IV.A].

To improved the security of the first two classes of secret arithmetic coding algorithms, some amendments have been suggested [27, 35, 36]. One amendment is masking the output of the encoder with a secret pseudo-random keystream, which indeed can improve the security, but is just an additional level of security, not an essential enhancement on the security of the original secret arithmetic coding scheme. Another amendment is frequently resetting the statistical model, which improves the security at the expense of compression efficiency. Other amendments also have various weaknesses and limitations.

Besides the above security problems, there are also problems about size preservation and format compliance. As long as the statistical model is changed, a secret

arithmetic coder will be unable to achieve the same compression efficiency as the original one, so size preservation cannot be maintained. Similarly, because different statistical models lead to compressed data of different sizes, the synchronization between encoder and decoder will be destroyed, when an encrypted multimedia signal is decoded by an decoder without the knowledge of the key. That is, format compliance cannot be maintained, either.

According to the above analysis, to maintain size preservation and format compliance, the statistical model should remain untouched. In other words, only the behavior of the arithmetic coder can be secretly modified. Some secret arithmetic coding algorithms were proposed following this idea [13, 11], which change the position of the subinterval corresponding to each input symbol. Because the length of each subinterval of each input symbol remains, the compression efficiency will not be influenced in principle. As a result, it might be possible to maintain format compliance and size preservation simultaneously. Experiments in [11] have shown that such an ideal result can be achieved for JPEG2000 standard. But the condition is different for H.264/AVC standard, in which the arithmetic coder is often used to code VLCs and the termination of the coding process is related to the current coded VLC itself. In this case, any decoding error will definitely lead to the loss of synchronization between encoder and decoder, and thus format incompliance happens.

4 Conclusion

Table 1 shows a summary of the results obtained in the last section, from which we can see that neither secret Huffman coding nor secret arithmetic coding can fulfill all the criteria, but secret arithmetic coding can offer a better solution as a whole. The main advantage of secret arithmetic coding is its potential to maintain size preservation and format compliance for some multimedia coding standards. Considering the compression efficiency of arithmetic coding is also better, we propose to use the arithmetic coding in any new multimedia coding standards.

Recalling the reason why some criteria cannot be fulfilled by secret Huffman coding and secret arithmetic coding, we see possibilities to change the status by making

Table 1 Performance comparison of secret Huffman coding and secret arithmetic coding.

	Huffman coding	arithmetic coding
security	yes (session key not reused)/no	
format compliance	no	yes/no
size preservation	no	yes/no
reuse of key	no	
encryption load	very low/conditional	
implementation	easy	

some security-oriented amendments to existing multimedia coding standards, such as the following ones:

- *adding a compulsory error-tolerance mechanism*, which can help relax the requirement on format compliance and allow some kind of minor decoding errors (for example, the loss of synchronization when decoding a macroblock);
- *adding content-dependent IDs*, which can be employed by stream ciphers as initial vectors to generate different session keys for different plaintexts;
- *limiting the use of VLCs*, which will help alleviate the side effect of VLCs on format compliance;
- *introducing termination markers (like those used in JPEG2000 standard)*, which might help to maintain format compliance for secret arithmetic coders when VLCs are encoded.

In addition, to overcome the conflict between security and selective encryption, more dependence among syntax elements within a small area might be added or enhanced.

Note that some simple methods might be able to fulfill all performance criteria if some amendments are made to existing multimedia coding standard. For instance, if there is an error-tolerance mechanism and context-adaptive entropy coding (either Huffman coding or arithmetic coding) is used, one can simply selectively encrypt the n leading bits of the entropy encoder's output. As shown in [42], such a selective encryption might be able to achieve an effect of "virtual full encryption" when some conditions are satisfied. We can see that all other criteria can be fulfilled easily, too.

To sum up, considering the fact that an ideal solution to multimedia encryption cannot be easily found for most existing multimedia coding standards, we do believe that new security-oriented standards should be developed. In future we will focus our research on this direction and try to show the feasibility to work out such new standards based on existing ones.

References

1. Agi, I., Gong, L.: An empirical study of secure MPEG video transmission. In: Proc. ISOC Symposium on Network and Distributed Systems Security (SNDSS 1996), pp. 137–144 (1996)
2. Akyildiz, I.F., Melodia, T., Chowdhury, K.R.: Wireless multimedia sensor networks: Applications and testbeds. Proc. IEEE 96(10), 1588–1605 (2008)
3. Barbir, A.: A methodology for performing secure data compression. In: Proc. Twenty-Ninth Southeastern Symposium on System Theory (SSST 1997), pp. 266–270. IEEE (1997)

4. Bergen, H.A., Hogan, J.M.: Data security in a fixed-model arithmetic coding compression algorithm. Computers & Security 11(5), 445–461 (1992)
5. Bergen, H.A., Hogan, J.M.: A chosen plaintext attack on an adaptive arithmetic coding compression algorithm. Computers & Security 12(2), 157–167 (1993)
6. Bhargava, B., Shi, C., Wang, S.Y.: MPEG video encryption algorithms. Multimedia Tools and Applications 24(1), 57–79 (2004)
7. Bose, R., Pathak, S.: A novel compression and encryption scheme using variable model arithmetic coding and coupled chaotic system. IEEE Trans. Circuits and Systems–I 53(4), 848–857 (2006)
8. Engel, D., Stütz, T., Uhl, A.: Format-compliant JPEG2000 encryption in JPSEC: Security, applicability and the impact of compression parameters. EURASIP J. Information Security, art. no. 94, 565 (2007)
9. Furht, B., Kirovski, D. (eds.): Multimedia Security Handbook. CRC Press, LLC (2004)
10. Furht, B., Muharemagic, E., Socek, D. (eds.): Multimedia Encryption and Watermarking. Springer, Heidelberg (2005)
11. Grangetto, M., Magli, E., Olmo, G.: Multimedia selective encryption by means of randomized arithmetic coding. IEEE Trans. Multimedia 8(5), 905–917 (2006)
12. Irvine, S., Cleary, J., Rinsma-Melchert, I.: The subset sum problem and arithmetic coding. Research Report 95/7, Department of Computer Science, University of Waikato (1995)
13. Ishibashi, H., Tanaka, K.: Data encryption scheme with extended arithmetic coding. In: Proc. SPIE, Mathematics of Data/Image Coding, Compression, and Encryption IV, with Applications, vol. 4475, pp. 222–233 (2001)
14. ISO/IEC: Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 2: Video. ISO/IEC 11172-2, MPEG-1 (1993)
15. ISO/IEC: Information technology – Digital compression and coding of continuous-tone still images: Requirements and guidelines. ISO/IEC 10918-1, JPEG (1994)
16. ISO/IEC: Information technology – Generic coding of moving pictures and associated audio information: Video. ISO/IEC 13818-2 (MPEG2) (1996) (last revised in 2000)
17. ISO/IEC: Information technology – Coding of audio-visual objects – Part 2: Visual. ISO/IEC 14496-2 (MPEG-4), last revised in 2004 (2001)
18. Information, I.T.U.-T.: technology – Coding of audio-visual objects – Part 10: Advanced Video Coding. ISO/IEC 14496-10, (2004) (last revised in 2005); Also published as ITU-T Rec. H.264 in 2003 under the title Advanced video coding for generic audiovisual services (last revised in 2005)
19. Jakimoski, G., Subbalakshmi, K.P.: Cryptanalysis of some encryption schemes for multimedia. IEEE Trans. Multimedia 10(3), 330–338 (2008)
20. Johnson, M., Ishwar, P., Prabhakaran, V., Schonberg, D., Ramchandran, K.: On compressing encrypted data. IEEE Trans. Singal Processing 52(10), 2992–3006 (2004)
21. Kankanhalli, M.S., Guan, T.T.: Compressed-domain scrambler/descrambler for digital video. IEEE Trans. Consumer Eletronics 48(2), 356–365 (2002)
22. Kim, H., Wen, J., Villasenor, J.D.: Secure arithmetic coding. IEEE Trans. Singal Processing 55(5), 2263–2272 (2007)
23. Li, S., Ahmad, J.J., Saupe, D., Kuo, C.C.J.: An improved DC recovery method from AC coefficients of DCT-transformed images. In: Proceedings of 17th IEEE International Conference on Image Processing (ICIP 2010), pp. 2085–2088 (2010), http://www.hooklee.com/default.asp?t=AC2DC

24. Li, S., Chen, G., Cheung, A., Bhargava, B., Lo, K.T.: On the design of perceptual MPEG-video encryption algorithms. IEEE Trans. Circuits and Systems for Video Technology 17(2), 214–223 (2007)
25. Li, S., Chen, G., Cheung, A., Lo, K.T., Kankanhalli, M.: On the Security of an MPEG-Video Encryption Scheme Based on Secret Huffman Tables. In: Wada, T., Huang, F., Lin, S. (eds.) PSIVT 2009. LNCS, vol. 5414, pp. 898–909. Springer, Heidelberg (2009)
26. Li, S., Karrenbauer, A., Saupe, D., Kuo, C.C.J.: Recovering missing coefficients in DCT-transformed images. In: Proceedings of 18th IEEE International Conference on Image Processing, ICIP 2011. IEEE (2011),
    `http://www.hooklee.com/default.asp?t=ICIP2011`
27. Lim, J., Boyd, C., Dawson, E.: Cryptanalysis of Adaptive Arithmetic Coding Encryption Schemes. In: Mu, Y., Pieprzyk, J.P., Varadharajan, V. (eds.) ACISP 1997. LNCS, vol. 1270, pp. 216–227. Springer, Heidelberg (1997)
28. Liu, X., Farrell, P., Boyd, C.: A Unified Code. In: Walker, M. (ed.) Cryptography and Coding 1999. LNCS, vol. 1746, pp. 84–93. Springer, Heidelberg (1999)
29. Liu, X., Farrell, P.G., Boyd, C.: Resisting the Bergen-Hogan Attack on Adaptive Arithmetic Coding. In: Darnell, M.J. (ed.) Cryptography and Coding 1997. LNCS, vol. 1355, pp. 199–208. Springer, Heidelberg (1997)
30. Liu, X., Farrell, P.G., Boyd, C.A.: Arithmetic coding and data integrity. In: Proc. Workshop on Coding and Cryptography (WCC 1999), pp. 291–299 (1999)
31. Lookabaugh, T.D., Sicker, D.C., Keaton, D.M., Guo, W.Y., Vedula, I.: Security analysis of selectively encrypted MPEG-2 streams. In: Proc. SPIE, Multimedia Systems and Applications VI, vol. 5241, pp. 10–21 (2003)
32. Qiao, L., Nahrsted, K.: Comparison of MPEG encryption algorithms. Computers & Graphics 22(4), 437–448 (1998)
33. SMPTE (Society of Motion Picture and Television Engineers): Standard for television – VC-1 compressed video bitstream format and decoding process. SMPTE 421M (2006)
34. Uehara, T., Safavi-Naini, R.: Attack on Liu/Farrell/Boyd arithmetic coding encryption scheme. In: Proc. IFIP TC6/TC11 Joint Working Conference on Secure Information Networks: Communications and Multimedia Security (CMS 1999), pp. 273–290 (1999)
35. Uehara, T., Safavi-Naini, R.: Attacking and mending arithmetic coding encryption schemes. In: Proc. Australasian Computer Science Conference, pp. 408–419 (1999)
36. Uehara, T., Safavi-Naini, R.: Designing secure arithmetic coding encryption schemes. In: Proc. 22nd Symposium on Information Theory and its Applications (SITA 1999), vol. 2, pp. 773–776 (1999)
37. Uehara, T., Safavi-Naini, R., Ogunbona, P.: Recovering dc coefficients in block-based dct. IEEE Trans. Image Processing 15(11), 3592–3596 (2006)
38. Uhl, A., Pommer, A.: Image and Video Encryption: From Digital Rights Management to Secured Personal Communication. Springer, Heidelberg (2005)
39. Wen, J., Severa, M., Zeng, W., Luttrell, M.H., Jin, W.: A format-compliant configurable encryption framework for access control of video. IEEE Trans. Circuits and Systems for Video Technology 12(6), 545–557 (2002)
40. Witten, I.H., Cleary, J.G.: On the privacy afforded by adaptive text compression. Computers & Security 7(4), 397–408 (1988)
41. Wu, C.P., Kuo, C.C.J.: Design of integrated multimedia compression and encryption systems. IEEE Transactions on Multimedia 7(5), 828–839 (2005)

42. Wu, X., Moo, P.W.: Joint image/video compression and encryption via high-order conditional entropy coding of wavelet coefficients. In: Proc. IEEE Conference on Multimedia Computing and Systems (CMS 1999), pp. 908–912 (1999)
43. Xie, D., Kuo, C.C.J.: Efficient multimedia data encryption based on flexible QM coder. In: Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents VI, vol. 5306, pp. 696–704 (2004)
44. Zeng, W., Yu, H., Lin, C.Y. (eds.): Multimedia Security Technologies for Digital Rights Management. Academic Press (2006)

Author Index