CONFERENCE OF PHD STUDENTS IN COMPUTER SCIENCE

Volume of extended abstracts

CS^2

Organized by the Institute of Informatics of the University of Szeged

July 20-23, 2000
Szeged, Hungary

Scientific Committee:

Mátyás Arató (KLTE)
Miklós Bartha (SZTE)
András Benczúr (ELTE)
Tibor Csendes (SZTE)
János Csirik (SZTE)
János Demetrovics (SZTAKI)
Sarolta Dibuz (Ericsson)
József Dombi (SZTE)
Zoltán Ésik (SZTE)
Ferenc Friedler (VE)
Zoltán Fülöp (SZTE)
Ferenc Gécseg (chair, SZTE)
Balázs Imreh (SZTE)
János Kormos (KLTE)
László Kozma (ELTE)
Attila Kuba (SZTE)
Eörs Máté (SZTE)
Gyula Pap (KLTE)
András Recski (BME)
Endre Selényi (BME)
Katalin Tarnay (NOKIA)
György Turán (SZTE)
László Varga (ELTE)

Organizing Committee:

Tibor Csendes, Éva Gombás, Mihály Markót, László Martonossy, Lajos Schrettner, Mariann Sebő.

Address of the Organizing Committee

c/o. Tibor Csendes
University of Szeged, Institute of Informatics
H-6701 Szeged, P.O. Box 652, Hungary
Phone: +36 62 544 305, Fax: +36 62 420 292
E-mail: cscs@inf.u-szeged.hu
URL: http://www.inf.u-szeged.hu/∼cscs/

Main sponsors

Nokia Hungary Ltd., SIEMENS — SYSDATA

Sponsors

Budapest University of Technology and Economics, Faculty of Electrical Engineering and Informatics; Cygron Research and Development Ltd.; Eötvös University, Institute of Informatics; John von Neumann Computer Society; NOVADAT Bt.; OM KFHAT; Szeged City Major's Office, University of Debrecen, Institute of Mathematics and Informatics; University of Szeged, Institute of Informatics; Zenon Ltd.

Preface

This conference is the second in a series. The organizers have tried to get together those PhD students who work on any fields of computer science and its applications to help them possibly in writing their first abstract and paper, and may be to give their first scientific talk. As far as we know, this is one of the few such conferences. The aims of the scientific meeting were determined on the council meeting of the Hungarian PhD Schools in Informatics: it should

- provide a forum for PhD students in computer science to discuss their ideas and research results,
- give a possibility to have constructive criticism before they present the results in professional conferences,
- promote the publication of their results in the form of fully refereed journal articles, and finally
- promote hopefully fruitful research collaboration between the participants.

The best talks will be awarded with the help of our sponsors. The papers emerging from the presented talks will be forwarded to the journals of Acta Cybernetica (Szeged), and Periodica Polytechnica (Budapest); and the mathematics oriented papers to Publicationes Mathematicae (Debrecen). The deadline for the submission of the papers is the end of August 2000. The manuscripts will be forwarded to the proper journals. To get acquainted with the style of the journals please study earlier issues of them. One sample paper is available at `http://www.inf.u-szeged.hu/~cscs/csallner.tex`.

Although we did not advertise it on the web, a high number of good quality abstracts have been submitted. If you encounter any problems during the meeting, please do not hesitate to contact one of the Organizing Committee members. The organizers hope that the conference will be a valuable contribution to the research of the participants, and wish a pleasant stay in Szeged.

Szeged, July 2000

Tibor Csendes

Contents

Preliminary Program

Overview

Thursday, July 20

- 10:00 - 14:00 Registration
- 14:00 - 14:15 Opening
- 14:15 - 15:00 Plenary talk
- 15:00 - 15:15 Break
- 15:15 - 16:45 Talks in 2 streams (3x30 minutes)
- 16:45 - 17:00 Break
- 17:00 - 18:00 Talks in 2 streams (2x30 minutes)
- 18:15 - 19:30 Reception at the Town Hall

Friday, July 21

- 08:30 - 10:00 Talks in 2 streams (3x30 minutes)
- 10:00 - 10:15 Break
- 10:15 - 11:00 Plenary talk
- 11:00 - 11:15 Break
- 11:15 - 12:45 Talks in 2 streams (3x30 minutes)
- 12:45 - 14:00 Lunch
- 14:00 - 15:30 Talks in 2 streams (3x30 minutes)
- 15:30 - 15:45 Break
- 15:45 - 17:15 Talks in 2 streams (3x30 minutes)
- 18:00 - 19:30 Supper

Saturday, July 22

- 08:30 - 09:30 Talks in 2 streams (2x30 minutes)
- 09:30 - 09:45 Break
- 09:45 - 10:30 Plenary talk
- 10:30 - 10:45 Break
- 10:45 - 12:45 Talks in 2 streams (4x30 minutes)
- 12:45 - 14:00 Lunch
- 14:00 - 20:00 Excursion and supper

Sunday, July 23

- 08:30 - 10:00 Talks in 2 streams (3x30 minutes)
- 10:00 - 10:15 Break
- 10:15 - 11:00 Plenary talk
- 11:00 - 11:15 Break
- 11:15 - 12:45 Talks in 2 streams (3x30 minutes)
- 12:45 - 14:00 Lunch
- 14:00 - 15:30 Talks in 2 streams (3x30 minutes)
- 15:30 - 16:00 Break
- 16:00 - 16:30 Closing session, announcing the Best Talk Awards
- 18:00 - 20:30 Supper

Monday, July 24

- 8:30 Departure

Detailed program

Thursday, July 20

10:00	Registration	
14:00	Opening session	
14:15	**Plenary talk** János Csirik *Average Case Analysis of Algorithms*	
15:00	Break	
Sections	**Networks**	**Discrete mathematics**
15:15	Mihály Bohus *Analysing Mobility Options of Data Communication*	Emese Balogh *Generating and Reconstructing hv-convex 2-dimensional Discrete Sets*
15:45	Gábor Gesztesi *Optimizing Object Location in Mobile Distributed Object Systems*	László Szûcs, G. Pécsi *Parallel verification and enumeration of tournaments*
16:15	Zalán Heszberger, József Bíró, János Zátonyi *Efficient CAC Algorithms Based on the Tail Distribution of Aggregate Traffic*	Tamás Herendi *Uniform distribution of linear recurrence sequences in residue class systems*
16:45	Break	
Sections	**Code generating, software engineering**	**Image processing**
17:00	Péter Felvégi *Low complexity parametrized codes for LZ77 compression*	Gábor Csornai, I. Fekete, I. László *Segmentation methods based on spatial characteristics of multispectral images*
17:30	Ákos Frohner, Zoltán Porkoláb, László Varga *Code Generation from UML models*	Marianna Dudásné Nagy, E. Máté, B. Kári *Generation of Normal Control Data for the Evaluation of Myocardial Perfusion Studies*
18:15	Reception at the Town Hall	

Friday, July 21

Sections	**Formal language**	**Artificial intelligence**
08:30	Szilvia Zvada, T. Gyimóthy *Using Decision Trees to Infer Semantic Functions of Attribute Grammars*	István Harmati, Bálint Kiss *Motion planning algorithms for stratified kinematic systems with application to the hexapod robot*
09:00	Barnabás Gõgös *Automatic Test Purpose Generation based on Formal Grammar Representation of Communication Protocols*	Gusztáv Jánvári *Strategy Selection in the Resolution Theory*
09:30	István Katsányi *Sets of numbers in different number systems and the Chomsky hierarchy*	Márk Jelasity, Boglárka Tóth, Tamás Vinkó *Measuring hardness of problems in evolutionary computations using Markov-processes*
10:00	Break	
10:15	**Plenary talk** György Maróti *Automata Theory with Maple*	
11:00	Break	
Sections	**Information system**	**Image processing**
11:15	Hung T. Tran, Tien V. Do *Generalised Subspace based Method for Steady State Analysis of QBD-M Processes*	Kálmán Palágyi *A 3D Directional Shrinking Algorithm*
11:45	Zsigmond Pap *Static Specification Completeness Checking of UML State Machines*	Tran Minh Son, Gy. Marosi, A. Gschwindt *SIRDS based autostereoscopic pictures*
12:15	Lóránt Farkas and Lajos Nagy *Wireless Application Protocol Performance Testing*	Attila Tanács *Investigation of Point-Based Image Registration Methods Assuming Rigid-Body and Linear Motions*
12:45	Lunch	

(see next page for the rest of the **Friday** program)

Friday, July 21 (continued)

Sections	**Optimization**	**Automata**
14:00	János Balogh, T. Csendes, R. P. Stateva *A New Global Optimization Technique for Chemical Phase Equilibrium Problems*	Miklós Bartha, Miklós Krész *On deterministic soliton automata*
14:30	András Erik Csallner, D. Ratz, R. Klatte *The Effects of the Boxing Method for Interval Subdivision Algorithm*	Jelena Kovacevic *Quasi-Orders on Automata*
15:00	M. Cs. Markót, P. G. Szabó, B. Tóth, T. Vinkó *Verification solutions of packing circle problems*	Zarko Popovic *Congruences on Finite Automata*
15:30	Break	
Sections	**Artificial intelligence**	**Databases, networks**
15:45	Radmila Jovanovic, Ognjen Radovic *Predicting Exchange Rate by Neural Networks*	László Keresztfalvi *Mobile technologies and future healthcare*
16:15	György Koch, József Dombi *New Clustering Procedures in Respect of Practical Realization*	István Szépkúti *On the Scalability of Multidimensional Databases*
16:45	András Kocsor, László Tóth, András Kuba *Phoneme Classification Using Kernel Principal Component Analysis*	Károly Farkas, Zoltán Balogh, Henrik Villför *IP Traffic Engineering over OMP technique*
18:00	Supper	

Saturday, July 22

Sections	**Discrete algorithms**	**Functional programming**
08:30	László Aszalós *A Method to Solve the Puzzles of Knights and Knaves*	
09:00	Sándor Vályi *About the axiomatization of first- and second-order spatio-temporal logics*	Zoltán Horváth, V. Zsók, P. Serrarens, R. Plasmeijer *Parallel Functional Reactive Skeletons in Concurrent Clean*
09:30	Break	
09:45	**Plenary talk** Tamás Rapcsák *Minimization on Stiefel Manifolds*	
10:30	Break	
Sections	**Numerical algorithms**	**Logic programming, picture processing**
10:45	Ester Martin Garzón, I. García *Parallel Implementation of the Large Sparse and Symmetric Eigenproblem*	László Harmath, Gy. Szilágyi, T. Gyimóthy *Debug Slicing of Logic Programs*
11:15	Eva M. Ortigosa, L.F. Romero, J.I. Ramos *Parallel simulation of spiral waves in reacting and diffusing media*	Zsolt Németh *Definition of a Parallel Execution Model with Abstract State Machines*
11:45	J.M. Gonzalez-Linares, P.M. Ortigosa, N. Guil and I. García *Parallelization of an algorithm for non rigid objects detection with stochastic global optimization*	Csongor Halmai, Balázs Erdõhelyi, Krisztián Ollé, Attila Kuba *New Results in 3D Surface Reduction*
12:15		László G. Nyúl and Jayaram K. Udupa *Standardizing the MR Image Intensity and Its Applications*
12:45	Lunch	
14:00	Excursion and supper	

Sunday, July 23

Sections	**Artificial intelligence, fuzzy**	**Software engineering**
08:30	Levente Sára *Dynamic Simulations*	András Dezsõ, Vince Bárány *State Space Transformation - A Specification Language and a Program Generator*
09:00	Ákos Zsiros *Learning Decision Trees in Continuous Space*	László Gulyás *The History of Software Localization*
09:30	Attila Gyenesei *A Fuzzy Approach for Mining Interesting Quantitative Association Rules*	Zoltán Hornák, Endre Selényi *Effective Virus Scanning Algorithms*
10:00	Break	
10:15	**Plenary talk** Katalin Tarnay *Formal Methods in Protocol Engineering*	
11:00	Break	
Sections	**Java, conformance testing**	**Optimization**
11:15	Zoran Putnik, M. Grabovac *Extension of Java with Turtle Graphics*	L.G. Casado, J.A. Martínez, I. García, T. Csendes *Adaptive Multisection in Derivative Free Interval Methods for Global Optimization*
11:45	Attila Ulbert and Markus Hof *Time Independent Invocation in Java CMS*	Csanád Imreh *On a two class on-line classification problem*
12:15	Antal Fazakas *Auto Case Test Selection from WAP Test Suites*	Péter Gábor Szabó, B. Tóth, T. Vinkó *Optimal disk packings in the square*
12:45	Lunch	

(see next page for the rest of the **Sunday** program)

Sunday, July 23 (continued)

Sections	**Protocols**	**Formal grammar, optimization**
14:00	József Hosszú *Discrete Simulation of Distributed Systems - Performance Evaluation of a Notification Channel Federation*	Csaba V. Rotter *Formal Grammars in Conformance Testing*
14:30	Endre Horváth, Axel Manthey *GPRS Function Test using TTCN*	J. A. Martínez, L. G. Casado, I. García, Ya. D. Sergeyev *Search space reduction criterion based on derivatives in Global Optimization algorithms*
15:00	Krisztián Kiss *SIP or H.323: Which Call Control Protocol is More Suitable for the 3G UMTS All-IP Network?*	G. Somlai, L. Sogor, L. Martonossy, M. Fidrich, G. Dikan, T. Tarjanyi, P. Hendlein and M. Bohus *Test of inter-working and translation mechanisms between IPv4 and IPv6*
15:30	Break	
16:00	Closing session, announcing the Best Talk Awards	
18:00	Supper	

Monday, July 24

- Departure

A Method to Solve the Puzzles of Knights and Knaves [1]

László Aszalós

In his book "What is the title of this book?" Raymond M. Smullyan used a lot of puzzles to illustrate the background of the Gödel incompleteness theorem. These puzzles became popular and nowadays are being published in amusement magazines too. In each section of the book different conditions are met. In the best known type of puzzles we have only two types of people, knights and knaves. Knights always tell the truth and knaves always lie.

Since in each puzzle there are only finitely many characters, we shall work with a finite number of characters, too. For the sake of simplicity, we denote them by a, b, c, ... To formalize the puzzles we need to extend the concept of a formula. In addition to usual connectives we shall use the symbols T_x, F_x, S_xA and C_xA. Here x denotes somebody among a, b, c, ... and A denotes a formula that may contain S_x or C_x; hence, we allow the nesting of these modal operators with no restriction, but the formula must be finite, of course. The reading of T_x, F_x, S_xA and C_xA are: x is a knight, x is a knave, x said that A and x can say that A, respectively. To construct a model we need to know about each person whether is he (she) a knight or a knave, and what he (she) said. If x is a knight, then T_x will be true and F_x will be false, and if x is a knave, then the opposite will be the case. If x said that A, then S_xA will be true; otherwise, it will be false.

In Smullyan book the characters are polite, they answer all the questions. If we take the general case, we can no more expect this. If somebody said n sentences, he does not need to say a new one, even if he could. A knight could say all the true formulae, which are infinitely many, but he always said a finite number of them. For this reason we shall work with *could say* or *can say*, with the modal operator C_x. The formula C_xA will be true if A is true and x is a knight or A is false and x is a knave. In all other cases C_xA will be false.

The are several methods for solving the puzzles. Smullyan's method is based on the type of characters. However, later, when we will have a lot of cases, this method will be uncomfortable and inefficient. David Gries rewrote the formulas using the logical law $S_xA \to (A \equiv T_x)$ and afterwards he was able to work with formulae of propositional logic. This rewriting was based on the fact that $Tx \equiv \neg Fx$. In treating the subsequent sections of the book we shall loose this nice property. Larry Wos rewrote the puzzles as first order formulae and used the theorem prover Otter to solve them. Using this method we can solve complicated puzzles, too, but there is no complete algorithm for solving all the first order formulae, which is due to the incompleteness theorem.

For this reasons I constructed my own method to solve this and other puzzles of the book. This method is based on the well-known method of analytic tableaux. I needed only to add new rules for the new type of formulae. I show that my method can be extended to the puzzles formulated with a third type of people - normal people - who can say anything. At the end of this article I describe the heart of my prover that can solve all these puzzles.

[1] Research was partly supported by the grant OTKA no. 354-19341

Generating and Reconstructing hv-convex 2-dimensional Discrete Sets

Emese Balogh

The reconstruction of 2-dimensional discrete sets from their projections has been studied in several classes. In certain classes the reconstruction is NP-hard, therefore the most frequently studied classes are those, where the reconstruction can be performed in polynomial time, like the hv-convex polyominoes and hv-convex 8-connected sets.

The reconstruction problem is to determine a 2-dimensional discrete set F from its two orthogonal projections H and V or report failure if such F does not exist. We have implemented and slightly modified a well-known algorithm in the literature for the reconstruction in the class of hv-convex polyominoes and in the class of hv-convex 8-connected sets. The algorithm approaches iteratively the solution by a nondecreasing and a nonincreasing sequence of discrete sets. The elements of the first sequence are called core sets and are subsets of F and the elements of the second sequence are called envelope sets and supposed to contain F.

In order to test the algorithm, we need to generate such sets at random. W. Hochstätter, M. Loebl, and C. Moll give a recursion formula for the number of hv-convex polyominoes, derive a bijection between an interval of natural numbers and the polyominoes of given perimeter which provides the possibility to generate them at random. In this paper we extend the recursion formula for the greater class of the hv-convex 8-connected sets, and using a similar bijection we are able to generate such sets at random.

A New Global Optimization Technique for Chemical Phase Equilibrium Problems

J. Balogh, T. Csendes, R.P. Stateva

Here we present a new approach for the chemical phase equilibrium stability problem: a new model with a modified tangent phase function, and a clustering stochastic optimization method using two derivative free local search procedures:

`ftp://ftp.jate.u-szeged.hu/pub/math/optimization/index.html`

A numerical study has been completed on typical lower dimensional problems of the literature. The comparison shows that the new technique is efficient and reliable. According to our numerical test results, the efficiency indicators, such as the number of function evaluations and CPU time required, are better than those of a comparable technique. The new method seems also to be very promising on larger dimensional problems.

Soliton automata and graph matchings

M. Bartha and M. Kresz

A soliton automaton is the mathematical model of so called soliton valves in certain carbohydrate molecules, and has the potential to serve as a future molecular switching device. The underlying object of a soliton automaton is a soliton graph, which is an undirected graph G having a perfect internal matching, i.e., a matching that covers all the vertices of G with degree at least two. Such vertices are called internal, whereas vertices with degree one are called external in G. The states of the automaton are all the perfect internal matchings of G, and transitions are defined by making alternating walks connecting two external vertices in G.

In this paper we will characterize soliton automata through its underlying graph object. First we will show that a decomposition of any soliton graph can be given according to its global internal structure. The corresponding soliton automaton, too, can be decomposed into component automata determined by the above structural results. Then a full characterization is obtained for automata based on the internal components, therefore the main contribution of the above results is to reduce the general problem to a simpler one.

Context-free grammars are also frequently used for the description of certain graph properties. We will give the grammars defining exactly the soliton graphs. The above structural decomposition will play important role in this result.

The final topic of this paper is the characterization of deterministic soliton graphs. To this end a reduction procedure is given for soliton graphs, by which every graph has a minimal representation. It is then proved that a 1-extendable and minimal soliton graph with external vertices has an alternating cycle with respect to some of its states iff the graph contains an even-length cycle. This result leads to a characterization of deterministic soliton graphs, saying that every internal elementary component of such a graph is a single mandatory edge, and the minimal representation of each external component is a graph not containing even-length cycles. Using this characterization, the deterministic property becomes straightforward to check for any soliton automaton given by its underlying graph.

Analysing mobility options of data communication

Mihály Bohus

Mobility plays very important role in the data communication. The trends are estimated extremely high number of mobile hosts in the next three years. Despite the barriers the demands are to apply the same applications as the fix hosts are used. It means we can find the mobility options in every layer of architectures (Internet/IPv4-6, SS7, ISDN/ATM).

In first part we prepare a comparative study about mobility protocols and their options (most critical features) of existing mobile data communication systems (paging, RDS, SDS, MDN, RDN, PLMN and DLMN). These systems are constructed by using the stacks of protocols, in the hierarchy minimum four layers (physical, data link, network and application) are involved in mobility.

Second part deals with modelling of hand-over, secure tunelling, roaming and cellular mobile data service used the Hierarchical Message Sequence Chart (HMSC) protocol description tool. It is suitable both for preparation of short protocol overview and both detailed communication behaviour description. This activity is valuable for studying and understanding these processes precisely and planning the protocol test suite.

References

[1] Dárdai, Á., Mobil Távközlés, Nap Kiadó , 1999.

[2] Saunders, S., Heywood, P., Dornan, A., Bruno, L., Allen, L., Itt a mobil IP, 1999 december.

[3] Huitema, C., IPv6, Prentice Hall, 1998.

[4] Bohus, M., et al.,Comparison of Ipv4 and Ipv6, Study report, 1999.

[5] Short, J., et al., Mobile wireless network system simulation, Wireless Networks 1 (451-467), 1995.

[6] Raychaudhuri, D., Wireless ATM: An enabling technology for multimedia personal communication, Wireless Networks 2 (163-171), 1996.

[7] HMSC, SDL-forum, http, 1999.

Adaptive Multisection in Derivative Free Interval Methods for Global Optimization[2]

L.G. Casado, J.A. Martínez, I. García, and T. Csendes

A new adaptive multisection technique in interval methods for global optimization is investigated, and numerical tests demonstrate that the efficiency of the underlying global optimization method can be improved substantially. The heuristic rule is based on experiences that suggest the subdivision of the current subinterval into a larger number of pieces only if it is located in the neighbourhood of a minimizer point. An estimator of the proximity of a subinterval to the region of attraction to a minimizer point is utilized.

This paper investigates adaptive multisection variants of a branch-and-bound algorithm [3, 4] for solving the box constrained global optimization problem [5]:

$$\min_{x \in X} f(x), \tag{1}$$

where the n-dimensional interval $X \subseteq \mathbb{R}^n$ is the search region, and $f(x) : X \subset \mathbb{R}^n \to \mathbb{R}$ is the objective function. The global minimum value of f is denoted by f^*, and the set of global minimizer points of f on X by X^*. That is,

$$f^* = \min_{x \in X} f(x) \qquad \text{and} \qquad X^* = \{x^* \mid f(x^*) = f^*\}.$$

The paper discussed a new adaptive decision rule for multisection algorithms in interval global optimization. According to the theoretical results, the suggested algorithmic variant can save in the best case up to one half of the inclusion function evaluations, while in the worst case, it may require an arbitrary large times more function evaluations than standard bisection. The underlying $p\tilde{f}$ [1, 2] algorithm parameter is not necessarily reliable: in extreme cases the sequence of actual subintervals can be distracted for an arbitrary long time, and also the dynamic multisection rule can suggest inefficient subdivision types.

An extensive numerical study was made that was based on the exact, a priori known global minimum value, on an approximation of it, and on the easily available upper bound $\tilde{f}$ of it. These new adaptive multisection methods improved the number of interval (by about one third) and real function evaluations (by around two third), and therefore also the execution time substantially (by up to a half). The memory complexity proved also to be smaller when larger accuracy was requested (by up to one half). The new adaptive multisection strategies seem to be indispensable, and can improve both the computational and the memory complexity substantially. The introduced adaptive multisection technique is appropriate for both differentiable and non-differentiable functions. Nevertheless, for differentiable functions our proposal could be improved further by applying it together with sophisticated accelerating devices and other algorithmic changes.

References

[1] Casado, L. G., García, I.: New load balancing criterion for parallel interval global optimization algorithm. Proceedings of 16th IASTED International Conference. Garmish-Partenkircen, Germany, 321 – 323 (1998).

[2] Casado, L. G., García, I., Csendes, T.: A Heuristic Rejection Criterion in Interval Global Optimization Algorithms. Submitted for publication. Available at `http://www.inf.u-szeged.hu/~csendes/publ.html`.

[3] Kearfott, R. B.: Rigorous global search: continuous problems. Dordrecht: Kluwer 1996.

[4] Ratschek, H., Rokne, J.: New Computer Methods for Global Optimization. Chichester: Ellis Horwood 1988.

[5] Törn, A., Žilinskas, A.: Global Optimization. Berlin: Springer 1987.

[2] This work was supported by the Ministry of Education of Spain (CICYT TIC99-0361, and by the Grants FKFP 0739/97, OTKA T 016413 and T 017241.

The Effects of the Boxing Method for Interval Subdivision Algorithms[3]

A.E. Csallner, D. Ratz and R. Klatte

The bound-constrained global optimization problem which is the scope of this work can be defined in general as $\min_{x \in X} f(x)$ where X is a — possibly multidimensional — interval. Note, that a great class of real-life problems are covered by the bound-constrained global optimization problem.

The original problem can be solved with verified accuracy with the aid of interval subdivision methods (see, e.g., [1]). These algorithms are based on the well-known branch-and-bound principle. Thus, a search tree is built where the whole search region — the interval X — is the root and the particular levels consist of subintervals which are partitions of their parents in the tree. Those branches that cannot be pruned have to be stored for later treatment. Thus, it is desirable to exclude as many subintervals from further investigation as possible.

The methods bounding the search tree are called accelerating devices. One of the most effective of these is the interval Newton step. It can, however, only be applied if f is once continuously differentiable. On the other hand, its time complexity is relatively high with regard to other accelerating devices like the cut-off test or the monotonicity test. Therefor it should only be deployed if there are no other possibilities to effectively bound the search tree. A new method ([2]) called the *boxing method* makes the decision not to carry out the interval Newton step if a good bound for the cut-off test has been found in the midpoint of an interval Y, but to split Y into several subboxes leaving a small interval containing the midpoint of Y.

The oral presentation discusses the effects of the boxing method and gives some possibilities to apply this idea to various interval subdivision algorithms.

References

[1] Ratschek H. and Rokne J. (1993), *Interval Methods*, In: Handbook of Global Optimization, Horst R. and Pardalos P.M. (eds.), Kluwer, Dordrecht, pp. 751–828.

[2] Wiethoff A. (1997), *Verifizierte globale Optimierung auf Parallelrechnern*, PhD Thesis, Universität Karlsruhe.

[3]This work has been supported by the Grants OTKA F025743 and D07/97.

Segmentation methods based on spatial characteristics of multispectral images

Gábor Csornai, István Fekete and István László

The importance of remotely sensed images increases parallel to the technology development. The quality of digital images is improving while their unit cost decreases. Accordingly, processing and information extraction methods must also evolve. It is clear that the vast data set of images can also generate a lot of practical problems. These are the unacceptable processing time, and those areas where human interaction can't be eliminated. This decreases efficiency and may cause some unpredictable error.

Our paper concentrates on agricultural applications of remotely sensed images. These include area estimation and early yield forecast. Remotely sensed images convey information about a certain area of the Earth's surface. The image data can be considered as a matrix, in which each element (pixel) consists of a so-called radiation intensity vector. This is quantitative data from an elementary unit of the surface. The radiation is measured by the sensors on board of the satellites.

As a starting procedure we can process pixels individually, ignoring their spatial properties and relationships. However, the adjacent matrix elements belong to adjacent spots, therefore these pixels are similar. Consequently, it's often useful to treat neighbouring similar pixels as a unit, called segment. A possible and sometimes better approach is to start to find segments, the homogeneous, spatially contiguous set of pixels. The segments are used as the basic elements in the course of further processing.

The advantage of the segmentation is particularly obvious if we consider the next processing phase, the clusterizing. Its purpose is to divide the similar pixels into a given (or restricted) number of data vector classes. In the straightforward (per pixel) approach clusterization doesn't utilize spatial relationship between pixels, it treats and evaluates the single intensity vectors in an isolated way. Therefore this approach sometimes does not recognize relations that are obvious to the human interpreter. The segmentation makes the classification easier, helps visual interpretation, and—as the number of segments is much less than that of pixels—decreases processing time.

Added auxiliary knowledge of some administrative boundary information may also highly improve the classification accuracy. The first, most common step in this way is to use the parcel boundaries database. Obviously, in the majority of cases the pixels' spatial structure is strongly correlated to this administrative subdivision. Therefore it seems appropriate to classify the pixel sets determined by parcels. This provides a good basis for the classification. In the process of segment delineation we can restrict the procedure within the current land parcel. This not only helps in adequate ground segments, but also enormously increases speed, as we have to test much less cases. In part of the cases administrative subdivision provides more usable information to the classification than segmentation does.

A sound theoretical background and the segmentation algorithm is based on a former technical report and article. The algorithm can be divided into two levels, both based on statistical hypothesis tests. At the first level the image is partitioned into small rectangular areas, called cells. At the second level the algorithm merges similar cells into segments. Its results are improved and adapted to the present application and conditions. The current processing technology allows us to study a much wider range of parametrization and to run the algorithm on larger data sets.

Our paper focuses on this segmentation process, presenting a considerably fast algorithm with some flexibility in parametrization. We also aim to evaluate the advantages and possibilities provided by these methods. The further goal is to develop a complete classification system that processes segmented images. This would yield a substantial improvement of the presently used methods. Towards this goal, a representative artificial data set will be used to validate the efficiency of the "per-field" approach. Our expectation seems to be realistic, because the intended method incorporates spatial information into the classification.

The importance of the utilization of advanced, proper and adequate methods in the applications of satellite images is particularly high in operational programs. Based on a vast 300 man/year R+D investment by FÖMI Remote Sensing Centre in Hungary, a national Crop Monitoring and Production Forecast System has been operational for four years now. Improvement in adequacy, speed and reliability is a continuous requirement to the program. A special problem, a possible direction of development is addressed by this study.

References

[1] Richards, J. A.: Remote Sensing Digital Image Analysis - An Introduction. (2nd ed.) Springer-Verlag, 1993.

[2] Winkler, P. (ed.): Remote Sensing for Monitoring the Changing Environment of Europe. A. A. Balkema, 1993.

[3] Csornai, G., Dalia, O., Farkasfalvy, J., Nádor, G.: Crop inventory studies using Landsat data on large area in Hungary. Workshop on Agricultural Applications of Remote Sensing, Nottingham, Apr. 3-7, 1989. In: Applications of Remote Sensing in Agriculture, Butterworths, pp.159-165, 1990.

State Space Transformation — A Specification Language and a Program Generator

András Dezső and Vincze Bárány

Programming Methodology is one of the main introductory courses in computer science at the Eötvös Loránd University. We use a relational model of programming to establish a deep mathematical understanding of small-scale programming. The main issues covered by our model are

- deriving correct programs by refining specifications,
- encapsulation, representation and implementation of types and
- recognition of problem patterns and reuse of existing solutions.

Our programming methodology course lays much emphasis on the latter. Basic reuse is achieved by *reducing* the problem to a known and more general one and specialising the solution accordingly. As expressed above, basic reduction requires the problem to be the *direct specialisation* of a known one. Advanced reuse involving state space transformation relaxes this requirement and allows to exploit less obvious analogies as well.

In 1995 we started a project at our department to capture our programming methodology by a specification language and a program generator. Our goal was to separate the mechanical aspects of working according to the methodology from those requiring insight. The mechanical tasks have to be performed by a program generator that takes input written in a specification language that both enables and forces the programmer to state all creative knowledge necessary for solving the problem.

In our article we are going to introduce the current state of our work, which is the specification language called A' and the corresponding program generator. A' has language capabilities to express and solve problems using

- direct reduction,
- nested direct reductions or
- reductions involving state space transformation.

The specification language is designed to be as high-level as possible without the need to incorporate pattern recognition or artificial intelligence techniques into the program generator. The result is a tool that covers most of the exercises of programming methodology courses related to problem reduction and state space transformation.

Generation of Normal Control Data for the Evaluation of Myocardial Perfusion Studies

Marianna Dudásné Nagy, Eörs Máté and Béla Kári

A new mathematical method will be presented. The goal is to generate a normal circumferential profile curve which can be considered as normal control data for the evaluation of myocardial perfusion images.

The imaging methods of nuclear medicine are frequently used in cardiology. One of the most important imaging procedure is the myocardial perfusion (MP) study, which provides quantitative and qualitative data about the functionality of the human heart. The defects of myocardial functions (low perfusion), both the pre-infarction state and the infarction itself, can be detected by this kind of study. The evaluation of the circumferential profile curve (CPC) is used for representing some features of the left ventricle (LV). The CPC is the polar-curve of a planar image of the LV. In the evaluation of quantitative studies in nuclear medicine it is a general step to compare the parameters of the actual study with normal control data. So it is in the case of applying CPC.

The simplest way to collect normal parameter values is to determine them on the base of studies performed on a group of healthy persons (e.g. volunteers). If the study is risky or involves some hazard, like in the case of nuclear medicine studies, where radioactivity is used, then this kind of data collection raises ethical problems.

During the development of our software for evaluating MP studies there were no normal data. The question was how to get normal control CPC data. There were sets of patient studies available, but they could not be considered as a normal population s data. All the methods published previously required input data of a normal population.

Our study images were taken as the starting point of our iterative method to create normal CPC. Normal CPC was created from the patients' ones independently from the physician's diagnosis. Mathematical statistics, image analysis and understanding were applied during the development of our generation method. Some hypotheses about the available sample of the data in the original population:

Hypothesis 1. There are only few CPCs in which the total length of the low perfusion intervals is greater than 180^o.

Hypothesis 2. The distribution localisation of MP defects is approximately uniform in $[0^o,360^o]$.

The basic idea is to take all the available CP curves of the population and to leave out those intervals of the curves that show probably low perfusion. All other parts were considered as the data of a normal population. The normal CPC is the mean of the normal population's CPCs.

The starting population of our method consisted of 83 patientsśtudies. 57% of the population had MP defect. The result of the method was considered as the normal CPC for MP study.

The clinical evaluation software which uses our normal data was tested. The scoring of the software and the results of a physician were compared. The sensitivity of our results is 79%, the specificity is 95%, and the accuracy is 92% as opposed to the physician's scores.

The method for generation of normal CPC, the results and our future plans will be presented. The steps of the evaluation of a MP study will be overviewed.

IP Traffic Engineering over OMP technique

Károly Farkas, Zoltán Balogh, Henrik Villför

These days we live through significant changes with regard to using of Internet that is equipped with very constrained traffic management solutions. The demand on using eligible and efficient traffic management and engineering methods is increasing. This kind of efforts hallmarks the protocols/methods being under research/standardisation phase, such as OSPF-OMP, MPLS-OMP, MPLS-Traffic Engineering, etc. This paper discusses application of OMP (Optimised MultiPath) technique for Traffic Engineering purposes in IP networks and reviews OMP simulation results.

1 Introduction

IP traffic on the Internet and private enterprise networks has been growing exponentially for some time. Today the use of efficient traffic management solutions in IP networks is more and more inevitable. In this paper we first briefly review OSPF routing protocol, Multi-Protocol Label Switching (MPLS) to provide a background for Traffic Engineering, and OPNET simulation tool which was used to examine the behaviours of OMP technique. Then we discuss the general issues of OSPF-OMP and MPLS-OMP. Finally we present the simulation results in case of a basic network topology using OSPF, OSPF-ECMP, OSPF-OMP, MPLS and MPLS-OMP techniques and in the last section we summarise the results and future plans.

1.1 OSPF

OSPF (Open Shortest Path First) is an internet routing protocol [1]. It is classified as an Interior Gateway Protocol (IGP). This means that it distributes routing information between routers belonging to a single Autonomous System (AS). The OSPF protocol is based on link-state and SPF technology. In a link-state routing protocol, each router maintains a database describing the AS's topology. Each participating router has an identical database. Each individual piece of this database is a particular router's local state (e.g., the router's usable interfaces and reachable neighbours). The router distributes its local state throughout the AS by flooding. The possible routes to reach a destination node are computed by using of SPF algorithm.

1.2 MPLS

MPLS (Multi-Protocol Label Switching) [2] integrates a label-swapping framework with network layer routing. The basic idea involves assigning short fixed length labels to packets at the ingress to an MPLS cloud. The forwarding function of a conventional router involves a capacity-demanding procedure that is executed per packet in each router in the network. MPLS simplifies the forwarding function in the routers by introducing a connection-oriented mechanism inside the connectionless IP networks so label switched paths (LSP) are set up for each route or path through the network in advance using an IGP (e.g. OSPF). Throughout the interior of the MPLS domain, the labels attached to packets are used to make forwarding decisions (usually without recourse to the original packet headers).

1.3 OPNET

OPNET (OPtimised Network Engineering Tools) [3] is a simulation tool, which provides a comprehensive development environment supporting the modelling of communication networks and distributed systems. OPNET allows large numbers of closely spaced events in a sizeable network to be represented accurately. It uses a modelling approach where networks are built of nodes interconnected by links. Each node's behaviour is characterised by the constituent components. The components are modelled as a state-transition diagram. We used OPNET as our simulation environment.

2 OMP Technique

We proposed to study Traffic Engineering [4] issues in IP networks. TE is a vast research area and we focused our efforts to study OMP (Optimised MultiPath) technique with OSPF and MPLS.

2.1 OSPF-OMP

OSPF may form multiple equal cost paths between source-destination pairs. In the absence of any explicit support to take advantage of this, a path may be chosen arbitrarily. ECMP (Equal Cost Multi-Path) technique have been utilised to divide traffic somewhat evenly among the available paths. However an unequal division of traffic among the available paths is generally preferable. Routers generally have no knowledge of traffic loading on distant links and therefore have no basis to optimise the allocation of traffic. OSPF-OMP [5] utilises the OSPF Opaque LSA option to distribute loading information, proposes a means to adjust forwarding and provides an algorithm to make the adjustments gradually enough to insure stability yet provides reasonably fast adjustment when needed.

2.2 MPLS-OMP

In an MPLS network MPLS ingress routers may establish one or more paths to a given egress to the MPLS domain. Load can be balanced across a complex topology using MPLS. It requires that the ingress router is capable of computing a hash with a sufficiently fine level of granularity based on the IP source and destination and selecting a forwarding entry based on the outcome of the hash. MPLS-OMP [6] is an extension to MPLS. It does require that the IGP be capable of flooding loading information. At the MPLS ingress an algorithm is applied to select alternate paths where needed and adjust forwarding. Forwarding is adjusted gradually enough to insure stability yet fast enough to track long term changes in loading.

3 Simulation Results

Our main goal concerning TE was to implement and examine load-balancing policy using OSPF-OMP and MPLS-OMP technique in OPNET simulation environment.

3.1 Test Network Topology

Our basic test network is depicted on Figure 1. The router nodes represent IP-based gateways in case of OSPF simulations and ATM-based gateways in case of MPLS simulations. The terminals and the server are connected to the nearest router node by Ethernet connections. The link costs are the same on each link, the capacity of the links is 64 Kbps. The test network models an almost ideal IP network in which the possibility of packet loss is zero using infinite queues at the router nodes. The terminals made connections to the server node and offered the traffic. The amount of the offered traffic by Terminal_2, Terminal_1 and Terminal_3 was 35 Kbps, 23 Kbps and 12 Kbps, respectively. The duration of every simulation was 3 hours. Terminal_1 and Terminal_2 started to transmit the traffic towards the server at the beginning of the simulations while Terminal_3 started the transmission 2 hours later.

3.2 OMP Simulation Results

We have run five different tests on our test network. In these tests we used OSPF, OSPF with ECMP, OSPF with OMP, MPLS and MPLS with OMP techniques, respectively. In each case we monitored the throughput on the critical links of the network (Link_3 and Link_4). In the first and fourth tests (when we used pure OSPF and MPLS) we got the result which was expectable if the routing is based on shortest path calculation. If the shortest paths between different source and destination pairs contain the same critical link then link overloading can occur. (In our case the shortest paths between Terminal_1 server and Terminal_2 server contain the same low capacity link (Link_3) which got overloaded.) The throughput of critical links is depicted on Figure 2, Figure 3. In case of OSPF-ECMP test the traffic generated by Terminal_2 was shared evenly between the equal cost paths so the network could avoid overloading on the critical links (see Figure 4). However this sharing solution is static so the aggregate traffic can reach such amount that could cause overloading on the critical links. In OSPF-OMP simulation the throughput on the Link_3 and Link_4 was controlled by the OMP algorithm (see Figure 5). When the system sensed high load on the critical link, OMP was activated and it directed load from the most heavily loaded link toward less loaded links to achieve a steady state when the most heavily loaded links of the network are about equally utilised. The convergence time of the algorithm depends on the degree of the load on the network and the setting of the OMP's parameters. In case of MPLS-OMP test the throughput of the links was higher due to the relatively big amount of ATM header traffic. Comparing the resulted graph (see Figure 6) with the graph of OSPF-OMP we can see that its structure is similar to the other

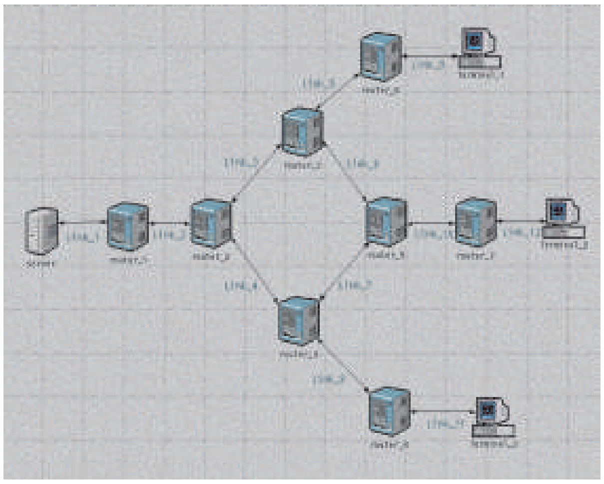

Figure 1: Test Network Topology

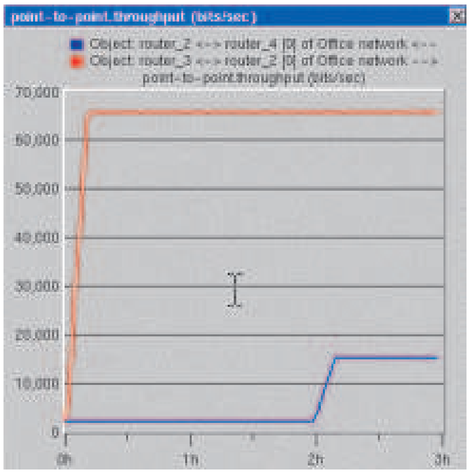

Figure 2: Throughput in case of MPLS

one due to the simple topology of the test network. The curves follow the changes of loading and converge to a state where the loading on the most heavily loaded links are equally distributed.

4 Conclusions and Future Plans

Today the importance of use of efficient traffic management solutions in IP networks is more and more increasing. If the offered traffic toward a destination is too high link overloading can be caused in the network using only pure OSPF, MPLS. OMP technique solves this problem by flooding loading information across the network and balancing load between the alternative paths. In this paper we examined some basic features of OSPF-OMP and MPLS-OMP techniques by implementing them in OPNET simulation environment. The simulation results back up our expectations that OMP decreases the load of the most loaded links and increases the utilisation of the network. Our future plans are to examine OMP in case of more different network topology by simulation moreover we would like to implement and measure OMP on a real MPLS network.

References

[1] OSPF Version 2 , IETF RFC 2328, www.ietf.org/RFC/RFC2328

[2] Multiprotocol Label Switching Architecture, Internet Draft, www.ietf.org/internet-drafts/draft-ietf-mpls-arch-06.txt

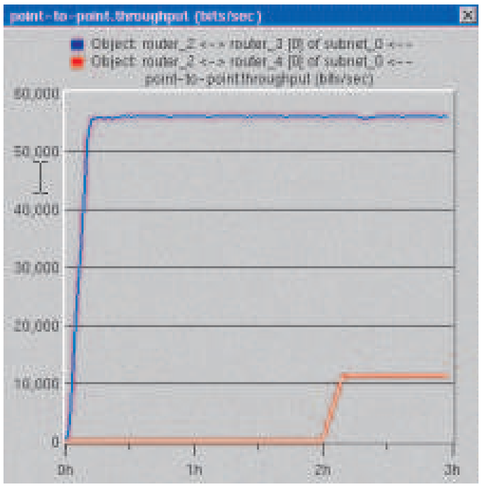

Figure 3: Throughput in case of OSPF

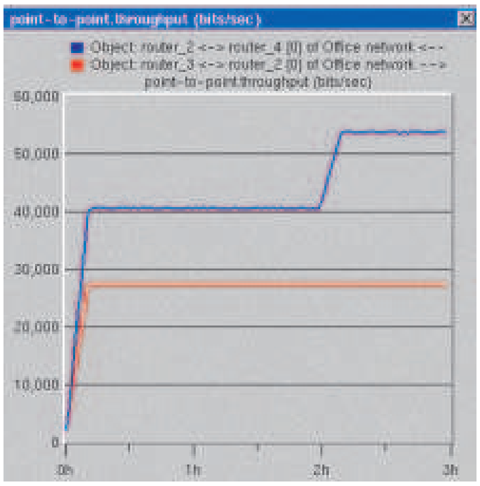

Figure 4: Throughput in case of OSPF

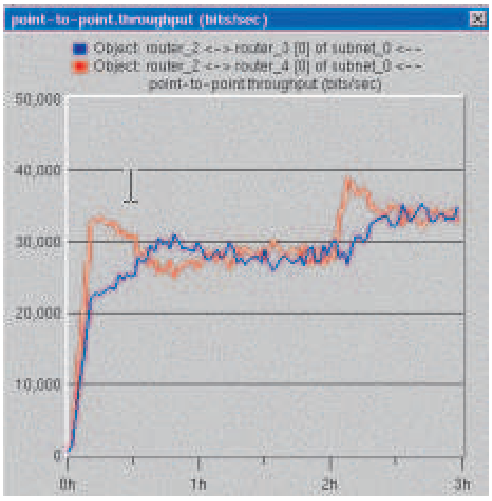

Figure 5: Throughput in case of OSPF-OMP

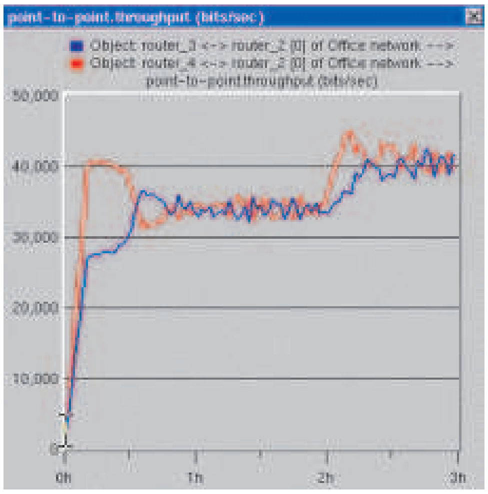

Figure 6: Throughput in case of MPLS-OMP

[3] OPNET Online Manual

[4] Requirements for Traffic Engineering over MPLS, IETF RFC 2702, www.ietf.org/RFC/RFC2702

[5] OSPF Optimised Multipath (OSPF-OMP), Internet Draft, www.ietf.org/internet-drafts/draft-ietf-ospf-omp-02.txt

[6] MPLS Optimised Multipath (MPLS-OMP), Internet Draft, www.ietf.org/internet-drafts/draft-ietf-mpls-omp-01.txt

Wireless Application Protocol Performance Testing

Lóránt Farkas and Lajos Nagy

Recently the implementation of a standard and the elaboration of adequate performance testing procedures and specifications for those equipments complying to that standard tend to be simultaneously achieved. The reasons for this phenomenon lie in the short lifecycle of some new standards due to the development of technology that pushes from behind towards new directions and new emerging standards and obsoletes existing ones.

This is also the case for the WAP protocol stack, which has recently been implemented and already widely used, in spite of the fact that reliable benchmarks and performance testing specifications have not been formulated yet.

We propose in this early stage of our work, to draw the main guidelines of the procedures of WAP performance testing, to establish the main steps in elaborating a testing specification for the protocol stack and the WAP network components and to capture and briefly analyze those elements of the WAP network, that essentially need a well-defined set of performance testing procedures.

In order for a standardized test methodology to be constructed, at least two basic elements of the WAP network has to be tested in a standard form: the WAP server and the physical bearer. Therefore two different test procedures, one for the bearer, the other for the WAP server, have to be elaborated.

The performances of the WAP server can be evaluated through testing how the implementation of the protocol stack, more precisely the Wireless Session Protocol (WSP) behaves in prescribed testing conditions, in a similar fashion that well-known benchmarks, like for example SPECWeb, prescribes standard testing specifications for the http protocol. An appropriate workload generator, standard WSP clients, a standard testing architecture and an adequate performance metric have to be defined as close to the real-world conditions as possible. This is a daunting task, since WAP protocol hasn't been in use for such a long time in order for known real-world conditions to exist. Therefore application classes and the respective average required data rates and file sizes have to be defined first, based on previous measurements on test systems. The performance metric of the testing procedure will to be linked to the maximum number of simultaneous connections while still meeting specific throughput and data error rate requirements.

The other aspect, concerning the performances of the bearer, has to be solved by a bearer emulator, which we intend to implement first for three different bearer types: GSM-SMS, GSM-USSD and IP. Our practical approach for the emulator is a workstation or PC having two separate ethernet cards. The software will delay the packets arriving from one card through a programmable delay corresponding to the simulated bearer and pass over to the second ethernet card. The WAP server and the workload generator will be linked to the separate cards and the overall bearer delay is being emulated by the delay between packet input and output through the two cards.

References

[1] The SPECWeb99 Benchmark, Standard Performance Evaluation Corporation, November 1999

[2] The Wireless Application Protocol Specification, Version 1.2, WAP Forum, November 1999

Auto Case Test Selection from WAP Test Suites

Antal Fazakas

The standardization institutes, delivers several standards in connection with conformance testing, such as Protocol Implementation Conformance Statement (PICS), Protocol Implementation eXtra Information for Testing (PIXIT) for describing the testing environment, Protocol Conformance Test Report (PCTR), and System Conformance Test Result (SCTR) for the proforma of the test results. The test suite on which the testing is based is also standardized for the given protocol, as the Abstract Test Suite (ATS).

For arrange the test selection in relation with a protocol stack previously should make the followings:

- filling the PICS, PIXIT tables or editors, if it is possible with help an iterative decision-helped system, and after analyze them (in SQL for example) advert the prerequisite, mandatory, optional an conditional logic within the PICS table items
- studies the relation between these documents, and the acceptable TTCN ATS test suite
- additionally ascertain between the PICS-TSS and TP and ATS and PCTR relation
- requirements and the claims of the auto case select editor

The most important problem in conformance testing is the enormous number of the possible test suites, and of course the long time needed to run them all, so the test selection is required.

The test cases create a test tree and from these should be selected the test branches according the PICS. The selection is based on adequate mathematical algorithm. The example protocol is from the mobile telecommunication world.

The **Wireless Application Protocol** (WAP) is the de-facto world standard for the presentation and delivery of wireless information and telephony services on mobile phones and other wireless terminals. The WAP specification was developed by the industry's best minds to address these issues. Wireless devices represent the ultimate constrained computing device with limited CPU, memory, and battery life, and a simple user interface. Wireless networks are constrained by low bandwidth, high latency, and unpredictable availability and stability. However, most important of all, wireless subscribers have a different set of essential desires and needs than desktop or even laptop Internet users.

Within the WAP, the **Wireless Transaction Protocol** (WTP) runs on top a datagram service and optionally a security service. WTP has been defined as a light weight transaction oriented protocol that is suitable for implementation in "thin" clients (mobile stations) and operates efficiently over wireless datagram networks.

The test case selection method is illustrated with WTP protocol.

The example model protocols, and its conformance statement is specified in the following materials:

- WAP Forum, WAP Client Implementation, Conformance Statement, Prototype 21 - September-1999
- WAP Forum, WAP Server Implementation, Conformance Statement, Prototype 21- September-1999 References

References

[1] K. Tarnay, Protocol Specification and Testing, Plenum Press New York, 1991

[2] M. Toro, K. Tarnay Principles of Validation of abstract test suites specified in concurrent TTCN. PSTV'95 Warsaw, Poland, 13-16 June, 1995

[3] prETS 300 406: "Methods for Testing and Specification (MTS); Protocol and profile conformance testing specifications; Standardization methodology".

[4] ISO/IEC 9646-2 (1994): "Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Part 2: Abstract Test Suite Specification".

[5] T. Csondes, B. Kotnyek A Mathematical Model in Test Selection, Euromicro 97, Proceedings of the 23rd Euromicro Conference, Budapest Hungary Sept 1-4, 1997

Low Complexity Parametrized Codes for LZ77 Compression

Péter A. Felvégi

The family of LZ77 derived compression algorithms is rather long. Newer and newer variants have been proposed since the publication of the original algorithm in 1977. Of these, some have focused on how to improve the match finding part in terms of temporal and spatial complexity, the others delt with the coding part of the algorithm. Nowadays, the most widely used popular compression programs also use some variation of the LZ77 algorithm (such as zip, gzip, arj, rar, zlib). These programs can achieve good compression ratios in acceptable time, though they sacrifice some compression for being 'fast'. The resource constraints are usually the same or similar for compression and decompression, these programs are usually run on PC's or workstations.

In some applications the resource constraints for the compression and decompression parts might be remarkably different. This is the case generally when compression takes place only once and decompression many times, e.g.: at embedded systems, static databases, executable files, distribution CD's, etc. For such applications, the main aim is to maximize compression while keeping the decompressor complexity as low as possible in terms of time and maybe for memory requirements, too. For the compression part, the constraints are far less strict. Since compression takes place only once, it can be done off-line using reasonable amount of computing power and memory.

With the above assumptions, the LZ77 algorithm can be improved in two ways: better parsing during the string matching part; and better coding of the literals, distance and length values. In this article we will focus on efficiently encoding the match lengths.

The range of the match lengths is between the minimal allowed (typically 2,3) and the maximal allowed (typically 256-64K) values. On this range, the distribution of the values tend to be asymptotically decaying, thus entropy coding offers a gain against the flat binary code. We omit adaptive codes and also omit static Huffman coding, because we want to keep decompressor complexity at a minimum.

Simple codes are already proposed by several people (Elias, Golomb, Rice, Fiala, et al.) for representing integer numbers of an assumed distribution. In this article I will present a family of codes that are parametrized by a few integers, and are capable to adapt better to the actual distribution of the values, thus increasing the coding efficiency. With the proper set of parameters, some of the cited codes can be achieved, too. The drawback of the parametrized codes is that the parameters must be optimized first, thus compressor complexity is usually higher than with the other codes – which was the original assumption.

I will present the comparison of these codes to the others, and also to Huffman codes and to the theoretical optimum based on the entropy. The work is still in progress, other parametrized codes may be found later for a special purpose/distribution.

Code Generation from UML models [4]

Ákos Frohner, Zoltán Porkoláb and Dr. László Varga

Creating a generic, object-oriented, component-based, transactional business system, which covers the whole lifecycle, is possible only with integration of commercial tools, component technologies, newly developed class libraries and using code generators. Most of the recently used tools for development techniques are focusing only of one the layers of the model from the code generation point of view. As a consequence the inter-layer connections are lost in the generated code.

In this article we describe a code generator technique, which uses a UML model as a starting point and generates the following layers directly:

- Presentation layer (clients), which provides the user interface.
- Business logic layer. This can be a physically and geographically distributed N-tier architecture:
  - Business objects: correspondents of business entities. They implement the basic behaviour of the entities including persistency.
  - Business processes: a series of operations on the business objects.
- Storage layer: either relational or object-oriented databases, special storage systems and legacy systems.

Meanwhile generating the code it preserves the original inter- layer relationships originated in the model.

Based on the experiences with 4GL systems it is obvious the need to provide customisation in the generated code. We offer a multi-paradigm approach [1] to let the developer to choose the appropriate solution for her or his implementation:

- Polymorph code sections: the calling part requests a general behaviour, the called code section substitutes a specialised version according to the context. [2]
- Run-time parameterisation of the generated code: the calling part requests a specialised behaviour using a metadata parameter in the routine call.
- Aspect weaving: the modifications (new aspects) of the code are placed in a separated source file, which is woven into the generated code before the compilation. [3]

References

[1] Jim Coplien: Multi Paradigm Design for C++, 1998, Addison- Wesley, ISBN: 0-201-82467-1

[2] Barbara Liskov: Data Abstraction and Hierarchy, SIGPLAN Notices 23, 5. May 1988

[3] Gregor Kiczales: Aspect Oriented Programming, AOP Computing surveys 28(es), 1996 p.154

[4] This work has been supported by the Grants OMFB ALK-00229/98

Parallel Implementation of the Large Sparse and Symmetric Eigenproblem

E. M. Garzón and I. García

The aim of this work is to solve the eigenproblem for large symmetric and sparse matrices through a divide–and–conquer strategy put forward by Cuppen [1, 2]. This method has been implemented on a Cray T3E multiprocessor system, using PVM parallel interface and a detailed analysis of the obtained performance has been carried out.

The solution of the eigenproblem has been decomposed in the next phases:

- *Obtaining a structured matrix from the input matrix* (A). In this phase, both a tridiagonal matrix (T) and an orthonormal matrix Q are obtained from A, where T is similar to A. These matrices hold the relation $A = QTQ^T$. The Lanczos Method with Complete Reorhtogonalization has been implemented to be carried out at this stage [3, 4, 5].
- *Solving the eigenproblem of* T. The goal of this stage is to compute the diagonal D and orthonormal M matrices. These matrices hold the relation $T = MDM^T$. The divide-and-conquer method (dc) has been implemented to solve this problem. This method is decomposed in the following stages:
  1. Decomposition in Sp subproblems (with $Sp = 2p$), through the Rank-One Modification of T.
  2. Solving eigenproblem of each subproblem through the QR algorithm.
  3. Reconstructing. The solution of the eigenproblem for each link of subproblems is computed through the solution of each. Once it has been determined the solution of all the links of the same dimension, other stage of the reconstruction begins in which pairs of previous stage subproblems are linked. As the process develops, the number of subproblems is halved, until just one link of is solved. In this way, the graph of tasks connected with the development of this stage can be represented through a binary tree.
- *Computing the eigenvectors of* A. If G denotes the orthonormal matrix whose columns are the eigenvectors of A, then $A = GDG^T$. On the other hand, the output matrices generated in the previous stages hold the relation $A = QTQ^T = QMDM^TQ^T$, so $G = QM$. Thus, this stage is carried out with a matrices product.

The eigenproblem of a sparse matrix is decomposed in several procedures whose performance is sequential, therefore it has been established a parallel implementation for each procedure in an independent way. Nevertheless, in order to link all the procedures in parallel, the output data distribution of a procedure establishes the parallel implementation of the procedure that receives these data as input.

The parallel implementation in phase 1 (Lanczos) is based on a decomposition in domains of input matrix. The input data are irregular, since A is sparse. Thus, a stage of preprocessing has been designed, namely Pivoting-Block, that is quick and guarantees that the input data partitions are balanced and also the operations associated to them [6, 7].

Then the parallel implementation of the Cuppen's method is based on decomposition in domains in both, the input and output data. The binary tree of tasks is distributed so that the same number of branches of the tree is allocated to each PE. If P refers to the number of PEs of the multiprocessor system and $P < Sp$ then initially each PE starts the reconstruction process independently, until $Sp = P$. Then groups of PEs that collaborate in the solution of a pair of subproblems are defined. As the reconstruction stages develop the subproblems are of a greater dimension and the number of PEs that collaborate is larger too.

The product $QM = G$ is carried out starting with a partition of Q and M by rows among PEs. This partition causes a remarkable penalization in the communications that should be established. It will be proved that the phases with the largest computational cost are the first and the last. This implementation will be evaluated for matrices of dimension between 782 and 7168, obtaining superlineal speed-up in most cases. It will be checked how the superlineal speed-up is due to low efficiency management of memory hierarchy with few PEs. It wil be proved that the management improves substantially as the number of PEs is increased.

References

[1] Cuppen, J.J.M, "A Divide and Conquer Method for the Symmetric Tridiagonal Eigenproblem", Numer. Math, vol. 36, pag. 177-195, 1981.

[2] Dongarra, J.J. and Sorensen, D.C " Fully Parallel Algorithm for the Symmetric Eigenvalue Problem", SIAM J. Sci. Stat. Comput., vol 8, num. 2, pag. 139-154, 1987.

[3] Paige, C.C. "Practical Use of the Symmetric Lanczos Process with Re-Orthogonalization.", BIT, 10, pag. 183-195. 1970.

[4] Paige, C.C. "Accuracy and Effectiveness of the Lanczos Algorithm for the Symmetric Eigenproblem.", Linear Algebra and its Applications, 34, pag. 235-258, 1980.

[5] Parlett, B.N. and Nour-Omid, B. "Towards a Black Box Lanczos Program.", Computers Physics Communications. Vol 53, pag 169-179, 1989.

[6] Garzón, E.M. and García, I. "Parallel Implementation of the Lanczos Method for Sparse Matrices: Analysis of the Data Distributions." Proceedings ICS'96. 1996.

[7] Garzón, E.M. and García, I. "Evaluation of the Work Load Balance in Irregular Problems Using Value Based Data Distributions." Proceedings of Euro-PDS'97, pag 137-143, 1997.

Optimizing Object Location in Mobile Distributed Object Systems

Gábor Gesztesi

Cellular wireless networks are becoming increasingly common access systems. This trend leads to an increasing need for high level services to be available in mobile environments. Distributed object technologies offer good solutions for complex service needs. However, these technologies, as they exist today, are not perfectly suited for wireless access. They need not only technical refinements, but their theoretical background needs to be extended as well.

One important mobility-specific issue in distributed object systems is object location and relocation according to the movements of the mobile units. In practice it is very common, that mobile clients wish to access services offered by servers in the fixed network. The services are available through method calls to remote objects representing the service sessions. Depending on the characteristics of the concerned service the following requirements may apply on the location of the objects:

1. **Fixed location** - The object must stay at its original location
   - *Predefined location* - The object must reside on a predetermined server (eg. centralized database access)
   - *Unmovable object* - The object cannot be relocated (eg. printing services)
2. **Nearest location** - In this case a nearest server is assigned to each cell. The object must be located on the server assigned to the current cell (eg. local time service)
3. **Optimized location** - The object may be located on any of several servers and may freely be relocated. Therefore the location of the object may be optimized by the system (eg. application servers)

Our work focusses on the third case. We set up a model for optimizing the location and relocations of the objects. Our model consists of a traffic model, covering the activity and movements of mobile units and a cost model for method calls and object transfers.

We tried to reuse existing traffic models used to dimension mobile network elements. These models fall into two main categories differing by their view of the network. *Traffic source models* describe the system from the mobile unit's point of view, while *network traffic models* describe the traffic as seen by the fixed network elements, like base stations. We found the network approach unsuitable for us, because it can hardly deal with the relative location of mobile units and object implementations. The traffic source model also needed some modifications.

We also present the optimal object location and movement strategy in our model. Certain parameters of the optimal strategy depends on the concrete topology of the network (the arrangement of mobile cells). We evaluated and analysed these parameters for a couple of simple regular topologies and give a method to evaluate them for other topologies.

Automatic Code Generation from SDL to a Functional Programming Language for Safety Critical Systems

László Gombos

Safety critical system is a system where human safety is dependent upon the correct operation of the system. Although safety critical systems have been in use for many years, the development of safety critical software is still a relatively new and immature subject. Our approach is to use functional languages for high reliable softwares, which allows higher abstraction level in the implementation and allows greatly improved modularisation. Functional programming languages provide two new kinds of modularisation technique - higher order functions and lazy evaluation.

We have choosen Clean as a target functional language. It is a strongly typed language with support for unique types. Unique arguments (such as the environment of the system) allows destructive updates preserving the referential transparency and makes it possible to improve the efficiency of the program execution.

Clean provides an extensible library (Object I/O library) to create interactive applications by composition of concurent, interleaved communication processes. Besides, the library has support for abstract devices such as timer and receiver objects.

The Specification and Description Language (SDL) is a standardized Formal Description Technique (FDT) for the specification of discrete reactive systems. SDL models the system as a number of concurent, communicating process instances interchanging signal instances with each other and with the environment of the system. Each process instance is an extended state finite state machine. The communication is based on asynchronous passing of parametral signal instances from one sender to one receiver. Time-dependencies can be modelled by means of timer instances and with the help of the Time and Duration predefined data types.

SDL is widely used for specifying and implementing huge and complicated telecommunication systems and protocols. The existing code generation tools, however, generate mostly C code, which is poorly suitable for safety critical systems.

In our project, we define some auxiliary Clean functions and semantically map SDL systems to Clean programs. This paper describes a code generator mechanism from SDL/PR to Clean. SDL objects, such as channels, process types and signals are defined on a high level of abstraction using algebraic datatypes. Procedure, process type and signal specialisation is discussed in details, and a mechanism is shown how to model unreliable system components with the help of spontaneous transitions.

References

[1] CCITT, *Recomendation Z.100, Specification and Description Language SDL*, 1993.

[2] Peter Acten: *A Tutorial to the Clean Object I/O Library - Version 1.2*, University of Nijmegen, 2000.

[3] Zoltán Horváth, Peter Achten, Tamás Kozsik, Rinus Plasmeijer: *Verification of the Temporal Properties of Dynamic Clean Processes*, In: Koopman, P., Clack. C. eds. Proceedings of the 11th International workshop on the Implementation of Functional Languages, IFL'99, Lochem, The Netherlands, September 7-10, 1999, pp. 203-218.

Parallelization of an algorithm for non rigid objects detection with stochastic global optimization

J.M. Gonzalez-Linares, P.M. Ortigosa, N. Guil and I. Garcia

In many real applications objects suffer different deformations due to several factors like sampling errors or the flexibility of the material. Many algorithms have been developed to recover such objects. Some of them are based in the use of free-form deformable models like "snakes". Others are based in parametric models, where an object matching algorithm, combined with this deformation model, is used as a Bayesian objective function that is minimized by a gradient descent method.

In this talk, a parallel algorithm for detecting rigid and non-rigid objects in an image will be presented. This algorithm uses a Bayesian inference scheme to register the parameters that describe the object of interest included in the image. This kind of techniques requires knowledge about the shape of the object. This prior information of the object shape is composed of a prototype template, a set of deformation transformations of the template, and a probability density on the set of deformation transformations.

The likelihood measures the similarity between the deformed template and the objects present in the image. A Generalized Hough Transform (GHT [1, 2]) is used to detect the deformed template in the image and to measure the similarity using a defined energy function. The prior information is combined with the likelihood to obtain the *a posteriori* probability. Finally, by obtaining the maximum of this probability (Maximum A Posteriori), the deformed template is identified in the image. This MAP estimate is computed using the evolutionary global optimization algorithm UEGO (stands for Universal Evolutionary Global Optimizer, [3, 4]).

Our proposed algorithm is very robust and it is able to detect non-rigid objects even in the presence of noise or occlusions. Unfortunately, it is computationally very expensive. To solve this disadvantage a possible parallelization of the algorithm is proposed. The parallel strategy consists on a master-slave configuration. The tasks are distributed among the processors to balance the computational load. It has been implemented using the message-passing paradigm with PVM. The parallel program has been tested in a Cray T3E using up to 32 processors.

References

[1] D.H. Ballard. “Generalizing the Hough Transform to detect Arbitrary Shapes”. Pattern Recognition, 14(2), 1981, pp. 111-122.

[2] N. Guil, J.M. Gonzalez-Linares, and E.L. Zapata. “Bidimensional shape detection using an invariant approach”. Pattern Recognition, 32(6), 1999, 1025-1038.

[3] P.M. Ortigosa, M. Jelasity, and I. Garcia. “Uego, an evolutionary algorithm for solving global optimization problems”. Journal of Heuristics, Kluwer Academic Publishers.

[4] M. Jelasity, P.M. Ortigosa and I. Garcia. “Description of two parallel implementations for UEGO algorithm”. Matrahaza, Hungary, March 1999. XIV Intl. Conf. on Mathematical Programming.

Automatic Test Purpose Generation based on Formal Grammar Representation of Communication Protocols

Barnabás Gőgös

Formal grammars are a new, simple and efficient representation of communication protocols. The great advantage of using formal grammars is a wide scale of tools and algorithms for generating and handling them. In this paper a method will be introduced for generating test purposes from a formal grammar protocol specification. Additionally we show that the usual tools for handling formal grammars are suitable for generating test purposes automatically.

The behavioral equivalence between the specification of the protocol and the implementation is theoretically possible, but the practical implementation is a very hard problem. In the case of real systems and protocols, the state space can be extremely big or infinite so testing of all state transitions is impossible. Thus, instead of the behavioral equivalence a set of test cases are used to test the most important aspects and behaviors of protocols. Usually test cases are based on test purposes. A test purpose declares an important feature of the protocol, but it is usually informal, so there can be contradiction between the test purposes and the specification. Furthermore, if a new version of the protocol appears, the upgrading process of test purposes is very complex and test cases can not be derived automatically from the formal protocol specification.

To cope with these problems, several proposals were made to express test purposes formally. Test purposes usually contains observable events such as abstract service primitive calls or receiving and sending protocol data units. Message Sequence Chart is adequate language for expressing these events, so it is suitable to express the formal test purposes. But how can Message Sequence Chart diagrams be generated from formal grammars? Test purposes express a relevant behavior of the protocol. This definition is adequate in case of informal test purposes, but does not satisfy the requirements in case of formal ones. In this work the formal test cases are defined as syntactically valid sentences in the formal grammar that representing the communication protocol. These sentences can be derived from the formal grammar specification by the help of formal grammar parsers. The result of this definition that the usual tools for parsing formal grammars can be used for generating test purposes automatically and if the protocol specification is changed, the upgrade process of test purposes can be made automatically.

In the second part of the paper an example will illustrate, the formal grammar specification of a simple protocol. Based on the definition of formal test purposes all the syntactically correct sentences will be generated automatically and finally, a Bison parser will be used to translate these sentences to Message Sequence Chart diagrams.

The History of Software Localization

László Gulyás

There are different possible ways to arrange the developments of programming technology (from its very beginnings to nowadays' trends) into one logical and coherent line of evolution. One of these options is to discuss the subject as a process of increasing software localization. According to this idea the history of computer science is dominated by the classical *divide and conquer* rule. That is, facing the more and more complex problems to solve (which was enabled by the increasing power of the underlying hardware), programmers needed more and more localization of (concentrated focus on) their software elements. Whatever trivial this idea may seem at first, it has the distinctive power to unify different programming techniques and, at the same time, to separate ones which are close to each other but contain very important underlying conceptual differences.

According to this view, monolithic programs first got divided into separate *compilation units* and *subroutines* within each source file. Later on, the movement of structured programming further localized the 'behavior' of the software by introducing the concept of *block* and made the first step toward data-localization through *user-defined types*. The increasing acceptance of the *Abstract Data Type* concept has finalized the localization of data (pairing that of code or 'behavior') by *modules* on one hand, and *object-orientation* on the other. Recently, these concepts have been furthered by standardization and continuity in time in *component architectures*. The last step along the pathway of increased localization, up to now, is represented by *autonomous agents* (according to the *weak definition* of agency), where localisation of code and data is completed by that of its control.

In this paper, we will elaborate on the idea described above, reporting on a survey of programming languages (including C, Pascal, Module-2, Objective-C, C++, Ada, Java). This survey conceptualizes the different techniques the surveyed languages use to handle different aspects of software localization. As a result of this work, we construct a taxonomy of programming languages containing classes of increasing software localization. We also present concrete, implemented instances for the defined classes. These examples show that some of the localization aspects span across a variety of programming languages, while others are rather specific to some environments.

In addition to the classification above, our results contribute to the explanation of the driving forces behind popular trends of today's computer science. For example, as sketched above, they make a clear and logical connection between such techniques as object-orientation, autonomous agents and component architectures.

A Fuzzy Approach for Mining Interesting Quantitative Association Rules

Attila Gyenesei

During the last ten years, data mining, also known as knowledge discovery in databases, has established its position as a prominent and important research area. Mining association rules is one of the important research problems in data mining. The problem of mining boolean association rules over basket data was introduced in [Agrawal et al.,1993]. Given a set of transactions, where each transaction is a set of items, an association rule is an expression of the from $X \Rightarrow Y$, where X and Y are sets of items. An example of an association rule is: "40% of transactions that contain beer and potato chips also contain diapers; 5% of all transaction contain all of these items". Here 40% is called the confidence of the rule, and 5% the support of the rule.

The problem of mining quantitative association rules was introduced in [Srikant et al.,1996]. For illustration, a table with three non-key attributes is shown in Table 1. Age and NumCars are quantitative attributes, whereas Married is a categorical attributes.

RecordID	**Age**	**Married**	**NumCars**
5	23	No	1
8	25	Yes	1
14	27	No	0
18	38	Yes	2
19	39	Yes	2

(min. support = 40%, min. confidence = 50%)

Rules (Sample)	**Support**	**Confidence**
$\langle Age : 30..39\rangle$ and $\langle Married : Yes\rangle \Rightarrow \langle NumCars : 2\rangle$	40%	100%
$\langle NumCars : 0..1\rangle \Rightarrow \langle Married : No\rangle$	40%	66.6%

Table 1: Example of Quantitative Association Rules

Several efficient algorithms for mining quantitative association rules have been published (see [Srikant et al.,1996] for an example). The algorithms find the association rules by partitioning the attribute domain, combining adjacent partitions, and then transforming the problem into a binary one. Although these quantitative association rule mining algorithms can solve some problems introduced by quantitative attributes, they introduce some other problems. The first problem is caused by the sharp boundary between intervals. For example, as shown in Figure 1, suppose $[11, 20]$, $[21, 30]$ and $[31, 40]$ are three intervals created on a quantitative attribute, and the minimum support is 40%.

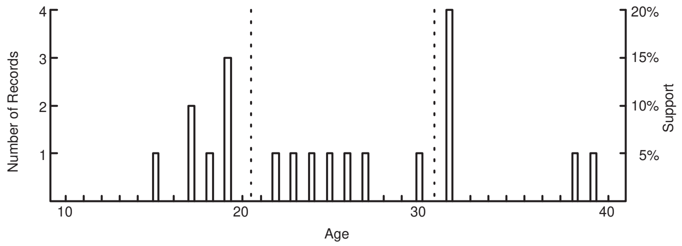

Figure 1: Example of Sharp Boundary Problem

In this case, none of these intervals will have enough support. However, the interval $[21, 30]$ should be interesting if we consider the values near both sides.

In this paper, we deal with mining fuzzy quantitative association rules of the following form:

If $X = \{x_1, x_2, \ldots, x_p\}$ is $A = \{f_1, f_2, \ldots, f_p\}$ then $Y = \{y_1, y_2, \ldots, y_q\}$ is $B = \{g_1, g_2, \ldots, g_q\}$,

where

$f_i \in$ {fuzzy sets related to attribute x_i}, $g_j \in$ {fuzzy sets related to attribute y_j}

and X, Y are itemsets, A and B contain the fuzzy sets associated with the corresponding attributes in X and Y. "X is A" is called the antencedent of the rule while "Y is B" is called the consequent of the rule. With fuzzy sets, a person may be both a member of "old" with 80% membership, and also a member of "middle-age" with 20% membership. An example of such an rule might be "If *Age* is *old* then *NumCars* is *small*".

In this paper we present a new algorithm for mining association rules based on fuzzy set theory. The algorithm uses new definitions for interesting measures. Finally, we describe the results of using approach on a real-life dataset.

References

[Agrawal et al.,1993] R. Agrawal, T. Imielinski and A. Swami (1993), *Mining association rules between sets of items in large databases*, ACM SIGMOD, Washington, DC, USA, pp. 207–216.

[Srikant et al.,1996] R. Srikant and R. Agrawal (1996), *Mining quantitative association rules in large relation tables*, Proceedings of ACM SIGMOD, pp. 1–12.

New Results in 3D Surface Reduction

Csongor Halmai, Balázs Erdőhelyi, Krisztián Ollé and Attila Kuba

The mechanical simulation of the human bones is an important task in the treatment planning, for example, to analyse the mechanical features of the components of the bone system, or to test the possible operational solutions from the viewpoint of tension.

In order to give quantitative mechanical results, some finite element analyzer software tool (e.g., Cosmos/M by Structural Research and Analysis Corporation) should be applied to the geometrical model of the simulated bone structure. As input data for the creation of the geometric model, CT images are used. In our presentation we show the steps of the geometric modelling. One of the most difficult problem that the bones are described with some hundreds of thousends of surface elements. On the other hand the finite element analyzer software can not analyze so huge models. So the task is to reduce the amount of data describing the model while its shape should be kept as much as possible. The shape is visualized by its VRML model. During the simplification method polyhedral estimation of the shape of the object is used and homogenuos bone material is supposed.

In this presentation the steps of the method are described. The difficulties, possible solutions and reached results are shown.

Debug Slicing of Logic Programs

László Harmath, Gyöngyi Szilágyi and Tibor Gyimóthy

Slicing methods are widely used for the debugging, testing and maintance of imperative programs. Intuitively, a slice should contain all those parts of a program that may affect the variables in a set V at a program point p. Slicing algorithms can be classified according to whether they only use statically available information (*static slicing*), or compute those statements which influence the value of a variable occurrence for a specific program input (*dynamic slice*). Dynamic slicing methods are more appropriate for debugging than static ones as during debugging we generally investigate the program behaviour under a specific test case. The main advantage of using a dynamic slice during debugging is that many statements can be ignored in the process of bug localization.

Different dynamic slicing methods have been introduced for debugging imperative programs [3]. Most of these methods are based on a dependence graph which contains the explicit control dependences and data dependences of the program. In [1, 2] a slicing method was introduced for logic programs, and this method being used to improve the efficiency of the Shapiro's algorithmic debugging algorithm. The slice presented in [1] contains those parts of a program that actually have an influence on the value of an argument of a predicate. This type of slice (called *data flow slice*) is safe if the structure of the proof tree for a goal is not changed.

However, during debugging we also need to identify those predicates that actually did not affect an argument in a predicate but could have affected it had they been evaluated differently. We can say that these predicates are in the *Potentially Dependent Predicate Set*. We note that the different evaluation of the predicates in this set could change the structure of the success branch of the proof tree.

In this paper we introduce a new type of slicing called *Debug slicing* for Prolog programs without side effects. A Debug slice of an Augmented Proof Tree includes all those predicates that may affect the value of an argument in any success branch's predicate. So this slice is very suitable for debugging. The Debug slice is the set of predicates which contains the predicates of the success branches of the SLD-tree, the Potentially Dependent Predicates and their data dependences.

This slicing method has been integrated into an interactive algorithmic debugging tool to reduce the number of questions to the user during a debugging session [2]. The size of the debug slice is larger than the size of the data flow slice, however data flow slice is not safe for debugging. On the other hand the Debug slice contains all parts of the program that may be responsible for the incorrect behaviour at a selected argument position.

References

[1] T. Gyimóthy and J. Paakki: Static Slicing of Logic Programs.*In Proceedings of Second International Workshop on Automated and Algorithmic Debugging (AADEBUG'95)*, pages 85-105, Saint Malo, France, May 1995.

[2] G. Kókai, L. Harmath, T. Gyimóthy: Algorithmic Debugging and Testing of Prolog Programs. *In Proceedings of the 14th International Conference on Logic Programming (ICLP'97). Eighth Workshop on Logic Programming Environments*, pages 14-21, Leuven Belgium, July 1997.

[3] F. Tip: A survey of Program Slicing Techniques. *Journal of Programming Languages*, Vol.3., No.3, pages 121-189, September, 1995.

Motion planning algorithms for stratified kinematic systems with application to the hexapod robot [5]

István Harmati and Bálint Kiss

The paper addresses the motion planning problem (MPP) of kinematic systems where the configuration space is stratified.

Roughly speaking, the MPP can be defined as a search among all feasible trajectories of the system that satisfy a desired control objective i.e. to move the system from a known initial state (or configuration) x_i to a desired final state x_f. Feasibility means that, along such trajectories, the system variables identically satisfy the system equations.

In this paper we restrict our attention to kinematic and stratified systems. Typically, such systems arise in the presence of non-holonomic constraints [4]. The state equation of a kinematic system reads:

$$\dot{x} = \sum_{i=1}^{m} g_i(x)u_i \quad x \in \mathcal{R}^n \tag{2}$$

where x is the state variable, u is the input variable and g_i are (sufficiently regular or smooth) vector fields on $\mathcal{R}^n$ $(n \geq m)$.

The systems considered here are also said to have stratified configuration space, notion introduced by Goodwine et. al. in [3]. For such systems, the configuration space is divided in strata (set of intersecting submanifolds of the configuration manifold) such that the equations of motions may differ for each strata and change discontinuously. We call the bottom stratum the submanifold with the highest co-dimension.

Legged robotic structures are typical examples for stratified systems where the possible leg contact configurations are defined by constraints. Since these constraints define submanifolds in the configuration space where the equations of motions are different, one obtains a stratified system.

The main problem arising in the control of such systems is that the bottom startum is not controllable. Therefore one has to switch to lower codimension strata to find feasible trajectory between two points of the bottom stratum. The switch changes in a discontinuous manner the equations of motions, situation which is not treated by conventional motion planning algorithms working on smooth configuration spaces. Thus these algorithms must be adapted (extended) to work with stratified systems.

A general motion planning algorithm, proposed by Lafferriere et al. [5] is used in [3] to solve the MPP for stratified systems. This algorithm uses piecewise constant inputs but it is imprecise if the Lie algebra generated by the vector fields g_i fails to be nilpotent. (Recall that in this case the Campbell-Baker-Hausdorff formula can be used to achieve any desired accuracy.) Notice also that the plannified trajectory may be composed from a prohibitively huge number of pieces and thus difficult to be realized in practice. Finally, let us note also that the algorithm does not give precise information about the actual trajectory of the system which can be nevertheless calculated by integrating the state equation with the plannified input sequence.

Motion planning can be easily solved for a restricted class of kinematic systems which are called differentially flat [1, 2]. For such systems, the MPP is reduced to a simple interpolation problem in the space of the flat output where the set of feasible trajectories is unconstrained in the sense that there is a one-to-one correspondence between sufficiently smooth trajectories of the flat output and feasible trajectories of the system.

We adapt this latter method to be used for stratified systems. Thus we construct trajectories made of pieces along which all but one input is zero. This can be done if we connect the initial and final points in the space of the flat output using the images of the flows associated to the control vector fields g_i in this space. Then the motion planning is reduced to a geometric problem, provided that the flows can be easily obtained. (This can be made off-line.) Due to the fact that we have direct influence on the geometry of the trajectory, obstacle avoidance can be also treated explicitly.

The application to the above methods is illustrated and compared using the example of the hexapod (six-legged) robot.

[5] Research partially supported by the Hungarian National Research Found under grant OTKA T 029072 and FKFP 0417/1997

References

[1] M. Fliess, J. Lévine, Ph. Martin, and P. Rouchon. Flatness and defect of nonlinear systems: introductory theory and examples. *Int. J. Control*, 61(6):1327–1361, 1995.

[2] M. Fliess, J. Lévine, Ph. Martin, and P. Rouchon. A lie bäcklund approach to equivalence and flatness of nonlinear systems. *IEEE Transactions on Automatic Control*, 38:700–716, 1999.

[3] B. Goodwine. *Control of stratified systems with robotic applications*. PhD thesis, California Institute of Technology, 1998.

[4] S.D. Kelly and R.M. Murray. Geometric phases and robotic locomotion. *J. Robotic Systems*, 12(6):417–431, 1995.

[5] G. Lafferriere and H.J. Sussmann. Motion planning for controllable systems without drift. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1148–1153, Sacramento, CA, USA, 1991.

Uniform distribution of linear recurrence sequences in residue class systems

Tamás Herendi

Random number sequences play an important role in numerical computations. In general, uniformly distributed sequences are the most useful in different applications. It is well known that linear recurrence sequences can be used for generating pseudorandom number sequences.

Let R be a Dedekind-domain, u be a linear rekurrence sequence and let P be a prime ideal in R. The sequence u is periodic, if we reduce modulo an arbitrary ideal I of R. We will say that u is uniformly distributed modulo I, if every residue class appear with the same frequency in a full period of u modulo I.

Several result deal with the conditions on parameters of linear recurring sequences which are uniformly distributed.

Our main result gives a possibility to determine linear recurrence sequences with arbitrarily big period length in arbitrarily big residue systems. The last property is important if we want to transform our sequence into the $[0, 1]$ interval. The bigger the residue system is the smoother the transformed sequence.

The precise formulation of the result is the following:

Theorem 1 *Let $P \in R$ be a prime ideal with $\pi \in \mathbb{Z}$ prime norm, $d \geq 2$ an integer, u a dth order linear recurring sequence in R and $S = \frac{3d^2+9d}{2} + 1$. If u is uniformly distributed $\pmod{P}^S$ then also uniformly distributed $\pmod{P}^s$ for any $s \in \mathbb{N}$.*

Efficient CAC Algorithms Based on the Tail Distribution of Aggregate Traffic

Zalán Heszberger, József Bíró and János Zátonyi

To guarantee the quality of services of today's communication networks together with efficient resource utilization requires well designed traffic management algorithms. Such algorithm is the call admission control (CAC), which makes decision about the acceptance of newcomer flows to a communication link. The operation of CAC algorithms often based on only very limited knowledge on the traffic for two main reasons: Firstly, the acceptance decision requires information on the newcomer flow in advance which is especially difficult e.g. in the case of interactive services. Secondly, even when the real characteristics of a source theoretically was available, in would be too complex (computationally expensive and time consuming) to determine or to utilize either. Characterizing the traffic is getting worse in the case of the sum of many traffic sources (or getting simpler?). This paper deals with performance measures for aggregate stream traffic which can be used for admission control. We consider traffic as fluids flowing into the link as pipes, and we assume that the emission rates of sources can be described by stationary rate processes. The widely-accepted and well-understood rate envelope (bufferless) multiplexing scheme is adopted for the traffic aggregation. This approach can provide low delay and delay jitter while the analysis of loss performance remains tractable.

In the paper, after introducing the tail distribution of aggregate traffic, making use of the well known Chernoff bounding method, we present techniques to estimate (upperbound) it, and use it in designing good performance measures for CAC. From the previous results we derive several simpler (thus less accurate) estimations. After the comparison of these techniques, their advantageous properties are also performed and conclusions are drawn in connection with their usefulness.

Effective Virus Scanning Algorithms [6]

Zoltán Hornák and Endre Selényi

Large number of computer viruses and their world-wide spreading is an interesting but very harmful factor in today's personal computer systems. Most anti-viral techniques used today are based on virus searching methods, that is they can detect and identify only those known viruses which were previously caught by anti-viral specialists and the specific virus detection codes were added to the anti-virus software. There are many methods for detecting and identifying viruses. Among these methods virus scanning is the mostly used and the most effective in practice.

Because of the large number of computer viruses and the continuously increasing size of the area which should be scanned, only the most effective scanning algorithms can be applied in the practice. On gigabyte-sized hard disks scanning of some ten thousand viruses with a brute force algorithm would take more than 1013 comparisons, which cannot be performed on a PC in reasonable time.

In contract with the traditional algorithm theory, where the behaviour of algorithms is analysed in the infinite, the discussed problem deals with the fastest solution of a specific, limited task. The best algorithm found uses the techniques of precondition, hashing and also the consideration of statistical probabilities. Although the resulting techniques are specific for virus scanning they can be used in other scanning problems as well.

The problem of virus scanning is defined as a search for occurrences of sequences in an area. A sequence is a string of bytes extended by special so called joker characters including ?, which matches any byte, +(number), which matches any (number) bytes, *(number), which matches any 0 to (number) bytes in length and [<alternative 1>, <alternative2>], which matches <alternative1> or <alternative2> sequences respectively.

For an effective search for occurrences of many sequences some candidate solutions were examined from speed to memory requirement points of view. All of the proposed solutions were based on parallel search, and used hashing technique. Hashing was combined with the optimised step-back precondition technique to improve speed, that is after a byte-stream was checked the processing did not return to the beginning of the byte-stream, but to an optimal position where an occurrence of a sequence was possible. This solution was proved to be very fast, but required a huge amount of memory.

Possible improvement to the above hashing algorithms is the reordering of the sequences to make the optimal step-back technique more effective. It turned out that the reordering of sequences is most effective if we also consider the statistical probabilities of byte and word occurrences in the scanned area.

Finally a mathematical model was developed to formalise the requirements toward an optimal sequence reordering. Experiences of the implementation of the algorithm showed that with realistic virus sequences this solution is so effective that the average number of comparisons is somewhere around the one sixth of the length of the scanned area, which is significantly better than the results of single hashing algorithms.

[6]This research was funded by the Hungarian Scientific Research Fund (F026251)

GPRS Function Test using TTCN

Endre Horváth and Axel Manthey

The introduction of General Packet Radio Services brought with it a challenge to carry out functional and conformance tests. Never before have a large number of telecom and datacom protocols been combined in a single network element. Demands to the test system were high, as the system should be able to grow with the implementation, support a large variety of protocols in different national variants and be utilized in simulated and target environment.

General Packet Radio Services (GPRS) is a standardized extension to existing GSM networks that offers packet switched data services. Two new network elements will be added to the GSM network architecture: the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN). These nodes are interconnected by means of an IP based core network and have signaling connections to existing GSM network elements such as Home Location Registers (HLR), Mobile Services Switching Center/Visitor Location Registers (MSC/VLR), Base Station Controllers (BSC) or Short Message Service Gateway MSCs and Interworking MSCs (SMS-GMSC, SMS-IWMSC). The SGSN serves packet data users in a defined geographical area while the GGSN connects to external packet data networks. From an end user's perspective GPRS offers permanent connectivity to IP networks, volume based charging and a higher bandwidth compared to existing GSM data services up to 115 Kbps. Circuit switched GSM services and packet switched GPRS services can coexist without disturbances, only one HLR based subscription is needed. Radio resources can be shared efficiently among several users. Horizontal applications (e.g. e-mail, FTP, HTTP) and vertical applications (such as telemetry, diagnostics, vending machines) can be offered. In addition SMS will be supported over GPRS.

TTCN (Tree and Tabular Combined Notation) is an internationally standardized test notation designed for testing of protocol implementations based on the OSI Reference Model. It can be used for specifying different types of tests used during different project phases and supports good logical structuring of test objects and good support for complex data structures including usage of ASN.1 (Abstract Syntax Notation One).

There are many protocols and interfaces in the GPRS network. Different protocols are used on different interfaces and all nodes interfacing the SGSN/GGSN have to be simulated, therefore testing a GPRS support node requires a complex test configuration simulating the other nodes adjacent to this entity. These expectations can be met by using *concurrent TTCN*. Not all the interfaces are available from the first design increment on: The test system is becoming more and more sophisticated as the design grows and even the protocols may change during the design due to standardization changes. *Modular TTCN* provides the possibility to upgrade the test suites easily, handles large sized test suite production and facilitates parallel work. The same test cases shall be executable in simulated and target environment as well as for different design increments that need to be maintained. This emphasizes the need to enable or disable different functionality. Testers can write easily adaptable test cases for different use cases by parameterized *TTCN test suites*.

TTCN test suites are independent of test methods, layers, protocols or test tools, so TTCN is applicable under many conditions. It enables the handling of different protocols and multiple interfaces in one test suite and supports testing of complicated systems like the GPRS support nodes combining telecom and datacom protocols. TTCN provides powerful means of modularization and parameterization of test suites.

A total of approximately 2500 test cases distributed over ca. 50 test objects has been produced and executed during the GSN verification project using TTCN. A large subset of these test cases has been repeated in all design increments (regression test) and helped to secure the product quality with a minimum of test case redesign.

Parallel Functional Reactive Skeletons in Concurrent Clean

Zoltán Horváth, Viktória Zsók, Pascal Serrarens and Rinus Plasmeijer

The skeletons are parameterised algorithmic schemes. We can use them to control parallel execution of the programs. There are two possibilities for integration of the skeletons in concurrent programming: one is to nest the skeletons in compiler and so the users are no aware of parallelism, and the other one is to write implementation modules that can be used and changed by users.

Skeletons in functional programming languages are expressed as higher order functions and they allow to implement the well-known parallel programming paradigm in portable, efficient programs [1]. The skeletons can be parameterised triply: by a function that computes the value of the result, by a strategy determining the dynamic behaviour and by types. Reactive skeletons are models of the reactive systems, which have some interactions with their environment, but doesn't give a final result. Usually they are non-deterministic.

In this paper we present a set of algorithmic skeletons in the parallel lazy functional language Concurrent Clean [2, 3]. Concurrent Clean provides process annotations for explicit thread creation and data-driven message passing system for the communication between processes. We would like to present the reactive skeletons with typical examples of concurrent systems.

References

[1] Galán, L.A.; Pareja, C.; Peña, R.: Functional Skeletons Generate Process Topologies in Eden. In *Int. Symp. on Programming Languages: Implementations, Logics and Programs, PLILP'96.* Aachen, Germany, pp. 289-303. Springer-Verlag LNCS 1140, 1996.

[2] Serrarens, P.R.: Explicit Message Passing for Concurrent Clean. In K. Hammond et al.: *Proc. of Implementation of Functional Languages, IFL'98*, pp. 298-308, Sept. 1998, London, to appear in LNCS.

[3] Kesseler, M. H. G.: *The Implementation of Functional Languages on Parallel Machines with Distributed Memory.* PhD thesis, University of Nijmegen, Constructing skeletons, pp. 153-173, 1996.

Discrete Simulation of Distributed Systems — Performance Evaluation of a Notification Channel Federation

József Hosszú

Event notification is essential in network management. Many components in a distributed network management system need an event notification mechanism, and so an event notification framework is expected to facilitate the development of the components. CORBA (Common Object Request Broker Architecture [OMG98]) provides an object-oriented platform to build such a framework, but its performance is not always sufficient for monitoring large-scale networks. The need for higher performance is rather necessary when message filters are applied in network elements.

CORBA is a distributed object middleware standard, on which client objects invoke operations on server objects in a location transparent fashion. CORBA provides various services named CORBAservices that are useful in developing distributed applications [OMG95]. The CORBA Event Service included in the CORBAservices can be used to implement event notification in CORBA-based network management systems. [Tom99] presents an event notification framework based on Java and CORBA providing a solution for high performance implementation of such a service, and meet very good results.

Performance evaluation of CORBA-based systems usually applies benchmarks. Tests are carried out using remote method calls and data transfer. [Bösz99] introduces a portable benchmark toolkit and reports measurements on a number of actual middleware products.

This paper presents how discrete simulation can be used for performance evaluation of distributed systems. With this methodology it is not needed to implement the system itself, only a model of proper specification is required. Simulation models for distributed systems can be easily adopted from other models, which are already used in network simulations with good results. The tool that supports our measurements is a powerful telecom simulation platform, a simulations development environment that supports object-oriented programming. The model used for demonstration represents a notification channel federation including an arbitrary number of event suppliers and event consumers connected to a scalable network. Performance and run-time behaviour are evaluated for various configurations, and results are presented.

References

[1] L. Böszörményi, A. Wickner, H. Wolf. Performance Evaluation of Object Oriented Middleware - Development of a Benchmarking Toolkit, Euro-Par'99, 5th International Euro-Par Conference, Toulouse, France, 1999.

[2] Object Management Group. CORBAservices: Common Object Services Specification, Revised Edition, Mar. 1995.

[3] Object Management Group. The Common Object Request Broker: Architecture and Specification, Revision 2.2, Feb. 1998.

[4] M. Tomono. An Event Notification Framework based on Java and CORBA, IM '99, Sixth IFIP/IEEE International Symposium on Integrated Network Management, Boston, MA, USA, May 1999.

Dynamic analysis of UML Statemachines

Gábor Huszerl

My main research topic is the dynamic analysis of Unified Modeling Language (UML) models for evaluation of performance, performability and timing properties. Instead of writing new analysis tools for the examination of the UML models, I am exploring the possibility of utilizing the proven ones. For performance evaluation and analysis Petri nets offer a mathematically well-defined methodology with precise semantics and a theoretical background. Petri nets are used in specification, design, verification and analysis of concurrent distributed systems for over 25 years, and there are already many Petri net tools known. I have defined a transformation of UML statemachines to Stochastic Reward Nets (SRNs), a special class of Petri nets, in special consideration of event processing, state hierarchy and priority of transitions. The transformation is defined by a set of design patterns. In order to support timing analysis I have extended the original UML models by including timing information [1, 2], and defined Petri net (SRN) patterns describing common semantics of timed and guarded state transitions. For analysing the arising SRNs well-known Petri net tools are used.

When transforming between different formalisms it is necessary to study the equivalence of the semantics of the source and target models. This paper examines a single design pattern of the above mentioned ones. This pattern implements the step semantics of the UML statemachines using SRNs.

UML [3] statemachines are hierarchical, where states can contain substates or concurrent submachines. Transitions may have their source and target states at different levels of the hierarchy. The transitions are triggered by events, and guarded by logical expressions. When several transitions are enabled, the maximal non-conflicting set of them may fire at the same time in a single step. Each step consists of the following hypothetical phases: dispatching an event, collecting the triggered and enabled transitions, selecting an appropriate subset of them and finally firing the selected transitions simultaneously.

SRNs have no hierarchy of places, and their transitions fire independently one at a time, thus the priority of transitions needs extra constructions in the patterns. These constructions are tree structures for every event type, determining which set of SRN transitions –representing individual UML transitions– may fire in the given step.

The SRN resulted by the transformation implements the phases of an above mentioned step. During the steps, there are intermediate states of the SRN, which have no valid counterpart in the statechart, however these states are transient ones vanishing before completion of the step.

This paper discusses the design patterns mapping the UML transition semantics to SRNs, and it focuses on the equivalence of the UML semantics of transition steps and the defined SRN structure. The proven equivalence of the UML model and the model directly analysed provides the designer precise information about performance, performability and timing properties of his current design.

References

[1] M. Dal Cin, Huszerl G., K. Kosmidis: *Transformation of Guarded Statecharts for Quantitative Evaluation of Dependable Embedded systems* - EWDC-10, ISBN 3-85403-125-4, pp. 143-187, Vienna, Austria, May 6-7, 1999

[2] M. Dal Cin, Huszerl G., K. Kosmidis: *Quantitative Evaluation of Dependability Critical Systems Based on Guarded Statechart Models* – In Proc. HASE'99, Washington DC, USA, 1999. November, pp. 37-45.

[3] *"UML Semantics", version 1.1* – Object Management Group, September 1997

On a two class on-line classification problem

Csanád Imreh

In an m*-class classification* problem, there are m classes and a number of different tasks arrive. The decision maker has to assign each task to exactly one of the m classes. This classification carries some cost, and we are to find a classification with minimum cost. In the on-line version we have to assign the tasks to a class immediately at their arrival and without any knowledge on the following tasks. This is a very general model which includes many load balancing and scheduling models from the literature.

In this work we present a mathematical model of the on -line classification problem, and we examine a special on-line two-class classification problem: we consider an on-line classification where the costs result from an off-line scheduling problem. We have two sets $\mathcal{P}$ and $\mathcal{S}$ of identical machines. The jobs arrive one by one. Each job j has two different processing times p_j and s_j, one for each set of machines. We have to decide in an on-line way on which set of machines to schedule each job. Finally, when the stream of jobs has come to an end, the constructed classification is evaluated in the following way: Schedule the jobs assigned to set $\mathcal{P}$ (respectively, the jobs assigned to set $\mathcal{S}$) on the machines of $\mathcal{P}$ (respectively, $\mathcal{S}$) so as to minimize the preemptive makespan, *i.e.* the maximal job completion time. Let C_P (respectively, C_S) denote this optimal makespan. Then the cost of the constructed classification is the maximum of the two makespans. This problem can be considered as a generalised version of the on-line scheduling problem.

We examine a greedy type algorithm for this problem, and we also present and examine an another algorithm which gives better result for the general case. As usually in the theory of on -line algorithms we measure the performance of an on-line algorithm by its competitive ratio. An on-line algorithm is called *c-competitive* if for each input the cost of the assignment produced by the algorithm is at most c times larger than the cost of the optimal assignment.

Strategy Selection in the Resolution Theory

Gusztáv Jánvári

The essay would like to deal with some interesting questions of the Resolution Theory in the field of Artificial Intelligence. These issues and the relevant answers try to find solutions for some efficiency problems of the automated reasoning, with discussing and processing some parts of the resolution theory and some very related fields. The main goal is to construct an algorithm, which would be able to always advise a given method for reasoning a given formula such that approximately best performance would be reached. In the followings I will give you a short overview about this job.

To give you the reasons why to concentrate on these topics, we must assume that we (or our theorem prover) have to prove very long logical formulas. Of course, the length is determined by the huge number of (different) predicate and variable symbols occurring in the formula, rather than the length of the symbols. It is a well known fact, that unification problem and theorem prover algorithms are more and more long in time by increasing the number of predicates and variables, and that certain growth is greater than linear. But, however, the time largely depends on the chosen kind of resolution method and strategy, too. So it would be worth to find a linear or an approximately linear algorithm, which would find out the most efficient strategy and method for a given problem. Anyway, a prover software has to verify a given formula by all means in order to seek after its structure and to make a syntactical verification (so it has to "run over" on the formula once at least), and our investigations may be inserted into this processing.

Analyzing resolution methods and strategies ("methods") is a very important part of this job. Why? Because we have to find some certain distinctive features of the syntactical structure of logical formulas that make those big differences between the efficiency of the various kind of methods. Afterwards we have to trace the methods apiece in order to be able to determine what kinds of formulas are required for a given method to reach good efficiency. The centre of this part should be to find special and usable measurement units (e.g. the deviation of the length of the longest clause from the average length of clauses).

After deriving some measurement units and giving a short summarize of our results, we will have to build up an algorithm doing the previous analysis alone. The goal is to increase the efficiency of proving, so this algorithm must also be efficient. And, of course, would be adaptable into other logical systems.

Measuring hardness of problems in evolutionary computations using Markov-processes

Márk Jelasity, Boglárka Tóth and Tamás Vinkó

In the recent years it has been clearified that evolutionary computations (EC) works very effective on real-life problems. However, the theory of EC is far from clear, finding a good characterization of problem difficulty is one of the main lines of EC theory research. So the question is, which problems are hard/easy to solve with the algorithms of EC. There are many suggested measure, for example epistasis variance and fitness-distance correlation (FDC). Epistasis variance is not so reliable and not so easy to compute. The FDC is much better, but there are some counterexample, which missleading this measure too. In this paper a new measure is suggested based on statistical properties of trajectories. These properties are approximated with the help of a heuristic based on transition probabilities between the elements of the search space, which derived from the Markov-processes. The basic idea is to examine the trajectories of the space with respect to a given operator and stopping criterion. The ending points of these trajectories form a very interesting set: these are the points the search is expected to converge. Based on the transition probabilities it is also possible to approximate the expected number of evaluations needed to get from a point to a given other point. This values can be used as distance measures and plots can be drawn that depict the convergence relations. With these definitions it is also possible to introduce a deceptiveness coefficient: a number from $[0, 1]$, which characterizes the problems: 0 indicates that the problem is misleading, 1 means that it is very friendly and there are transitions for other cases.

With our method any mutation or other genetic operator can be used, however only the mutation operator examined because the probabilities mentioned above need a huge amount of calculation. So this paper deal with relative small problems, but we don't model the fitness function explicitly, it is handled as a black box. Though the computational efforts are higher, the method is still feasible due to some techniques that speed up the convergence.

The layout of our paper is the following: after the introduction, in the section 2 some definitions and notions are given. In section 3 our method is demonstrated on four well-known problems from EC literature: ridge function, long path problem, a fully deceptive function and a combinatiorial problem: the subset sum problem. For the problems, three types of figures are introduced: iteration, deceptiveness and endpoint figures. In this section some explanation is given on how to read these figures. Finally our method is validated via some empirical results.

Predicting Exchange Rate by Neural Networks

Radmila Jovanovic and Ognjen Radovic

Predicting foreign exchange rates parity is a very important operation for financial managers, creditors and corporation tellers, so that in the future period it is to be certain preoccupation of scientific circles and financial institutions. The problem complexity did not make it possible to achieve satisfactory results by simple regressive analysis of time series, and that resulted in mass use of computers for all more complex models of exchange rates flows.

In this paper are given results of predicting foreign exchange rates in a short period of time by training pegged neural network over published exchange rates historical values. As examples, parities of the most significant world currencies are taken as values for learning the network. Predictions one day in advance are about 60%. Somewhat better results are got by previous data filtration using Kalman filters.

Possibilities of predicting exchange rates by neural networks practically enable bringing got results closer to real exchange rates flow. Successfulness in that extrapolishing is about 60%. Such results surely do not satisfy, but it is by improvements in neural network use that they are made better in this field. The most serious problem in the networks creation is finding out what factor is more important, namely estimating their specific weight in the given moment. In case the neural networks reach that level of perfection and predict exchange rates flow more precisely, there are always a lot of factors left which cannot be expressed quantitatively and where it is to start either with assumptions and low of profitability or free assumption concerning influence of given factor on observed phenomenon.

Predicting successfulness is to be one of the significant information for those who make decisions on buying or selling foreign currency, namely those who influence the structure of foreign trade and foreign debts in a certain way.

Sets of numbers in different number systems and the Chomsky hierarchy

István Katsányi

It is a thoroughly studied subject within the discipline of formal languages and automata theory, that under which conditions will a set of numbers in m–ary notation be regular for a given $m \geq 1$. Cobham has solved the bases of this problem in [1]. His results were extended and generalized by many authors in many ways for example in the papers [2], [3], [4], [5], [6], [7]. Luca and Restivo suggested in their paper ([3]) to study the open problem of the context–free case. In this work, we examine the context–free, the context sensitive and the recursively enumerable classes in addition to the regular languages, hence the examination of the Chomsky–hierarchy in this regard becomes complete.

Let $\mathcal{N}$ denote the set of nonnegative integers, $\mathcal{REG}$, $\mathcal{CF}$, $\mathcal{CS}$, $\mathcal{RE}$ the classes of regular, context–free, context sensitive and recursively enumerable languages, respectively. For a set $A \subseteq \mathcal{N}$, and for an integer $a \geq 1$ let us denote by $L_a(A)$ the language that represents the set A in the base a number system.

One of the main results of our paper is the next theorem:

Theorem 1 *The following table is filled in correctly.*

	$a = 1, b \geq 2$	$a \geq 2, b = 1$	$a, b \geq 2,$ $\exists n, m \geq 1:$ $a^n = b^m$	$a, b \geq 2,$ $\nexists n, m \geq 1:$ $a^n = b^m$
$\mathcal{REG}$	$\mathcal{REG}$	$\mathcal{CS}$	$\mathcal{REG}$	$\mathcal{CS}^*$
$\mathcal{CF}$	$\mathcal{REG}$	$\mathcal{CS}$	$\mathcal{CF}$	$\mathcal{CS}^*$
$\mathcal{CS}$	$\mathcal{RE}^*$	$\mathcal{CS}$	$\mathcal{CS}$	$\mathcal{CS}$
$\mathcal{RE}$	$\mathcal{RE}$	$\mathcal{RE}$	$\mathcal{RE}$	$\mathcal{RE}$

Each element of the table determines the Chomsky–class, that for every set $A \subseteq \mathcal{N}$ the language $L_b(A)$ belongs to, provided that a and b have the property written in the heading of the column of the element, and $L_a(A)$ belongs to the class shown in the heading of the row of the element. With the exception of the elements marked with a $*$, the presented classes are the smallest ones in the Chomsky–hierarchy with this property.

The other main part of this paper is to study, that when do the arithmetical operations alter the Chomsky–class of the sets represented in a number system. First define some operation over sets of numbers:

Definition 1 *Let $A, B \subseteq \mathcal{N}$ be two sets of numbers, and let $c \geq 0$ be an integer. Let us define*

$$A + B = \{a + b \mid a \in A, b \in B\}, \qquad A \cdot B = \{ab \mid a \in A, b \in B\},$$
$$A^B = \{a^b \mid a \in A, b \in B\},$$
$$c + A = A + c = \{c\} + A, \qquad c \cdot A = A \cdot c = \{c\} \cdot A,$$
$$c^A = \{c\}^A, \qquad A^c = A^{\{c\}}.$$

Theorem 2 *For every base $a \geq 1$, the following table is filled in correctly.*

	$c + A$	$c \cdot A$	$A + B$	$A \cdot B$	A^B
$\mathcal{REG}$	$\mathcal{REG}$	$\mathcal{REG}$	$\mathcal{REG}$	$\mathcal{CS}^*$	$\mathcal{CS}^*$
$\mathcal{CF}$	$\mathcal{CF}$	$\mathcal{CF}$	$\mathcal{CS}^*$	$\mathcal{CS}^*$	$\mathcal{CS}^*$
$\mathcal{CS}$	$\mathcal{CS}$	$\mathcal{CS}$	$\mathcal{CS}$	$\mathcal{CS}$	$\mathcal{CS}$
$\mathcal{RE}$	$\mathcal{RE}$	$\mathcal{RE}$	$\mathcal{RE}$	$\mathcal{RE}$	$\mathcal{RE}$

Each element of the table determines the Chomsky–class, that the language $L_a(C)$ belongs to for every $c \geq 0$, $A, B \subseteq \mathcal{N}$, provided that C is the result of the operation written in the heading of the column of the element, and $L_a(A)$ (and $L_a(B)$, if appropriate) belongs to the class shown in the heading of the row of the element. With the exception of the elements marked with a $*$, the presented classes are the smallest ones in the Chomsky–hierarchy with this property.

As a corollary of theorem 2, we get results, which in some sense extend the theorem of Horváth about the ranges of polinomials published in [8]:

Theorem 3 *Let $a \geq 1$ be the base of our number system, $A_0, A_1 \subseteq \mathcal{N}$ be finite sets, $X \subseteq \mathcal{N}$ be a set for which $L_a(X) \in \mathcal{F}$, where $\mathcal{F}$ is one of the classes $\mathcal{RE}, \mathcal{CS}, \mathcal{CF}, \mathcal{REG}$. Then $L_a(A_1 \cdot X + A_0) \in \mathcal{F}$.*

Theorem 4 *Let $a \geq 1$ be the base of our number system, $A_0, A_1, \ldots A_n \subseteq \mathcal{N}$ be finite sets, $X \subseteq \mathcal{N}$ be a set for which $L_a(X) \in \mathcal{F}$, where $\mathcal{F}$ is one of the classes $\mathcal{RE}, \mathcal{CS}$. Then $L_a(A_n X^n + \cdots + A_1 \cdot X + A_0) \in \mathcal{F}$.*

Acknowledgement
The author wishes to thank László Hunyadvári for his useful suggestions.

References

[1] Alan Cobham. On the base-dependance of sets of numbers recognizable by finite automata. *Mathematical Systems Theory*, 3(2):186–192, 1969.

[2] Karel Culik II and Arto Salomaa. Ambiguity and decision problems concerning number systems. *Information and Control*, 56(3):139–153, 1983.

[3] Aldo de Luca and Antonio Restivo. Star-free sets of integers. *Theoretical Computer Science*, 43(2-3):265–275, 1986.

[4] Juha Honkala. Bases and ambiguity of number systems. *Theoretical Computer Science*, 31(1-2):61–71, May 1984.

[5] Juha Honkala. A decision method for the recognizability of sets defined by number systems. *RAIRO Inform. Théor. Appl.*, 20(4):395–403, 1986.

[6] Juha Honkala. A decision method for the unambiguity of sets defined by number systems. *The Journal of Universal Computer Science*, 1(9):648–653, September 1995.

[7] Christian Michaux and Roger Villemaire. Presburger arithmetic and recognizability of sets of natural numbers by automata: New proofs of Cobham's and Semenov's theorems. *Annals of Pure and Applied Logic*, 77(3):251–277, 19 February 1996.

[8] Horváth Sándor. Ranges of polinomials are non–context–free in any number system. To appear in: Pure Mathematics and Applications.

Mobile Technologies and Future Health-care

László Keresztfalvi

Mobile technology development and investments are on the leading edge beside the Internet. Technical solutions satisfy comfortably more and more complex needs. Also the huge public interest makes the end products cheaper and cheaper.

Modern health-care gets more important worldwide. Affecting everybody these systems tend to be one of the largests. Unfortunately the required technology and the health-care itself is also very expensive without showing direct financial income. After all health-care gets lots of investments (mostly by state sponsorship) to build and modernize its information technology background in order to shorten service time, increase patient satisfaction and of course cut down charges.

Combining these opportunities modern technology can extend hospital care allowing home monitoring which can help to utilize more efficiently the hospital capacities, instant practitioner support, online patient data querying and much more prospects.

In this article I show the basic ideas of some mobile communication standards (e.g. Bluetooth, WAP) which are currently being developed and their applicability in health-care with some real-life examples. I introduce the solutions and some problems of an already set up cardiological (ECG) home care center. The patient data exchange seems to be a general problem for health-care systems so finally I flash some chance to take over this problem.

SIP or H.323: Which Call Control Protocol is More Suitable for the 3G UMTS All-IP Network?

Krisztián Kiss

The today's telephony service is provided mostly over circuit-switched networks, which are referred to as Public Switched Telephone Network (PSTN). In recent years a new trend is beginning to emerge: the telephony service over IP networks, which results the integration of voice and data applications. However, for operation of an IP telephony architecture it is needed to provide a signaling infrastructure, which offers at least the same capabilities and features as the Signaling System (SS7) architecture in PSTN.

The purpose of a multimedia signaling protocol is to enable two-way communication between two or more endpoints. Currently, there are two major candidates for packet-based communication being specified, the H.323 protocol suite from International Telecommunication Union (ITU-T) and the Session Initiation Protocol (SIP) from the Internet Engineering Task Force (IETF). Both protocols define methods how the endpoint can register to a multimedia network, initiate call setup procedure, exchange capability information, and establish, modify and release calls. Both protocols also support multiparty conferencing and supplementary services. Neither SIP nor H.323 has been originally designed for a mobile IP network, and thus architecture supporting the mobility of the endpoints imposes some new requirements on the protocols. The proposal from 3GPP for the Release 2000 UMTS standard [1] has been developed with the goal of allowing operators to deploy an All-IP based architecture to deliver 3rd Generation wireless services. The architecture, which contains the following key segments: radio network, GPRS (General Packet Radio Service) network, call control, gateways to external network, and service architecture, has been kept generic and is not based on a specific call control mechanism such as H.323 or SIP.

H.323 [2] is a standard that specifies the components, protocols and procedures that provide multimedia communication services: real-time audio, video, and data communications over packet networks including IP based networks. The H.323 standard specifies four kinds of components: terminal, gateway, gatekeeper and Multipoint Control Unit (MCU). H.323 system control is provided by three separate signaling functions: the H.245 Control Channel [3], the Q.931 Call Signaling Channel [4], and the RAS (Registration, Admission and Status) Channel [4]. The RAS signaling function uses H.225.0 [4] messages to perform registration, admissions, bandwidths changes, status, and disengage procedures between endpoints and Gatekeepers. The call signaling function uses H.225.0 call signaling to establish a connection between two H.323 endpoints. The H.245 Control Function uses the H.245 Control Channel to carry end-to-end control messages governing operation of the H.323 entity, including capabilities exchange, opening and closing of logical channels, mode preference requests, flow control messages, and general commands and indications.

SIP [5] is an application-layer signaling protocol, which is used to establish and control multimedia sessions or calls, both unicast and multicast. SIP specifies basically two kinds of components: terminals (user agents) and SIP network elements, which can be proxy servers, redirect servers, location servers or registrars. In SIP the gateways are considered as special cases of user agents. The SIP protocol includes basic call signaling, user location, registration and as an extension also advanced signaling. The other services, such as quality of service, directory access, service discovery, session content description and conference control, are orthogonal and reside in separate protocols. SIP has a modular architecture, where different functions are performed in different protocols. Protocols can easily be replaced, and even components of H.323 can be integrated into the SIP environment. SIP uses the Session Description Protocol (SDP) [6] to describe the capabilities and media types supported by the terminals. SIP is independent of the environment, and does not require any reliable transport protocol. In fact any datagram or stream protocol that delivers a whole SIP request or response in full can be used. Such protocols are UDP and TCP in the Internet.

H.323 is an umbrella standard actually covering many protocols, all of which are needed for H.323 multimedia services. H.323 uses binary coding and ASN.1, making stack implementation a big effort. SIP, on the other hand, being lightweight and text-based, is easier to implement and is supported mainly by the Internet community. The complexity of the implementation may be an issue in the terminal, but not so much in the network. In the network, the most important issues are reliability, support for future extensions, and performance. SIP has been said to be more scalable because the SIP call processing functionality implementation can be stateless, thus requiring less memory in the network element. However, new requirements set by the mobile IP telephony network, like charging, statistics, and IN triggering would either require dedicated servers for these purposes, or a stateful implementation, thus resulting in a similar memory consumption than with H.323.

At the moment, neither of the protocols supports the requirements imposed by mobile IP telephony. The basic functionality of the two protocols is quite similar, and with the modifications both could satisfy the requirements of a mobile IP telephony network. Also, should both protocols be selected as alternatives for UMTS Release 00, the interworking of the protocols is not foreseen to be a major obstacle.

References

[1] 3rd Generation Partnership Project, Technical Specification Group Services and System Aspects, Architecture for an All-IP network, 3G TR 23.922 version 1.0.0, October 1999

[2] ITU-T Rec. H.323: Packet-based multimedia communications systems, Ver. 02/98

[3] ITU-T Rec. H.245: Control Protocol for multimedia communication, Ver. 09/98

[4] ITU-T Rec. H.225.0: Call signaling protocols and media stream packetization for packet-based multimedia communication systems, Ver. 02/98

[5] Handley, M., Schulzrinne, H., Schooler, E., Rosenberg, J.: SIP: Session Initiation Protocol, IETF RFC 2543, March 1999

[6] Handley, M., Jacobson, V.: SDP: Session Description Protocol, IETF RFC 2327, April 1998

New Clustering Procedures in Respect of Practical Realization

Koch György and József Dombi

Clustering is one of the most frequently used statistical analyzing method. Its aim is to divide a set $A = \{x_1, x_2 ..., x_N\}$ of N objects into C categories called clusters in such a way that objects belonging to the same cluster should be as similar to each other as possible while objects in different clusters should be dissimilar.

Since there is no widely acceptable mathematical formula for similarity or dissimilarity, clustering problem has no universally usable optimal solution. The goodness of a clustering algorithm can only be tested empirically.

Depending on the given information there are two categories of clustering methods.

• If the parameters of the objects are given, the so-called coordinate based clustering algorithms can be used. These methods handle the objects like points in the m-dimensional space. Fastest algorithms in this group have $O(n)$ complexity which makes them suitable to process large databases. Well-known methods from this class are Kohonen Clustering Networks(KCN) and Fuzzy C- Means(FCM). We developed a new coordinate based clustering algorithm which has $O(n)$ complexity too and has several advantages over the previous methods. This algorithm is called Shepherd Method(SM) because of its behaviour. In Shepherd Method we iteratively separate the clusters with hyper planes (like a special Voronoi diagram) so it can be used easily for classification too. We also worked out a method for SM to automatically determine the number of clusters.

• Distance based clustering algorithms are the second group of clustering algorithms. In this case we do not know the exact parameters of the objects only their distances from each other or the similarity/dissimilarity between them. Hence this similarity is given by a matrix of size n*n, these algorithms have $O(n^2)$ time complexity, but these methods usually give better results than algorithms from the coordinate based group. One of the best distance based method is Chameleon which was published in 1999. We developed a new distance based method, which gives the same or even better results that Chameleon does. It is worth to mention that we achieved to handle relatively small amount of objects. Our method is called SmallSteps because it tries to find connected graphs which have egdes with a maximum weight which is computed on the environments of the objects. SmallSteps is capable to detect clusters with different shapes, sizes or densities and it is able to automatically determine the number of clusters needed and has the speciality to divide clusters into sub-clusters.

Phoneme Classification Using Kernel Principal Component Analysis

András Kocsor, László Tóth and András Kuba

1. Introduction

In an earlier paper[4] we compared the effect of the linear feature transformation methods Principal Component Analysis(*PCA*), Linear Discriminant Analysis(*LDA*) and Independent Component Analysis(*ICA*) on several learning algorithms. The algorithms compared were *TiMBL*(the *IB1* algorithm), *C4.5*(*ID3* tree learning), *OC1*(oblique tree learning), Artificial Neural Nets(*ANN*), Gaussian Mixture Modeling(*GMM*) and Hidden Markov Modeling(*HMM*). The domain of the comparison was phoneme classification using a certain segmental phoneme model, and each learner was tested with each transformation in order to find the best combination. Furthermore, in that paper we experimented with several feature sets such as filter bank energies, mel-frequency cepstral coefficients(*MFCC*) and gravity centers. This paper reports on our experiments to extend these investigations towards nonlinear methods. Namely, we show how the well-known Principal Component Analysis(*PCA*) can be non-linearized using the so-called "kernel-idea"[5]. Besides presenting the "Kernel-idea" we also give formulas both for the original *PCA* and the *Kernel-PCA*. In this paper we thoroughly examine how this nonlinear feature transformation effects the efficiency of several learning algorithms. As we mentioned previously in our earlier study we experimented with several feature sets, and we found the best one to be the critical band log-energies. So in this study we use only this quite traditional technique to extract frame-based features from the speech signal. We also learned from our previous investigations that from the learning algorithms *PCA*[2][6] was the most beneficial for *GMM*[1] and *ANN*[1]. Thus, in this paper we present classification results only for these two methods (apart from those of an *HMM* recognizer, which are given to serve as a reference point). Since the crucial point of this study is the *Kernel-PCA*, we give a brief look at it in this extended abstract.

2. Feature Space Transformation Methods with Kernels

Before executing a learning algorithm, additional vector space transformations may be applied on the extracted features. The role of these methods is twofold. Firstly they may improve classification performance, and secondly they may also reduce the dimensionality of the data. This is due to the fact that these techniques search for a transformation which emphasizes more important features and supresses or even eliminates less desirable ones.

2.1 Linear Feature Space Transformation Methods

Without loss of generality we will assume that the original data set lies in $\mathbb{R}^n$, and that we have l elements $\mathbf{x_1}, \ldots, \mathbf{x_l}$ in the training set and t elements $\mathbf{y_1}, \ldots, \mathbf{y_t}$ in the testing set. The feature transformation methods in many cases require certain preprocessing steps, which usually mean simple linear transformations. The results of this preprocessing step will be denoted by $\hat{\mathbf{x}}_\mathbf{1}, \ldots, \hat{\mathbf{x}}_\mathbf{l}$ and $\hat{\mathbf{y}}_\mathbf{1}, \ldots, \hat{\mathbf{y}}_\mathbf{t}$ for the training and testing vectors, respectively. After applying a feature space transformation method for the preprocessed data, the new data set lies in $\mathbb{R}^m \quad (m \leq n)$, the transformed training and testing vectors being denoted by $\mathbf{x}'_\mathbf{1}, \ldots, \mathbf{x}'_\mathbf{l}$ and $\mathbf{y}'_\mathbf{1}, \ldots, \mathbf{y}'_\mathbf{r}$ respectively. With the linear feature space transformation methods, we search for an optimal (in some cases orthogonal) linear transformation $\mathbb{R}^n \rightarrow \mathbb{R}^m$ of the form $\mathbf{x}'_\mathbf{i} = \mathbf{A}^\top \hat{\mathbf{x}}_\mathbf{i}, \quad i \in \{1, \ldots, l\}$, noting that the precise definition of optimality can vary from method to method. The column vectors $\mathbf{a_1}, \ldots, \mathbf{a_m}$ of the $n \times m$ matrix $\mathbf{A}$ supposed to be normalized. These algorithms use various objective functions $\tau() : \mathbb{R}^n \rightarrow \mathbb{R}$ which serves as a measure for selecting one optimal direction (i.e. a new base vector). Usually, linear feature space transformation methods search for m optimal directions. Although it is possible to define functions that measure the optimality of all the m directions *together*, we will find the directions of the optimal transformations *one-by-one*, employing the τ measure for each direction separately. One, but quite heuristic way of this is to look for unit vectors which form the stationary points of $\tau()$. Intuitively, if larger values of $\tau()$ indicate better directions and the chosen directions needs to be independent in some ways, then choosing stationary points that have large values is a reasonable strategy.

2.2. Kernel Transformation Methods

In this subsection the symbols $\mathcal{H}$ and $\mathcal{F}$ denote real vector spaces that might as well be finite of infinite in dimension. Also, we suppose to have a mapping $\Phi : \mathbb{R}^n \rightarrow \mathcal{H}$, which is not necessarily linear, and $\dim(\mathcal{H})$ is either finite or infinite. Furthermore, we suppose to have given an algorithm $\mathcal{P}$, with its input formed by preprocessed training points $\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l$ of the vector space $\mathbb{R}^n$. (In our case this algorithm is the *PCA*.) The output of the algorithm $\mathcal{P}$ is a linear transformation $\mathbb{R}^n \to \mathbb{R}^m$, where both the degree of the dimension reduction (represented by m) and the $n \times m$ transformation matrix $\mathbf{A}$ are determined by the algorithm itself. We will denote the transformation matrix $\mathbf{A}$ resulting for the training data by $\mathcal{P}(\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l)$.

The goal of the nonlinearization methods is to transform the training vectors into a point set in $\mathcal{H}$ by a mapping Φ, and instead of the original ones in $\mathbb{R}^n$, we apply the algorithm $\mathcal{P}$ on these transformed points in $\mathcal{H}$. Thus, employing the algorithm $\mathcal{P}$ on the input elements $\Phi(\hat{\mathbf{x}}_1), \ldots, \Phi(\hat{\mathbf{x}}_l) \in \mathcal{H}$ we gain a linear trasformation $\Psi : \mathcal{H} \to \mathcal{F}$. Similarly as before, we will denote the matrix of the resulting linear mapping Ψ with $\mathcal{P}(\Phi(\hat{\mathbf{x}}_1), \ldots, \Phi(\hat{\mathbf{x}}_l))$. Since Φ is not linear in general, the composite transformation $\Psi \circ \Phi$ of Φ and Ψ will not necessarily be linear either.

In the case of the Kernel transformation methods the algorithm $\mathcal{P}$ is turned into an equivalent algorithm $\mathcal{P}'$ for which the following holds: $\mathcal{P}(\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l) = \mathcal{P}'(\hat{\mathbf{x}}_1^\top \hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_i^\top \hat{\mathbf{x}}_j, \ldots, \hat{\mathbf{x}}_l^\top \hat{\mathbf{x}}_l)$, for arbitrary $\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l$. This will, of course, hold in $\mathcal{H}$ too, that is: $\mathcal{P}(\Phi(\hat{\mathbf{x}}_1), \ldots, \Phi(\hat{\mathbf{x}}_l)) = \mathcal{P}'(\Phi(\hat{\mathbf{x}}_1)^\top \Phi(\hat{\mathbf{x}}_1), \ldots, \Phi(\hat{\mathbf{x}}_i)^\top \Phi(\hat{\mathbf{x}}_j), \ldots, \Phi(\hat{\mathbf{x}}_l)^\top \Phi(\hat{\mathbf{x}}_l))$, for arbitrary $\Phi(\hat{\mathbf{x}}_1), \ldots, \Phi(\hat{\mathbf{x}}_l)$. Thus, the point of the kernel methods is to form an algorithm $\mathcal{P}'$, which is equivalent to $\mathcal{P}$, but its inputs are the dot products of the inputs of $\mathcal{P}$.

The complexity of the linear *PCA*($\mathcal{P}$) is a non-linear function of the dimensionality of the input vectors. Thus if $\dim(\mathcal{H})$ is much larger than n, then the corresponding $\mathcal{P}'$ algorithm may become practically infeasible. This problem can be alleviated if we have a low-complexity (for example linear) Kernel function $\kappa() : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ for which $\Phi(\mathbf{x})^\top \Phi(\mathbf{y}) = \kappa(\mathbf{x}, \mathbf{y}), \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. In this case the value of $\Phi(\hat{\mathbf{x}}_i)^\top \Phi(\hat{\mathbf{x}}_j)$ can also be computed with few (for example $O(n)$) operations, even if the dimension of $\Phi(\hat{\mathbf{x}}_i)$ and $\Phi(\hat{\mathbf{x}}_j)$ are infinite. In practice, however, we usually face the problem just the opposite way: given a $\kappa() : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ functional as Kernel, we are looking for a mapping Φ for which $\Phi(\mathbf{x})^\top \Phi(\mathbf{y}) = \kappa(\mathbf{x}, \mathbf{y}), \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. There are many good publications about the proper choice of the Kernel functions, and also about their theory in general[5].

In our studies we employed the following Kernels: $\kappa_1(\mathbf{x}, \mathbf{y}) = \left(\mathbf{x}^\top \mathbf{y}\right)^p, \ 0 < p \in \mathbb{R}$ and $\kappa_2(\mathbf{x}, \mathbf{y}) = \exp\left(-\mathbf{x} - \mathbf{y}^2/r\right), 0 < r \in \mathbb{R}$. Thus, after choosing a Kernel function the only thing left is to take the $\mathcal{P}'$ version of the algorithm and replace the input elements $\hat{\mathbf{x}}_1^\top \hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_i^\top \hat{\mathbf{x}}_j, \ldots, \hat{\mathbf{x}}_l^\top \hat{\mathbf{x}}_l$ with the elements $\kappa(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_1), \ldots, \kappa(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j), \ldots, \kappa(\hat{\mathbf{x}}_l, \hat{\mathbf{x}}_l)$. The algorithm that results from this substitution can perform the *PCA* transformation with practically acceptable complexity, even in a space infinite in dimension. This transformation together with a properly chosen Kernel function results in the non-linear feature space transformation, i.e. *Kernel-PCA*.

2.3. Steps of the Methods In the following sections the discussion of the methods *PCA* and *Kernel-PCA* will be decomposed into three steps:

- **Preprocessing Step** Describes the preprocessing that might be required by the method.
- **Transformation Step** Here we derive the algorithms themselves.
- **Transformation of Test Vectors** Here we discuss that, having obtained a transformation based on the training vectors, what kind of processing it implies on the test vectors.

3. Principal Component Analysis Preprocessing Step:

- **Centering:** We shift the original sample set $\mathbf{x}_1, \ldots, \mathbf{x}_l$ with its mean $\boldsymbol{\mu}$, to obtain a set $\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l$, with a mean of $\mathbf{0}$: $\quad \hat{\mathbf{x}}_1 = \mathbf{x}_1 - \boldsymbol{\mu}, \ldots, \hat{\mathbf{x}}_l = \mathbf{x}_l - \boldsymbol{\mu}, \qquad \boldsymbol{\mu} = l^{-1} \sum_{i=1}^{l} \mathbf{x}_i$.

Transformation Step:

Normally in *PCA* $\tau(\mathbf{a})$ is $\mathbf{a}^\top \mathbf{C}\mathbf{a} / \mathbf{a}^\top \mathbf{a}$ ($\mathbf{a} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$), where $\mathbf{C}$ is the sample covariance matrix for the standardized data ($\mathbf{C} = l^{-1} \sum_{i=1}^{l} \hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^\top$). Practically speaking, $\mathbf{a}^\top \mathbf{C}\mathbf{a} / \mathbf{a}^\top \mathbf{a}$ defines $\tau(\mathbf{a})$ as the variance of the $\{\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l\}$ n-dimensional point-set projected onto the vector $\mathbf{a}$. Therefore this method prefers directions having a large variance. It can be shown that stationary points of $\tau(\mathbf{a})$ correspond to the right eigenvectors of the sample covariance matrix $\mathbf{C}$ where the eigenvalues form the corresponding function values. Thus it is worth defining *PCA* based on the stationary points where the function $\tau()$ has dominant values. If we assume that the eigenpairs of $\mathbf{C}$ are $(\mathbf{c}_1, \lambda_1), \ldots, (\mathbf{c}_n, \lambda_n)$ and $\lambda_1 \geq \ldots \geq \lambda_n$, then the transformation matrix $\mathbf{A}$ will be $[\mathbf{c}_1, \ldots, \mathbf{c}_m]$, i.e. the eigenvectors with the largest m eigenvalues. Since the sample covariance matrix $\mathbf{C}$ is symmetric positive semidefinite, the eigenvectors are orthogonal and the corresponding real eigenvalues are nonnegative. After this orthogonal linear transformation the dimensionality of the data will be m. It is easy to check that the sample $\mathbf{x'}_i = \mathbf{A}^\top \hat{\mathbf{x}}_i, \ \ i \in \{1, \ldots, l\}$ represented in the new orthogonal basis will be uncorrelated, *i.e.* the covariance matrix $\mathbf{C}'$ of it is diagonal. The diagonal elements of $\mathbf{C}'$ are the m dominant eigenvalues of $\mathbf{C}$. In our experiments, m (the dimensionality of the transformed space) was chosen to be the smallest integer for which $(\lambda_1 + \ldots + \lambda_m)/(\lambda_1 + \ldots + \lambda_n) > 0.99$ holds. Note that there are many other alternatives, however, for finding a reasonable m.

Transformation of test vectors:

For an arbitrary test vector $\mathbf{y}$ the transformation is $\mathbf{y}' = \mathbf{A}^\top \hat{\mathbf{y}}$, where $\hat{\mathbf{y}}$ denotes the preporecessed $\mathbf{y}$.

4. Formulas for *Kernel-PCA* Having chosen a proper κ Kernel function for which $\kappa(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x})^\top \Phi(\mathbf{y}), \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ holds for a mapping $\Phi : \mathbb{R}^n \to \mathcal{H}$, we now give the *LDA* transformation in $\mathcal{H}$.

Preprocessing Step:

- **Kernel Centering:** We shift the data $\Phi(\mathbf{x_1}), \ldots, \Phi(\mathbf{x_l})$ with its mean $\boldsymbol{\mu}^\Phi$, to obtain a set $\hat{\Phi}(\mathbf{x_1}), \ldots, \hat{\Phi}(\mathbf{x_l})$ with a mean of $\mathbf{0}$: $\quad \hat{\Phi}(\mathbf{x_1}) = \Phi(\mathbf{x_1}) - \boldsymbol{\mu}^\Phi, \ldots, \hat{\Phi}(\mathbf{x_l}) = \Phi(\mathbf{x_l}) - \boldsymbol{\mu}^\Phi, \quad \boldsymbol{\mu}^\Phi = \frac{1}{l}\sum_{i=1}^{l} \Phi(\mathbf{x_i})$.

Transformation Step:

We employed the following measure in $\mathcal{H}: \quad \tau^{\hat{\Phi}}(\mathbf{a}) = \mathbf{a}^\top \mathbf{C}^{\hat{\Phi}} \mathbf{a} / \mathbf{a}^\top \mathbf{a}, \quad \mathbf{a} \in \mathcal{H} \setminus \{\mathbf{0}\}$, where $\mathbf{C}^{\hat{\Phi}}$ $(= l^{-1} \sum_{i=1}^{l} \hat{\Phi}(\mathbf{x_i})\hat{\Phi}(\mathbf{x_i})^\top)$ is the covariance matrix of the sample $\hat{\Phi}(\mathbf{x_1}), \ldots, \hat{\Phi}(\mathbf{x_l})$. Analogously to *PCA*, we define *Kernel-PCA* based on the stationary points of $\tau^{\hat{\Phi}}(\mathbf{a})$, which are given as the eigenvectors of the symmetric positive semidefinite matrix $\mathbf{C}^{\hat{\Phi}}$. Because of the special form of $\mathbf{C}^{\hat{\Phi}}$ we can suppose that $\mathbf{a} = \sum_{i=1}^{l} \alpha_i \hat{\Phi}(x_i)$. The following formulas give $\tau^{\hat{\Phi}}(\mathbf{a})$ as the function of α_t and $\kappa(\mathbf{x_i}, \mathbf{x_j})$

$$\tau^{\hat{\Phi}}(\mathbf{a}) = \frac{\mathbf{a}^\top \mathbf{C}^{\hat{\Phi}} \mathbf{a}}{\mathbf{a}^\top \mathbf{a}} = \frac{\left(\sum_{t=1}^{l} \alpha_t \hat{\Phi}(x_t)^\top\right) \mathbf{C}^{\hat{\Phi}} \left(\sum_{s=1}^{l} \alpha_s \hat{\Phi}(x_s)\right)}{\left(\sum_{t=1}^{l} \alpha_t \hat{\Phi}(x_t)^\top\right) \left(\sum_{s=1}^{l} \alpha_s \hat{\Phi}(x_s)\right)} = \frac{\boldsymbol{\alpha}^\top \frac{1}{l} \mathbf{K}^{\hat{\Phi}} \mathbf{K}^{\hat{\Phi}} \boldsymbol{\alpha}}{\boldsymbol{\alpha}^\top \mathbf{K}^{\hat{\Phi}} \boldsymbol{\alpha}}, \tag{3}$$

where[7] $\mathbf{K}^{\hat{\Phi}}{}_{ts} = \left(\Phi(\mathbf{x_t})^\top - \left(\frac{1}{l}\sum_{i=1}^{l} \Phi(\mathbf{x_i})^\top\right)\right) \left(\Phi(\mathbf{x_s}) - \left(\frac{1}{l}\sum_{i=1}^{l} \Phi(\mathbf{x_i})\right)\right) =$ $\kappa(\mathbf{x_t}, \mathbf{x_s}) - \left(\frac{1}{l}\sum_{i=1}^{l} \left(\kappa(\mathbf{x_i}, \mathbf{x_s}) + \kappa(\mathbf{x_t}, \mathbf{x_i})\right)\right) + \frac{1}{l^2}\sum_{i=1}^{l}\sum_{j=1}^{l} \kappa(\mathbf{x_i}, \mathbf{x_j})$. From differentiating $\tau^{\hat{\Phi}}()$ with respect to $\boldsymbol{\alpha}$ we get that the stationary points are the solution vectors of the general eigenvalue problem $\frac{1}{l}\mathbf{K}^{\hat{\Phi}}\mathbf{K}^{\hat{\Phi}}\boldsymbol{\alpha} = \lambda \mathbf{K}^{\hat{\Phi}}\boldsymbol{\alpha}$, which in this case is obviously equivalent to the problem $\frac{1}{l}\mathbf{K}^{\hat{\Phi}}\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha}$. Furthermore, since $\kappa(\mathbf{x_t}, \mathbf{x_s}) = \kappa(\mathbf{x_s}, \mathbf{x_t})$ and[8] $\boldsymbol{\alpha}^\top \frac{1}{l}\mathbf{K}^{\hat{\Phi}}\boldsymbol{\alpha} = \frac{1}{l}\mathbf{a}^\top\mathbf{a} \geq 0$, the matrix $\frac{1}{l}\mathbf{K}^{\hat{\Phi}}$ is symmetric positive semidefinite, and thus its eigenvectors are orthogonal and the corresponding real eigenvalues are non-negative. Let the m positive dominant eigenvalues of $\frac{1}{l}\mathbf{K}^{\hat{\Phi}}$ be denoted by $\lambda_1 \geq \ldots \geq \lambda_m > 0$ and the correcponding normalized eigenvectors by $\boldsymbol{\alpha}^{\mathbf{1}}, \ldots, \boldsymbol{\alpha}^{\mathbf{m}}$. Then the orthogonal matrix of the transformation we need can be calculated as below.

$$\mathbf{A}_{\hat{\Phi}} := \left[\frac{1}{\sqrt{l\lambda_1}} \sum_{i=1}^{l} \alpha_i^1 \hat{\Phi}(\mathbf{x_i}), \ldots, \frac{1}{\sqrt{l\lambda_m}} \sum_{i=1}^{l} \alpha_i^m \hat{\Phi}(\mathbf{x_i})\right], \tag{4}$$

where the factors $1/\sqrt{l\lambda}$ are needed to keep the columnvectors of $\mathbf{A}_{\hat{\Phi}}$ normalized.

Transformation of Test Vectors:

Let $\mathbf{y}$ be an arbitrary test vector. After preprocessing $\Phi(\mathbf{y})$ we get that $\hat{\Phi}(\mathbf{y}) = \Phi(\mathbf{y}) - \boldsymbol{\mu}^\phi$. Then

$$\mathbf{y}' = \mathbf{A}_{\hat{\Phi}}{}^\top \hat{\Phi}(\mathbf{y}) == \left[\frac{1}{\sqrt{l\lambda_1}} \sum_{i=1}^{l} \alpha_i^1 c_i, \ldots, \frac{1}{\sqrt{l\lambda_m}} \sum_{i=1}^{l} \alpha_i^m c_i\right]^\top, \tag{5}$$

where $c_i = \hat{\Phi}(\mathbf{x_i})^\top \hat{\Phi}(\mathbf{y}) = \kappa(\mathbf{x_i}, \mathbf{y}) - \left(\frac{1}{l}\sum_{j=1}^{l} \left(\kappa(\mathbf{x_i}, \mathbf{x_j}) + \kappa(\mathbf{x_j}, \mathbf{y})\right)\right) + \frac{1}{l^2}\sum_{j=1}^{l}\sum_{k=1}^{l} \kappa(\mathbf{x_j}, \mathbf{x_k})$. In our experience the strategy for obtaining a suitable m was the same as in PCA.

[7] Schölkopf et al. give $\mathbf{K}^{\hat{\Phi}}$ in a matrix form using additional matrices. Our formula, however, turned out to be easier to code, and resulted in a more effective program.

[8] Here we temporarily disregard the constraint $\mathbf{a} \neq 0$.

References

[1] Alder, M. D. *Principles of Pattern Classification: Statistical, Neural Net and Syntactic Methods of Getting Robots to See and Hear,* http://ciips.ee.uwa.edu.au/˜mike/PatRec, 1994.

[2] Battle, E., Nadeu, C. and Fonollosa, J. A. R. Feature Decorrelation Methods in Speech Recognition. A Comparative Study. In *Proceedings of ICSLP'98*, 1998.

[3] Kocsor, A., Kuba, A. Jr. and Tóth, L. An Overview of the OASIS speech recognition project, *In Proceedings of ICAI'99*, 1999(in press).

[4] Kocsor, A., Tóth, L., Kuba, A. Jr., Kovács, K., Jelasity, M., Gyimóthy, T. and Csirik, J. A Comparative Study of Several Feature Transformation and Learning Methods for Phoneme Classification submitted to *International Journal of Speech and Technology*.

[5] Schölkopf, B., Mika, S., Burges, C. J. C., Knirsch, P., Müller, K., Rätsch, G. and Smola, A. J. Input Space vs. Feature Space in Kernel-Based Methods, *IEEE Transactions on Neural Networks*, 1999(in press).

[6] Tipping, M. E. and Bishop, C. M. Probabilistic Principal Component Analysis, Technical Report NCRG/97/010, Neural Computing Research Group, Aston University, Birmingham, United Kingdom, 1997.

Quasi-Orders on Automata

Jelena Kovačević

A reflexive and transitive binary relation is called a *quasi-order*, or a *preorder*, in some sources. As known, quasi-orders play an important role in many mathematical theories, in Mathematical Logic, Algebra, Topology, Computer Science etc. The main aim of the present paper is to give some applications of quasi-orders in Automata Theory.

A nonempty set Q equipped with a quasi-order is said to be a *quasi-ordered set*. If Q can be represented as a disjoint union of a family $\{Q_\alpha\}_{\alpha\in Y}$ of its subsets such that the elements from different Q_α's are incomparable, then Q is said to be a *direct sum* of quasi-ordered sets Q_α, $\alpha \in Y$. Direct sum decompositions of quasi-ordered sets were investigated by Ćirić, Bogdanović and Kovačević in [3], 1998, through ideals, filters (or dual ideals), and *double ideals*, which are defined as subsets which are both ideals and filters. It was proved that double ideals of a quasi-ordered set Q form a complete atomic Boolean algebra which is the center both of the lattice $\boldsymbol{I}(Q)$ of ideals and the lattice $\boldsymbol{F}(Q)$ of filters of Q, and that the following holds:

Theorem 2 *Every quasi-ordered set Q can be uniquely represented as a direct sum of direct sum indecomposable automata.*

This is the greatest direct sum decomposition of Q and its summands are exactly the atoms of the Boolean algebra of double ideals of Q.

Using the same concept we established a correspondence between direct sum decompositions of Q and direct product decompositions of the lattices $\boldsymbol{I}(Q)$ and $\boldsymbol{F}(Q)$.

The mentioned results can be applied to automata if they are considered as quasi-ordered sets with respect to quasi-orders defined in a natural way. Namely, if A is an automaton (deterministic or nondeterministic), then we define a quasi-order $\preceq$ on A saying that $a \preceq b$ if and only if there exists an input word which takes a into b. Here we show that the lattice of subautomata of the automaton A is isomorphic to the lattice of filters of the quasi-ordered set $(A, \preceq)$, and that the lattice of direct sum decompositions of the automaton A is isomorphic to the lattice of direct sum decomposition of the quasi-ordered set $(A, \preceq)$. As consequences we obtain the results given by Ćirić and Bogdanović in [5], 1999, which say that every automaton A can be represented as a direct sum of direct sum indecomposable automata A_α, $\alpha \in Y$, and that the lattice $\mathrm{Sub}(A)$ of subautomata of A can be represented as a direct product of direct product indecomposable lattices L_α, $\alpha \in Y$, where $L_\alpha \cong \mathrm{Sub}(A_\alpha)$, for every $\alpha \in Y$.

We also give other applications of quasi-orders to study of direct sum and subdirect product decompositions of deterministic (not necessarily finite) automata. Namely, we define a quasi-order π on an automaton A to be *positive* if $a\,\pi\,au$, for every state a and input word u. If π is a positive quasi-order on A, then a state a is called π*-reversible* if $au\,\pi\,a$, for every input word u, and A is said to be π*-connected* if for any two states a, b there exists an input word v such that $a\,\pi\,bv$. States which are $\preceq$-reversible are called *reversible*, whereas $\preceq$-connected automata are exactly the connected ones. An automaton is *reversible* if any its state is reversible. Using these concepts we prove the following:

Theorem 3 *Let A be an automaton with a countable input alphabet. Then the following conditions are equivalent:*

(i) *There exists a positive quasi-order π on A such that A is π-connected and every π-reversible state of A is an ordinary reversible state.*

(ii) *A satisfies one of the following two conditions:*
   (1) *A is an extension of a reversible automaton by a trap-connected automaton.*
   (2) *A does not have a trap and it is a subdirect product of countably many trap-connected automata.*

On the other hand, we define a positive quasi-order π on A to satisfy the *quadrangle property* if for any two states a, b, $a\,\pi\,b$ implies that for every input word u there exists an input word v such that $au\,\pi\,bv$. The next three theorems demonstrate the role of quasi-orders with the quadrangle property in direct sum decompositions of automata.

Theorem 4 *If π is a positive quasi-order on an automaton A having the quadrangle property, then A can be represented as a direct sum of automata A_α, $\alpha \in Y$, such that any A_α is direct sum indecomposable and π_α-connected, where π_α is the restriction of π onto A_α.*

Conversely, if A is a direct sum of automata A_α, and for any $\alpha \in Y$, A_α is π_α-connected, then $\pi = \bigcup_{\alpha\in Y} \pi_\alpha$ is a positive quasi-order on A with the quadrangle property.

Theorem 5 *If A is an automaton with a countable input alphabet, then the following conditions are equivalent:*

(i) *There exists a positive quasi-order π on A having the quadrangle property such that every π-reversible state of A is an ordinary reversible state.*
(ii) *A is a direct sum of direct sum indecomposable automata A_α, $\alpha \in Y$, such that any A_α is either an extension of a reversible automaton by a trap-connected automaton or it does not have a trap and it is a subdirect product of countably many trap-connected automata.*

Theorem 6 *The following conditions on an automaton A are equivalent:*

(i) *The quasi-order $\preceq$ on A has the quadrangle property;*
(ii) *A is locally connected, i.e. every monogenic subautomaton of A is connected;*
(iv) *A is a direct sum of connected automata;*
(v) *$D(H) = \{a \in A \mid au \in H,$ for some input word $u\}$ is a subautomaton of A, for every subautomaton H of A.*

References

[1] S. Bogdanović, M. Ćirić, T. Petković, B. Imreh and M. Steinby, *Local properties of unary algebras and automata*, (to appear).
[2] M. Ćirić and S. Bogdanović, *Lattices of subautomata and direct sum decompositions of automata*, Algebra Colloquium **6**:1 (1999), 71–88.
[3] M. Ćirić, S. Bogdanović and J. Kovačević, *Direct sum decompositions of quasi-ordered sets and their applications*, Filomat (Niš) **12:1** (1998), 65–82.
[4] M. Ćirić, S. Bogdanović and T. Petković, *The lattice of subautomata of an automaton - A survey*, Publ. Inst. Math. Belgrade 64(78) (1998), 165–182.
[5] J. Kovačević, *Decompositions of ordered sets, semigroups and automata*, Master Thesis, University of Niš, Faculty of Sciences and Mathematics, 1999 (in Serbian).
[6] J. Kovačević, M. Ćirić, T. Petković and S. Bogdanović, *Decompositions of automata and reversible states*, in: A. Adam and P. Dömösi (eds.), Proceedings of the Nineth International Conference on Automata and Formal Languages, Publ. Math. Debrecen (submitted).

Verification solutions of packing circle problems

Mihály Csaba Markót, Péter Gábor Szabó, Boglárka Tóth and Tamás Vinkó

The discussed packing circle problem can be formalized in the following way: place a given number of n equal circles without overlapping into a unit square maximizing the size of the circles. In the last decade some computer-aided proofs came to light presenting globally optimal packings based on traditional real arithmetic. Due to the well-known representing and rounding problems of real numbers, methods with guaranteed accuracy are required to verify these results. In our present study two different kinds of interval branch-and-bound algorithms are introduced, providing reliable optimal solutions for both local and global cases.

Search space reduction criterion based on derivatives in Global Optimization algorithms [9]

J.A. Martínez, L.G. Casado, I. García and Ya.D. Sergeyev

Interval Global Optimization algorithms are based on a Branch and Bound scheme using the following five rules: bounding, termination, selection, subdivision and elimination. The research in interval Global Optimization algorithms try to determine the appropiated B&B rules. An example is the selection of a specific subdivision rule [1]. The elimination rule is one of the most investigated. The simplest elimination rules are the midpoint and monotonicity tests [4]. Most of the proposals have been devised to improve the efficiency based on the derivative information, such as monotonicity, concavity and Newton Method tests [2, 3]. Here we develop a new elimination and subdivision technique which also uses derivative information in one dimensional functions.

This paper investigates interval Global Optimization algorithms for solving the box constrained Global Optimization problem:

$$\min_{x\in X} f(x), \tag{6}$$

where the interval $X = [\underline{x}, \overline{x}] \subseteq \mathbb{R}$ is the search region, and $f(x) : X \subset \mathbb{R} \to \mathbb{R}$ is the objective function. The global minimum value of f is denoted by f^*, and the set of global minimizer points of f on X by X^*. That is,

$$f^* = \min_{x\in X} f(x) \qquad \text{and} \qquad X^* = \{x^* \mid f(x^*) = f^*\}.$$

Herein real numbers are denoted by $x, y, \ldots$, and a real bounded and closed interval by $X = [\underline{x}, \overline{x}]$, where $\underline{x} = \min X$ and $\overline{x} = \max X$. The set of compact intervals is denoted by $\mathbb{I} := \{[a, b] \mid a \leq b,\ a, b \in \mathbb{R}\}$. A function $F : \mathbb{I} \to \mathbb{I}$ is called *inclusion function* of f in $X \subseteq \mathbb{R} \to \mathbb{R}$, if $x \in X$ implies $f(x) \in F(X)$. In other words, $f(X) \subseteq F(X)$, where $f(X)$ is the range of the function f on X. It is assumed in the present study that the inclusion function of the objective function is available (possibly given by interval arithmetic).

For a given interval X, we denote $f_- = \overline{F}(\underline{x})$ and $f_+ = \overline{F}(\overline{x})$. When we express a real number as an interval, we shall usually retain the simpler noninterval notation. For example x in place of $[x, x]$ [2].

Let's denote the derivative of the inclusion function in the interval X by $G(X) = F'(X) = [\underline{G}, \overline{G}]$, the straight line with slope $\underline{G}$ at point $\underline{x}$ by L, the straight line width slope $\overline{G}$ at point $\overline{x}$ by U, and the intersection between L and U by (x_m, y_m)

The new ideas are described in Algorithm 1.

The algorithm is based on a more efficient (in comparison with traditional approaches) usage of the search information about the lower and upper bounds of the first derivative. A graphical example of algorithm 1 is shown in Figure 7.

Extensive numerical examples will be presented and compared with traditional interval Global Optimization algorithm using Newton method.

[9] This work was supported by the Ministry of Education of Spain (CICYT TIC99-0361, and by the Grants FKFP 0739/97, OTKA T016413 and T 017241.

Algorithm 1 Description of Algorithm

```
proc ISBD(F, X, ε, T, Q) ≡
  F                                              Inclusion Function of F
  Q = {}, T := (X)                               Final and Work Lists
  f̃ = min{f₋, f₊}
  G(X) := [G̲, G̅]
  Calculate L and U
  (x_m, y_m) = L∩U
  while ( T ≠ {} )
        T := T − {X}
        if y_m < f̃
          f_m = f(x_m)
          if (f_m < f̃)
            f̃ = f(x_m)
            X_new = [L ∩ f̃, U ∩ f̃]
          if (U ∩ f̃) ∈ [x̲, x_m]
            X_left = [x̲, U ∩ f̃]
            G(X_left) := [G̲, G̅]
            Calculate L and U
            (x_m, y_m) = L∩U
            if ( w(X_left) < ε )                 Termination Criterion
              then Q := Q + {X_left}
               else T := T + {X_left}
          if (L ∩ f̃) ∈ [x̲, x_m]
            X_rigth = [L ∩ f̃, x̅]
            G(X_rigth) := [G̲, G̅]
            Calculate L and U
            (x_m, y_m) = L∩U
            if ( w(X_rigth) < ε )                Termination Criterion
              then Q := Q + {X_rigth}
               else T := T + {X_rigth}
  end
```

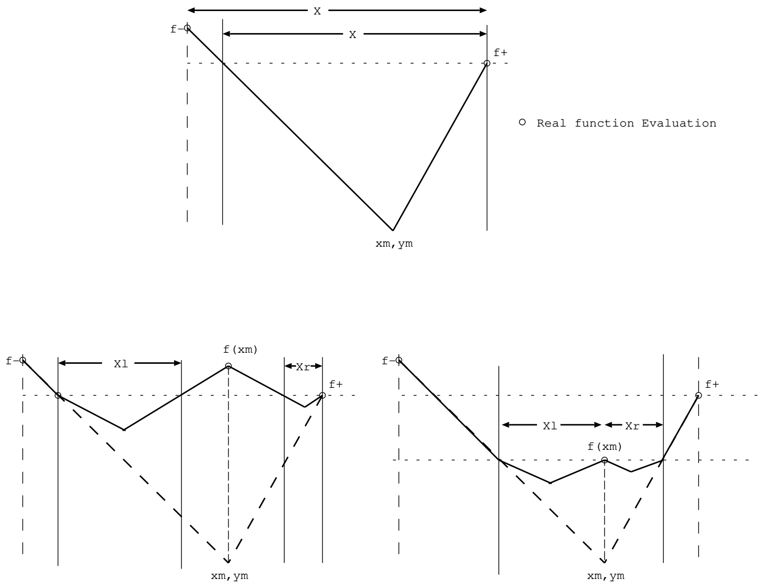

Figure 7: Example of algorithm execution

References

[1] Csallner, A. E., Csendes, T., Markót, M. C.: Multisection in Interval Branch-and-Bound Methods for Global Optimization I. Accepted for publication in the J. Global Optimization.

[2] Hansen, E.: Global optimization using interval analysis. New York: Marcel Decker 1992.

[3] Kearfott, R. B.: Rigorous global search: continuous problems. Dordrecht: Kluwer 1996.

[4] Ratschek, H., Rokne, J.: New Computer Methods for Global Optimization. Chichester: Ellis Horwood 1988.

[5] Törn, A., Žilinskas, A.: Global Optimization. Berlin: Springer 1987.

Definition of a Parallel Execution Model with Abstract State Machines[10]

Zsolt Németh

LOGFLOW is a fine-grained all-solution parallel (reduced) Prolog system for distributed memory architectures. Its abstract execution model is logicflow [3] that can be considered as a kind of macro dataflow scheme, whereas its abstract machine model is the Distributed Data Driven Prolog Abstract Machine [4] (3DPAM). 3DPAM tries to make a connection between a dataflow based execution model and a kind of von Neumann physical architecture.

However a new, hybrid multithreaded architectural platform offers the possibility to create a more efficient Prolog abstract machine. Its ability to hide latencies due to remote memory access or synchronisation (multithreading) opens a new way for representing Prolog data (heap) and managing the variables. On the other hand, its hybrid feature, i.e. support for both the fast sequential and dataflow execution, is close to the macro dataflow model of LOGFLOW and makes possible an efficient realisation of dataflow nodes and token streams. To exploit the latter property at the abstract machine level, a new abstract execution model is necessary, too. The new execution model can be derived from the logicflow in three steps by changing the way how solution streams are separated, the way how solutions are propagated and by grouping together elementary nodes. Whereas the gain in efficiency is obvious, it is not the case for correctness and semantical equivalence of the models.

Abstract State Machines (Gurevich's ASMs, formerly known as evolving algebras) offer a way for the design and analysis of complex hardware and software systems [1][2]. They are similar to Turing machines in a sense that they simulate algorithms yet, they are able to describe semantics at arbitrary levels of abstraction. An ASM consists of a finite set of transition rules by which the system is driven form state to state, each represented by sets with relations and functions (algebras). By refinement steps a 'more abstract' model can be turned into a 'more concrete' one and by relating their states and transition rules (by proof mapping), their relative correctness and completeness can be proven. In several refinement steps the equivalence of different models can be shown.

In this paper the redesign steps of the logicflow model are presented. The original logicflow model is described by an ASM then the successive steps of the modifications are introduced by new, refined ASMs. The sequence of refinements results the new abstract execution model. If all subsequent models can be mapped to their sibling models with respect to correctness, the original logicflow and the resulted one can be stated as functionally equivalent. Furthermore, the execution model can be related to sequential Prolog models showing different properties.

ASMs are also able to help the design process from the abstract execution model to the abstract machine model. While the previous transition between two execution models remained at the same level of abstraction, the principles of abstract execution model are turned into more specific forms of abstract machine at lower and lower level of abstraction by ASMs. The paper tries to give an insight into the design of compilation schemes and abstract instructions by ASMs.

The aim of the paper is defining a new execution model together with its abstract machine and showing their correctness with respect to their ancestor model and the usual Prolog execution. The ASM methodology seems to be a proper framework to fulfil the two tasks at the same time.

References

[1] E. Börger: High Level System Design and Analysis using Abstract State Machines. ASM Workshop, Magdeburg, 21-22 September, 1998.

[2] Yu. Gurevich: Evolving Algebras: An Attempt to Discover Semantics. In: Current Trends in Theoretical Computer Science. Eds. G. Rozenberg, A. Salomaa, World Scientific, 1993. pp. 266-292.

[3] P. Kacsuk: Execution Models for a Massively Parallel Prolog Implementation. Journal of Computers and Artificial Intelligence, Vol. 17, No. 4. Slovak Academy of Sciences, 1998 (part 1) and Vol. 18, No. 2., 1999 (part 2)

[4] P. Kacsuk: Distributed Data Driven Prolog Abstract Machine. In: P. Kacsuk, M.J.Wise: Implementations of Distributed Prolog. Wiley, 1992.

[10]The work reported in this paper was supported by the National Research Grant (OTKA) registered under No. T-022106.

Standardizing the MR Image Intensity Scale and Its Applications [11]

László G. Nyúl and Jayaram K. Udupa

Magnetic Resonance Imaging (MRI) has revolutionized radiological imaging of the internal structures of the human body. It has the advantage of being noninvasive with no known health hazards. A variety of MRI protocols are currently available, with and without the use of contrast agents, such as T1, T1 with a contrast agent, T2 and proton density (Pd) with spin-echo (SE) or fast spin-echo (FSE) sequences, magnetization transfer (MT), FLAIR, SPGR, and GRASS. These protocols allow the setting up of different contrasts among the different tissues within the same organ system. Ironically, this richness of acquisition schemes comes with a major difficulty. Unlike in other modalities such as x-ray computerized tomography, MR images taken for the same patient on the same scanner at different times may appear different from each other due to a variety of scanner-dependent variations, and therefore, the absolute intensity values do not have a fixed meaning. This implies that MR images cannot be displayed at preset windows; one may have to adjust the window settings per case. The lack of a standard and quantifiable interpretation of image intensities also poses problems in image segmentation and quantification.

We have devised a two-step method wherein all images (independent of patients and the specific brand of the MR scanner used) can be transformed in such a way that for the same protocol and body region, in the transformed images similar intensities will have similar tissue meaning. In the first step, the parameters of the standardizing transformation are "learned" from an image set. In the second step, for each MR study, these parameters are utilized to determine the mapping needed to deform its histogram into the standardized histogram. The method was tested quantitatively on 90 whole brain studies of Multiple Sclerosis patients for several protocols and qualitatively for several other protocols and body regions. As measured by mean squared difference and the coefficient of variation of the mean tissue intensities, standardized images have statistically significantly more consistent range and intensity meaning for tissues than those without. This consistency achieved seems to be independent of the scanners. Fixed gray level windows can be established for the standardized images and used for display without the need of per case adjustment. Image analysis and tissue segmentation methods are considerably improved in terms of their constancy of parameter settings and their degree of automation. With standardization, numerical meaning is achieved, and hence numerical diagnosis and study of diseases may become possible. The method can be easily implemented in a PACS via DICOM value of interest look up tables.

[11] This work was supported in part by the NIH grant NS 37172 and in part by the Department of Army under Grant DAMD 179717271.

Parallel simulation of spiral waves in reacting and diffusing media

E.M. Ortigosa, L.F. Romero and J.I. Ramos

Reaction-diffusion equations are ubiquitous in biology, combustion, etc., because of their relevance in pattern formation, ignition and extinction phenomena, etc. [1-3]. Many studies of reaction-diffusion equations have been concerned with equations for activators and inhibitors in one or two spatial dimensions, e.g., the Belousov-Zhabotinskii, Brusselator and Oregonator models [3]. In three dimensions, there have been very few analytical and numerical studies of spiral waves, presumably because of both the large difficulties in examining wave propagation in three-dimensional (3D) space and the cost of such simulations [4].

In this talk, 3D simulations of the propagation of spiral waves in cubes in the presence and absence of extinction sources will be presented. These numerical simulations have been carried out by means of both time-linearized and nonlinearized techniques with and without approximate factorization of the 3D operator into one-dimensional ones.

The numerical methods employed in the discretization of the governing partial differential equations have been implemented in a parallel fashion in both shared- and distributed-memory computers. The parallelization of the approximate factorization technique has been carried out with a block dynamic cartesian decomposition and its efficiency is very near to one. The parallelization of time-linearization methods has been performed in terms of both the overlapping of communications and computations. This overlapping has been carried out by using asynchronous messages in the message-passing version and prefetch directives in shared-memory computers.

The presented model have been applied to study the formation and propagation of spiral waves in three dimensions, as well as their interactions with the boundaries of the computational domain, their local extinction, and the formation of spiral filaments. In the experiments calculations have been performed using 200x200x200 grid points in an Origin-2000 distributed-shared memory (DSM) computer up to 16 processors.

The numerical results indicate that, as the spiral wave approaches the boundaries of the computational domain, its tip describes a curved trajectory and the spiral re-emerges from another place. They also show that local heating may result in local and temporal extinction of the spiral wave and the formation of islands or pockets disconnected from the wave. A study of the isoscalability indicates that it is possible to reach speed-ups of about or higher than 56 with 800x800x800 grid points in a 64-processor computer.

References

[1] Murray, J.D. *Mathematical Biology*, Springer-Verlag: New York, 1989.

[2] Williams, F.A. *Combustion Theory*, second edition, Addison-Wesley: New York, 1985.

[3] Holden, A.V., Markus, M. & Othmer, H.G. (eds). *Nonlinear Wave Propagation in Excitable Media*, Plenum Press: New York, 1991.

[4] Keener, J. & Sneyd, J. *Mathematical Physiology*, Springer-Verlag: New York, 1998.

[5] Barrett, R., Berry, M., Chan, T.F., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C. & Van de Vorst, H., *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM: Philadelphia, 1994.

[6] Aykanat, C., Özgüner, F. & Scott, D.S. Vectorization and parallelization of the conjugate gradient algorithm on hypercube-connected vector processors, *Microprocess. Microprogram.*, **29**, pp. 67–82, 1990.

[7] Chronopoulos, A. T. & Gear C. W. S-step iterative methods for symmetric linear systems, *J. comput. Appl. Math.*, **25**, pp. 153–168, 1989.

[8] Van der Wijngaart, R.F. Efficient Implementation of a 3-Dimensional ADI Method on the iPSC/860, *Supercomputing'93*, pp. 102–111, 1993.

A 3D Directional Shrinking Algorithm

Kálmán Palágyi

1. Introduction

A 3D binary picture [5] is a mapping that assigns the value of 0 or 1 to each point with integer coordinates in the 3D digital space denoted by $\mathbb{Z}^3$. Points having the value of 1 are called black points and form the objects of the picture, while 0's are called white ones and form the background, the holes, and the cavities of the picture.

The shrinking of binary pictures to similarly connected representations that have smaller foregrounds (i.e., fewer 1's) has found application as a fundamental preprocessing step in image processing [3]. Two forms of such shrinking have been emerged:

1. The picture is transformed to its topological kernel, where the shrunk picture is topologically equivalent to the original one;
2. objects (connected components) are shrunk to isolated points (i.e., single–point residues which may then be deleted).

The only 3D topology preserving shrinking algorithm has been proposed by Bertrand and Aktouf [2]. Their thinning algorithm can extract the topological kernel of an object if no end–point condition is applied. The strategy which is used for deleting 1's in parallel without altering the topology of the picture based upon subfields: the cubic grid $\mathbb{Z}^3$ is divided into 8 subfields which are successively activated in each iteration step. The parallel algorithm examines the $3 \times 3 \times 3$ neighborhood of the object points.

Two 3D shrinking algorithms belonging to the second type are known: Arcelli and Levialdi [1] proposed a parallel algorithm capable of transforming any finite object to an isolated 1 in a finite number of iteration step. Only the $2 \times 2 \times 2$ neighborhood of 1's are investigated and the object to be shrunk never leaves its circumscribing box. Hall and Küçük [4] developed the other algorithm which uses 2 subfields and examines the $3 \times 3 \times 3$ neighborhood of the object points.

In this work, a new 3D parallel shrinking algorithm is proposed for extracting the topological kernel of a binary picture. Our strategy which is used for preserving the topology is called directional or border sequential: Iteration steps are divided into 6 successive parallel subiterations, where only border 1's of a certain kind can be deleted in each subiteration. The algorithm examines the $3 \times 3 \times 3$ neighborhood of 1's and it is topology preserving for any $(26, 6)$ pictures.

2. Basic Notions and Results

Let p be a point in the 3D digital space $\mathbb{Z}^3$. Let us denote $N_j(p)$ (for $j = 6, 18, 26$) the set of points *j–adjacent* to a point p (see Fig. 8).

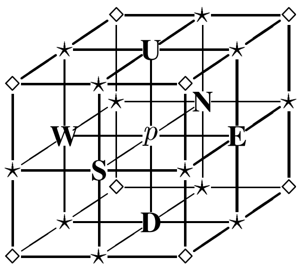

Figure 8: Frequently used adjacencies in $\mathbb{Z}^3$. The set $N_6(p)$ contains the central point p and points marked **U**, **N**, **E**, **S**, **W**, and **D**. The set of points $N_{18}(p)$ contains the set $N_6(p)$ and points marked "$\star$". The set of points $N_{26}(p)$ contains the set $N_{18}(p)$ and points marked "$\diamond$".

The sequence of distinct points $\langle x_0, x_1, \ldots, x_n \rangle$ is a *j–path* (for $j = 6, 18, 26$) of length n from point x_0 to point x_n in a non–empty set of points X if each point of the sequence is in X and x_i is j–adjacent to x_{i-1} for each $1 \leq i \leq n$. Note that a single point is a j–path of length 0. Two points are *j–connected* in the set X if there is a j–path in X between them. A set of points X is *j–connected* in the set of points $Y \supseteq X$ if any two points in X are j–connected in Y.

The *3D binary (m,n) digital picture* $\mathcal{P}$ is a quadruple $\mathcal{P} = (\mathbb{Z}^3, m, n, B)$ [5]. Each element of $\mathbb{Z}^3$ is called a *point* of $\mathcal{P}$. Each point in $B \subseteq \mathbb{Z}^3$ is called a *black point* and value 1 is assigned to it. Each point in $\mathbb{Z}^3\backslash B$ is called a *white point* and value 0 is assigned to it. Adjacency m belongs to the black points and adjacency n belongs to the white points. A *black component* is a maximal m–connected set of points in B. A *white component* is a maximal n–connected set of points in $\mathbb{Z}^3\backslash B$.

We are dealing with (26,6) pictures. It is assumed that any picture contains finitely many black points.

A black point p is said to be a *border point* if the set $N_6(p)$ contains at least one white point. A border point p is called a **U**–*border point* if the point marked by **U** in Fig. 8 is white. We can define **N**–, **E**–, **S**–, **W**–, and **D**–border points in the same way.

A black point is called a *simple point* if its deletion does not alter the topology of the picture. We make use the following result for (26,6) pictures:

CRITERION 1. [6]
Black point p is simple in picture $(\mathbb{Z}^3, 26, 6, B)$ if and only if all of the following three conditions hold:

1. *The set $(B\backslash\{p\}) \cap N_{26}(p)$ contains exactly one 26–component.*
2. *The set $(\mathbb{Z}^3\backslash B) \cap N_6(p)$ is not empty and it is 6–connected in the set $(\mathbb{Z}^3\backslash B) \cap N_{18}(p)$.*

Parallel reduction operations delete a set of black points and not only a single simple point (and each white point remains the same). We need to consider what is meant topology preservation when a number of black points are deleted simultaneously. The following sufficient conditions for parallel reduction operations of 3D (26,6) pictures are stated:

THEOREM 2. [7]
Let $\mathcal{T}$ be a parallel reduction operation. Let p be any black point in any picture $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$ so that p is deleted by $\mathcal{T}$. Let $Q \subseteq (N_{18}(p)\backslash\{p\}) \cap B$ be any set of black points in picture $\mathcal{P}$. Operation $\mathcal{T}$ is topology preserving for (26,6) pictures if all of the following conditions hold:

1. *p is simple in the picture $(\mathbb{Z}^3, 26, 6, B\backslash Q)$.*
2. *No black component contained in a unit lattice cube can be deleted completely by operation $\mathcal{T}$.*

3. The New Shrinking Algorithm

The proposed directional 6–subiteration shrinking algorithm can be sketched by the following program:

Input: binary array X representing the picture $\mathcal{P} = (\mathbb{Z}^3, 26, 6, B)$;
Output: binary array Y representing the shrunk picture.

```
6_subiteration_shrinking_algorithm(X,Y)
  begin
    Y = X;
    repeat
      Y = deletion_from_U(Y);
      Y = deletion_from_D(Y);
      Y = deletion_from_N(Y);
      Y = deletion_from_S(Y);
      Y = deletion_from_E(Y);
      Y = deletion_from_W(Y);
    until no points are deleted;
  end.
```

Our algorithm terminates when there are no more black points to be deleted. Since all considered input pictures are finite, the shrinking algorithm will terminate.

Deletable points in a subiteration are given by a set of $3 \times 3 \times 3$ matching templates. A black point is deletable if at least one template in the set of templates matches it. Templates are described by three kinds of elements, "•" (black), "○" (white), and "." ("don't care"), where "don't care" matches either black or white point in a given picture.

The first subiteration (deletion_from_U) assigned to the deletion direction **U** can delete certain **U**–border points; the second subiteration associated with the deletion direction **D** attempt to delete **D**–border points, and so on.

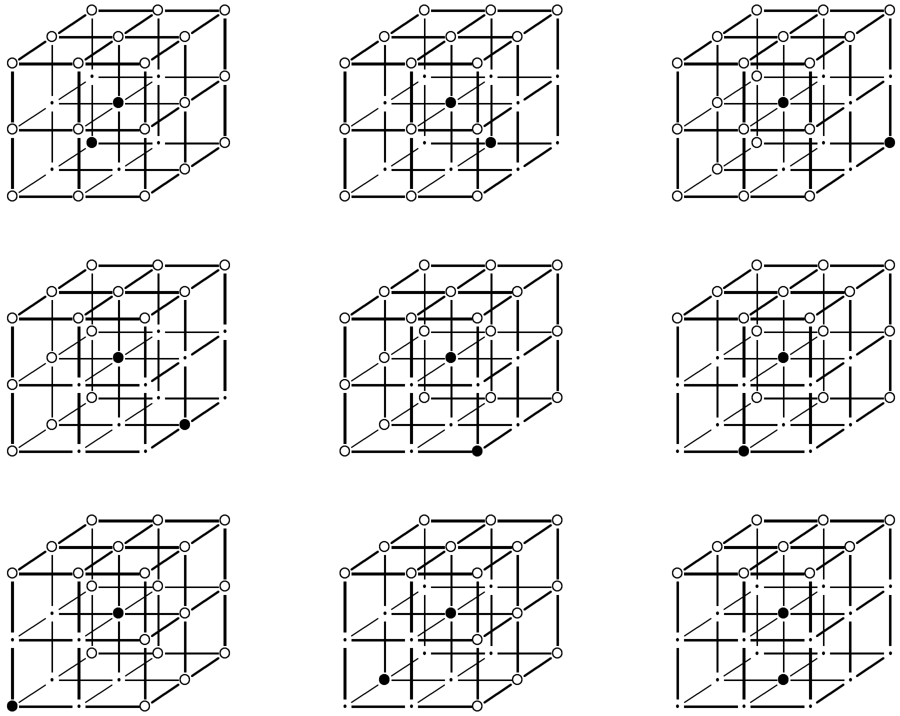

Figure 9: The set of templates assigned to the first subiteration of the proposed algorithm. These templates can delete certain **U**–border points. Notations: every position marked "•" matches a black point; every position marked "∘" matches a white point; every "·" ("don't care") matches either a black or a white point.

The set of templates presented in Fig. 9 is assigned to the first subiteration. The deletable points of the other five subiterations can be obtained by proper rotations and/or reflections of the templates in Fig. 9. Note that choosing another order of the deletion directions yields another algorithm, but it does not alter the topological correctness.

Note that the templates of our algorithm can be regarded as a "careful" characterization of simple points. This Boolean characterization makes easy implementation possible.

The topological correctness is stated by the following theorem:

THEOREM 3.
Each subiteration of the proposed 3D shrinking algorithm is a topology preserving reduction for $(26,6)$ *pictures. (Therefore, the entire algorithm is topology preserving, too.)*

Theorem 3. can be proved easily by using Criterion 1. and Theorem 2.

The proposed algorithm is capable of transforming each simply–connected object (i.e., object without holes and cavities [5]) to an isolated point. The shrunk multiply–connected object (i.e., object containing holes or cavities) is a closed "thin" curve or a closed "thin" surface.

References

[1] C. Arcelli, S. Levialdi, Parallel shrinking in three dimensions, *Computer Graphics and Image Processing* **1**, 1972, 21–30.

[2] G. Bertrand, Z. Aktouf, A 3D thinning algorithms using subfields, in *Proc, Conf. on Vision Geometry III*, Proc. SPIE **2356**, 1994, 113–124.

[3] R.W. Hall, T.Y, Kong, A. Rosenfeld, Shrinking binary images, in *Topological Algorithms for Digital Image Processing*, Machine Intelligence and Pattern Recognition 19, North–Holland, 1996, 31–98.

[4] R.W. Hall, Ş. Küçük, Parallel 3D shrinking algorithms using subfields notions, in *Proc. 11th IEEE Int. Conf. Pattern Recognition*, 1992, 395–398.

[5] T.Y. Kong, A. Rosenfeld, Digital topology: Introduction and survey, *Computer Vision, Graphics, and Image Processing* **48**, 1989, 357–393.

[6] G. Malandain, G. Bertrand, Fast characterization of 3D simple points, in *IEEE Int. Conf. Pattern Recognition*, 1992, 232–235.

[7] K. Palágyi, A. Kuba, A parallel 3D 12–subiteration thinning algorithm, *Graphical Models and Image Processing* **61**, 1999, 199–221.

Static Specification Completeness Checking of UML State Machines

Zsigmond Pap

Safe programs never process such operations, that can cause accident, human casualty or environment harm. The design of safe programs is always verified according to the specification, but if the specification is incomplete or inconsistent, verification and validation can't find the possible implementation errors, which may result in accidents. Accordingly, the checking of the completeness and consistency of the specification is cru- cial.

There are numerous formal specification methods proposed in the literature. Generally they use a descrip- tion language that can be easily processed and checked by a computer. The specification check must be fast and efficient. This means, for example, that the checker should not build the full reachability tree correspond- ing to the specification, because this is a very slow process and requires a lot of resources. Our goal is to work out a static checking method and tool, which can check some aspects of the specification without gener- ating the reachability tree.

Our tool aims at checking specifications developed using UML, the Unified Modeling Language. Especially the formalism used to describe the dynamic behaviour of the system, i.e. the Statechart specification should be checked. Statecharts, as extension of state diagrams, consist of states and labelled transitions, but allow state hierarchy and concurrency. A state can be normal or complex. Complex states can be concurrent super- states, which result in a hierarchical tree of states. There are special (pseudo) states, like the initial state or the history state. The transitions are labelled by a trigger event, a guard condition and actions. When an event occurs, all transitions triggered by this event can fire if the state machine is in the specified source state, and the guarding condition is true. When a transition fires, the state machine goes to the destination state, and all actions associated with the transition start. There are special transitions. The destination state of a conditional transition depends on some conditions. The fork transition allows starting concurrent sub-automata, the join transition help to step into a single state from concurrent states. The completion transition has no trigger event, fires immediately when the guarding condition becomes true.

The completeness checking of the Statechart specification is based on the following rules:

- All possible trigger events should be considered when the transitions leaving a given state are examined. There must not be ambiguous, non-deterministic situations, when the state machine can reach different states and there is no priority between the corresponding transitions (triggered by the same event). How- ever, two transitions triggered with the same event can be unambiguous, if the guarding conditions can't be true at the same time.
- In a hierarchical model the sub-states inherit the transition from their parent states. To verify the com- pleteness, it is sufficient to check the leaf-states only, i.e. all non pseudo-states without sub-states. If at least two transitions have true guard conditions and can fire at the same time and on the same hierarchy level, the specification is ambiguous.
- The join and fork transitions can be transformed to normal transitions with special guard conditions. The conditional transitions need the following checking: the branch cases must form a tautology, and one of them must be true in any case.
- If there is a state without incoming transitions (except the initial pseudostate), then this state is not reach- able, which means a specification error.
- If in a given hierarchy level the guard conditions form a tautology, a transition on a higher level might be unnecessary.
- Each state of a program must be limited in time. Accordingly, from every stable state must start a transi- tion triggered by a time-out event. This transition fires when the maximum time in that state is exceeded.

Our checker was integrated with the commercial UML CASE tool "MID Innovator". From the repository of this tool a script can export the design to an Oracle database via ODBC. The checker program works on this database, which makes the navigation of the model easier. The specification completeness checking was implemented in Prolog. The rules mentioned above are expressed as Prolog questions. Our implementation has an interface toward SQL, the language used to handle the database. Accordingly, every Prolog question is converted into SQL commands. The interface sends these to the Oracle server, and converts the answers into Prolog predicates.

Up to now, the checker was successfully used to examine the completeness and consistency of several (small scale) UML designs.

****

Application-level semi-on-line monitoring of VisualMP applications on clusters of workstations

Norbert Podhorszki

A new application-level, software tracing monitor is designed and implemented for the VisualMP graphical parallel programming environment to support semi-on-line monitoring of message-passing programs in heterogeneous environments. We present the design aspects and implementation issues of the monitor.

VisualMP is a graphical programming environment integrating several tools to support the life-cycle of building parallel applications. Its major goal is to provide an easy-to-use, integrated set of programming tools for development of general message-passing applications to be run in heterogeneous computing environments. Its main benefits are the visual interface to define all parallel activities in the application, the syntax independent graphical definition of message passing instructions, full support of compilation and execution on heterogeneous environment and the integrated use of the debugger and performance visualiser. Tools of the VisualMP program development environment are the GRAPNEL graphical parallel programming language, GRED graphical editor to write parallel applications in GRAPNEL, the GRP2C pre-compiler to produce the C code with PVM or MPI function calls from the graphical program, the DIWIDE distributed debugger, the PROVE execution and performance visualisation tool and the GRM distributed monitor. For detailed overview of the tools of VisualMP, see [1] and [2]. PROVE is presented in [4].

We believe that post-mortem analysis of long traces is not a practical way to evaluate and improve an application. When several millions of events are visualised at once no one can effectively use it to focus on performance problems. Moreover, a post-mortem visualiser is not capable of visualising very large trace data. Practical ways of performance evaluation of long-running programs can be on-line visualisation, presentation of statistics for the whole execution and the use of automatic performance analysis tools. Having already a trace visualiser we decided to explore the possibilities of the first two methods, keeping in mind the possible use of automatic tools in the future.

In off-line monitoring, trace events are stored in local or global storages (memory and files) and are processed after execution. In on-line monitoring trace events are sent immediately to a tool that processes (visualises or evaluates) them. Instead of sending each individual trace event to the tool, events can be buffered in local storage and collected into a global trace only when they are needed for processing. We call this method semi-on-line monitoring (and visualisation). One end of this method is off-line monitoring (i.e. buffering is used and traces are collected only when the application is finished). The other end of this method is on-line monitoring (no buffer is used).

In order to support the examination of long running or cyclic applications PROVE is modified for semi-on-line visualisation. For this purpose a new distributed monitor - GRM - was designed and developed that supports trace collection and provides a consistent global time reference at any time during execution. During execution the user can force trace collection at any time and events locally buffered until that time are collected by the monitor and the trace is presented in PROVE. Collection can be started either by the user, the application itself (with a special instrumentation function call), regularly by the visualisation tool or the debugger tool.

The monitoring is event-driven, both trace collection and counting are supported. The collection is fully software tracing and the instrumentation method is direct source code instrumentation. For a classification of monitoring techniques see [3]. The direct source code instrumentation is the easiest way of instrumentation. Since VisualMP keeps in hand the whole cycle of application building and source instrumentation is supported graphically we chose this option. The GRP2C precompiler inserts instrumentation function calls into the source code and the application process generates the event trace. Both trace collection and statistics are supported by the same monitor and the same instrumentation of the application. Trace collection is needed to give data to PROVE for execution and performance visualisation. Statistics have less intrusion to the execution by generating fixed amount of data and they support initial evaluation of long-running programs.

The main goals in the design of a new monitor were strongly related to the VisualMP environment. They were:

1. Support of monitoring and visualisation of programs on GRAPNEL level. 2. It is part of an integrated development environment. 3. Portability. Use on heterogeneous clusters on UNIX operating systems (Irix, Solaris, Linux, DEC-Alpha, etc.) 4. Semi-on-line monitoring and visualisation to

support - evaluation of long-running programs, - debugger in VisualMP with execution visualisation. 5. Dynamic instrumentation to support - evaluation of long-running programs, - automatic performance analysers integrated into VisualMP in the future. 6. Both statistics and event trace collection should be supported. 7. Trace data should not be lost at program abortion. We want to visualise the execution to the point of abortion. 8. The execution of the application and the development environment should be separate. Thus, an application can be developed on a local host while it is executed on a remote cluster (and it is visualised semi-on-line on the local host).

The above goals simplified the design of the monitor in several aspects. The tight integration of GRM into VisualMP enabled us to put several functionalities of a stand-alone monitoring tool into other tools of VisualMP. For example, - instrumentation is naturally done in the graphical editor, - trace events are generated only by instrumentation function calls in the application processes not by the monitor - trace events of different processes are not sorted into time order since the preprocessing phase in PROVE does not need a globally sorted trace, - local monitors are started on the hosts defined by the environment, - the monitor is started and stopped by GRED, - the monitor does no bookeeping of processes, it does not even recognise the finish of the application.

The design goals gave constraints on the monitor, too. The high portability requirement forced us to use only standard UNIX solutions for every problems related to monitoring. Operating system or hardware specific solutions, counters or performance analysis tools must have been avoided. Fortunately, keeping us only on GRAPNEL level we could implement GRM in standard UNIX.

If trace events generated by a process are stored in local memory they can be lost when the process aborts. GRM should use shared-memory segments for trace storage in order to keep trace events accessible to VisualMP to the last point of execution. The use of shared-memory segments makes also possible the dynamic instrumentation of the program without modifying the application program code. However, this solution increases the intrusion since a new race condition is introduced by the use of a shared object.

Three basic problems arise when an application executed in a distributed environment is monitored. They are clock synchronisation, handling large amount of trace data and intrusion. This paper discuss these problems and the answers of GRM for the challenges.

References

[1] P. Kacsuk, G. Dózsa, T. Fadgyas, and R. Lovas: "GRADE: a Graphical Programming Environment for Multicomputers", Journal of Computers and Artificial Intelligence, Slovak Academy of Sciences, Vol. 17, No. 5, 1998, pp. 417-427

[2] P. Kacsuk: "Macrostep-by Macrostep Debugging of Message Passing Parallel Programs", PDCS'98, Los Angeles, 1998, pp. 527-531

[3] J. Chassin de Kergommeaux, E. Maillet and J-M. Vincent: "Monitoring Parallel Programs for Performance Tuning in Cluster Environments", In "Parallel Program Development for Cluster Computing: Methodology, Tools and Integrated Environments" book, P.Kacsuk and J.C.Cunha eds, Chapter 6., will be published by Nova Science in 2000.

[4] P. Kacsuk: "Performance Visualization in the GRADE Parallel Programming Environment", accepted for HPCN Asia, Beijing, China, 2000.

Congruences on Finite Automata

Zarko Popović

For a class $\boldsymbol{C}$ of algebras and an algebra A, a congruence relation ϱ on A is called a *$\boldsymbol{C}$-congruence* on A if the factor algebra A/ϱ belongs to $\boldsymbol{C}$. A class $\boldsymbol{P}$ of finite algebras is called a *pseudovariety* if it is closed under formation of subalgebras, homomorphisms and finite direct products. It is known (see [1] and [4]) that if $\boldsymbol{P}$ is a pseudovariety and A is any algebra, then the set $\mathrm{Con}_{\boldsymbol{P}}(A)$ of $\boldsymbol{P}$-congruences on A is a filter of the congruence lattice $\mathrm{Con}(A)$ of A, so any finite algebra has the *least $\boldsymbol{P}$-congruence*. The main aim of this paper is to find the least congruences which correspond to certain particular psudovarieties of finite automata.

An automaton A is called *directable* if there exists an input word which takes all states of A into one single state, it is *trapped* if there exists an input word which takes any state of A into some trap of A, it is *trap-directable* if it is directable and has a trap, it is *locally trap-directable* if any monogenic subautomaton of A is directable, and it is *generalized directable* if there exists an input word u such that $au = 3Dauvu$, for every state $a \in A$ and every input word v. Imreh and Steinby in [6], 1995, determined the least $\boldsymbol{Dir}$-congruence on a finite automaton, where $\boldsymbol{Dir}$ denotes the pseudovariety of *directable automata*. Here we shall determine the least congruences corresponding to the classes $\boldsymbol{Trap}$, of trapped automata, $\boldsymbol{TDir}$, of trap-directable automata, $\boldsymbol{LDir}$, locally directable automata, $\boldsymbol{GDir}$, generalized directable automata, etc. These classes were introduced and studied by Petković, Ćirić and Bogdanović in [9], 1998, and more information about them can be also found in the survey article by Bogdanović, Imreh, Ćirić and Petković [2].

One of the starting point of our investigation is the result due to Kovacević, Ćirić, Petković and Bogdanović [7], which says that every finite automaton can be uniquely represented as an extension of a reversible automaton by a trap-directable automaton, where a *reversible* automaton is the one which can be represented as a direct sum of sum of strongly connected automata. Using this result, we prove the following theorems.

Theorem 7 *Let a finite automaton A be represented as an extension of a reversible automaton B by a trap-directable automaton, where B is represented as a direct sum of strongly connected automata $B_\alpha, \alpha \in Y$. Then the relation τ on A defined by*

$$(a,b) \in \tau \;\Leftrightarrow\; a = 3Db \;\; or \;\; (a,b) \in B_\alpha, \textit{ for some } \alpha \in Y,$$

is the least $\boldsymbol{Trap}$-congruence on A.

Theorem 8 *Let a finite automaton A be represented as an extension of a reversible automaton B by a trap-directable automaton C. Then the Rees congruence ρ_B on A determined by B is the least $\boldsymbol{TDir}$-congruence on A.*

The least $\boldsymbol{GDir}$-congruence on a finite automaton is characterized in terms of the least $\boldsymbol{Dir}$-congruence, described by Imreh and Steinby in [6].

Theorem 9 *Let a finite automaton A be represented as an extension of an automaton B by a trap-directable automaton, let B be represented as a direct sum of strongly connected automata $B_\alpha, \alpha \in Y$, and for each $\alpha \in Y$ let θ_α be the least $\boldsymbol{Dir}$-congruence on B_α. Then the relation θ on A defined by*

$$(a,b) \in \theta \;\Leftrightarrow\; a = 3Db \;\; or \;\; (a,b) \in \theta_\alpha, \textit{ for some } \alpha \in Y,$$

is the least $\boldsymbol{GDir}$-congruence on A.

If $\boldsymbol{P} \subseteq \boldsymbol{Dir}$ is a pseudovariety, then the class $L(\boldsymbol{P})$ of all finite automata whose any monogenic subautomaton belongs to $\boldsymbol{P}$ is also a pseudovariety, and it consists of automata which are direct sums of automata from $\boldsymbol{P}$ (see [3] and [8]). Using this fact and a theorem proved by Ćirić and Bogdanović in [5], 1999, which says that every automaton can be uniquely represented as a direct sum of direct sum indecomposable automata, we prove the following theorem.

Theorem 10 *Let $\boldsymbol{P} \subseteq \boldsymbol{Dir}$ be a pseudovariety, let a finite automaton A be represented as a direct sum of direct sum indecomposable automata $A_\alpha, \alpha \in Y$, and for each $\alpha \in Y$ let θ_α be the least $\boldsymbol{P}$-congruence on A_α. Then the relation θ on A defined by*

$$(a,b) \in \theta \;\Leftrightarrow\; (a,b) \in \theta_\alpha, \text{ for some } \alpha \in Y,$$

is the least $L(\boldsymbol{P})$-congruence on A.

References

[1] S. Bogdanović and M. Ćirić, *A note on congruences on algebras*, in: Lj. D. Kočinac (ed.), Proceedings of the II Mathematical Conference in Priština (1996), Univ. Priština, Priština, 1997, pp. 67–72.

[2] S. Bogdanović, B. Imreh, M. Ćirić and T. Petković, *Directable automata and their generalizations* (*A survey*), Proc. VIII Inter. Conf. "Algebra and Logic", (Novi Sad, 1998), Novi Sad J. Math. **29**(3) (1999), 25–68.

[3] S. Bogdanović, M. Ćirić, T. Petković, B. Imreh and M. Steinby, *Local properties of unary algebras and automata*, (to appear).

[4] M. Ćirić and S. Bogdanović, *Posets of $\mathcal{C}$-congruences*, Algebra Universalis **36** (1996), 423–424.

[5] M. Ćirić and S. Bogdanović, *Lattices of subautomata and direct sum decompositions of automata*, Algebra Colloquium **6**:1 (1999), 71–88.

[6] B. Imreh and M. Steinby, *Some remarks on directable automata*, Acta Cybernetica **12**, (1995), no. 1, 23–35.

[7] J. Kovačević, M. Ćirić, T. Petković and S. Bogdanović, *Decompositions of automata and reversible states*, in: A. Adam and P. Dömösi (eds.), Proceedings of the Nineth International Conference on Automata and Formal Languages, Publ. Math. Debrecen (submitted).

[8] T. Petković, *Varieties of automata and semigroups*, Ph. D. Thesis, University of Niš, 1998, (in Serbian).

[9] T. Petković, M. Ćirić and S. Bogdanović, *Decompositions of automata and transition semigroups*, Acta Cybernetica (Szeged) **13** (1998), 385–403.

[10] Ž. Popović, S. Bogdanović, M. Ćirić and T. Petković, *Generalized directable automata*, (to appear).

[11] Ž. Popović, S. Bogdanović, T. Petković and M. Ćirić, *Trapped automata*, in: A. Adam and P. Dömösi (eds.), Proceedings of the Nineth International Conference on Automata and Formal Languages, Publ. Math. Debrecen (submitted for publication).

Extension of Java with Turtle Graphics

Zoran Putnik and Marijana Grabovac

Computer graphics has entered all aspects of computer usage, especially after introduction of World Wide Web into everyday life and after the number of people who are using it increased. This emphasized the need for better quality and enlarged quantity of graphics we are faced with. At the same time, WWW induced new and powerful concept - programming language Java. This programming language, together with technical and technological improvements, brought the possibility of inclusion of larger amount of multimedia elements and network into all kinds of software.

On the other hand since the invention of programming language LOGO, turtle graphics has been recognized as an excellent educational tool. It is especially suitable for a development of procedural thinking, offering more natural way for unification of a program with data structures and serving as a ready-made prototyping tool. The language was designed at Bolt Beranek and Newman INC, in Cambridge, Massachusetts, USA, in the late 1960s [1]. The real reason for its development was testing of an idea that programming may be used as an educational discipline to teach children, especially about notions hard for understanding (mathematical concepts, experiments with abstract concepts).

In any programming language, there are two basic ways of employing computer graphics: usage of basic graphic functions and usage of turtle graphics. While the use of basic graphics functions confirms essentially to use of (any) other (kind of) functions of a given language, turtle graphics, less known and used, has its own "rules" and techniques, common for every language. Traditionally and informally, turtle graphics may be described as a process of picture creation through a movement of "an animal" (i.e. a turtle), which crawls across a screen, towing a pencil and leaving a trail [1].

Turtle graphics has been explored at the Institute of Mathematics in Novi Sad for some time, and it was at first introduced as an extension of functional programming language LispKit LISP developed at the Institute [2]. Later, research concentrated on educational values of turtle graphics in [3]. As it may be noticed, at the time, study focussed on combination of functional programming and turtle graphics. Yet, with the introduction and expansion of Java programming language, some problems existant in mentioned research, related with functional programming are naturally overcame.

Now, turtle graphics has been implemented in Java programming language as one Java applet. The implementation of turtle graphics in Java is important because it brings the expressiveness of LOGO to Java language. At the same time, implementation of turtle graphics in Java programming language implicitly introduced it into several other operating systems. Java applets, with easy control of mouse, buttons, "choices" and authority over the computer even for non-programmers enabled greater concentration on abilities of turtle graphics and their use in education.

The choice of Java programming language proved as an excellent one, since we got a language that is portable, interpreted, high-performance, simple and object-oriented, which are all characteristics of a modern programming language of a high quality.

In this paper we have implemented something that has already proved its educational and creative value. In this way we have presented it to the wider public in the environment that is secure and user friendly. Since the program was carried out in Java it can operate on any platform (operating system) without the need for recompilation or reimplementation.

References

[1] Ross, P. *LOGO Programming for the IBM OC*, Addison Wesley, Wokingham.

[2] Putnik, Z., Budimac, Z., and Ivanovic, M. *Turtle Walk Through Functional Language*, SIGPLAN Notices, Vol. 26, no. 2, pp 75-82.

[3] Putnik, Z., and Budimac, Z., *Bringing LOGO to the World of Purely Functional Programming*, In Proc. of 'EUROLOGO '97' - Conference, Budapest, Hungary, pp. 406-409.

Formal Grammars in Conformance Testing

Csaba V. Rotter

This paper is a continuation of the previous CSCS conference material where the conformance test notation and the formal grammars was compared. A communication protocol can be considered as a black box from the point of view of testing. An unambiguous behaviour model can produce a theoretically, well based solution.

The paper gives a new approach in conformance test modeling. Protocol testing is the practical way to check correctness of protocol implementation with respect to their specification. This checking needs many different test cases. These test cases can be made by hand or can be generated automatically by help of different tools. The new idea is to generate the test cases from formal grammar description of protocols. This method consists of three main steps:

- Formal grammar from a specific protocol description (SDL, MSC ...)
- Formal grammar based test language from formal grammar protocol language
- Formal test language (TTCN) from formal grammar test language

In the first step the goal is to derive a formal grammar notation from SDL (Specification and Description Language) or MSC (Message Sequence Charts). The mapping forms of the grammars are the followings; N–>N1*T; or N–>T; (Chomsky Class 3 grammars). In this way we can obtain a tree of the protocol messages. This tree has one starting point and many end points. The sentences between the starting point and the endpoints correspond to test purposes.

In the second step these sentences are generated. The result of this generation is a "test language".

The third step is to derive a TTCN (Tree and Tabular Combined Notation) from the previously generated sentences. The insufficiency of this method is that the data part actually can not be generated automatically: the solution of this problem will be the next steps.

References

[1] Arto Salomaa: Computation and Automata, Cambridge University Press 1985

[2] Harangozó J.: The Description of Data Link Protocols for Computer Networks Using Formal Languages, Fifth Data Comm. Symp., Snowbird, Utah, 1977

[3] E. Csuhaj-Varju, J. Dassow, V. Mitrana, Gh. Paun: Grammar Systems. A Grammatical Approach to Distribution and Cooperation, Gordon and Breach, London, 1994

[4] E. Csuhaj-Varju, J. Dassow, V. Mitrana, Gh. Paun; Cooperation in grammar systems: similarity, universality, timing, Cybernetica, 4 (1993), 271-282

[5] K. Tarnay: Protocol Specification and Testing, Plenum Press New York 1991

[6] Boja Zsuzsa Harangozó, Mária Törő: TTCN and a Hungarian software, Selected Paper on Hungarian Telecommunication 1996, pp.25-30

[7] Mária Törő, K. Tarnay: Principles for validation of abstract test suites specified in concurrent TTCN PSTV'95 Warsaw, Poland, 13-16 June 1995

[8] Wireless Application Protocol Wireless Transaction Protocol Specification, Version 26-January-2000 http://www.wapforum.org/

Dynamic Simulations

Levente Sára

The subject of our research is to develop models, which consist of several parallel processes. In these models there exist great number of effects and counter-effects. Changing of some values in the system indicates unpredictable side-effects and feedbacks. These effects can not be managed by classical mathematical tools and methods, because they are highly non-linear. Dynamics of such systems can easily be followed with simulations.

We have developed a program to create and develop dynamics models quickly and easily by using visualization tools. During the process of a simulation certain values can be selected to see how they have changed. The program uses various visualization tools to realize the trends or principles which control the actions and "behavior" of the whole model. We use hyperbolic like screen view, which represents the model of a graph. The main advantage is that we are not forced to use a specific simulation description language to build the model, and so the developing of a new model is extremely fast. Data (the nodes of the graph) and the relations between the nodes can be created. We can define our own functions to describe the nodes. Multi-valued logical operators are extremely efficient. The program automatically creates and manages the most often used functions, such as average, weighted average, sum, product etc. Weight and speed of the effects are represented by the relation with assigned functions.

The program helps to analyze the simulation results by visualization tools. In this way, you can see the speed and orientation of processes in diagrams, graphs and colors instead of endless columns of numbers.

Dynamic simulation models can be applied in almost every field of life: from the inside processes of an enterprise or modeling of the market grow, through demographic and social changes affecting a whole country until studying the cycle of atmosphere or researching the interactions of ecological situations. E.g. in 1993 the periodical "System Dynamics Review" published an article about the agricultural investments and their effects in South Sudan. The author introduced a dynamic model of the rural life of Sudan. There are other studies on the inherences of the CO-concentration in the air and the tropical forest-fires using possible dynamic models. We implemented these examples to improve our program.

Test of inter-working and translation mechanisms between IPv4 and IPv6 [12]

G. Somlai, L. Sógor, L. Martonossy, M. Fidrich, G. Dikán,
T. Tarjányi, P. Hendlein and M. Bohus

The Internet Protocol version 4 (IPv4) has been available since 1981. As it can be seen today, its success is indiscutable. However, the rapid growth of the Internet has created a number of problems for the administration and operation of the global network. Some new challenges that must be handled have also appeared, such as security, mobility and real-time support. The new version 6 of the Internet Protocol (IPv6) includes expanded addressing capabilities, header format simplification, improved support for extensions and options, plug and play services, authentication and privacy capabilities, native mobility support, and also real-time and quality services.

Maintaining compatibility with the large installed base of IPv4 hosts and routers while deploying IPv6 will streamline the task of transitioning the Internet to IPv6. There are two main cases of interoperation between IPv4 and IPv6, which should be solved. The first is that of running an IPv6 network that communicates with other IPv6 networks using the existing IPv4 infrastructure. The solutions for this case (configured and automatic tunneling, 6over4, 6to4) have already been cleared out, most of the mechanisms has been defined in RFCs. The second case is that of IPv4 and IPv6 hosts that want to communicate with each other directly. This communication is very complicated mainly because of the different addressing schemes and it can be solved using translation techniques at one of three different protocol levels. At the IP level, Network Address Translation - Protocol Translation (NAT-PT) employing the Stateless IP/ICMP Translator (SIIT) mechanism can be used. Relaying of UDP and TCP sessions can be done at the transport level, while application proxies such as SOCKS run at the application level.

This article presents the methods used and results obtained in testing the communication among: i) IPv6 hosts in different IPv6 networks, which are interconnected through IPv4 infrastructure and ii) an IPv6 host on an IPv6 network and an IPv4 host on an IPv4 network. For the first the *6to4* method and *configured tunneling* was used, while for the second one the *SIIT* and a *simplified version of NAT-PT*. For 6to4 and SIIT we had to develop our own pieces of software. Different configurations were used, consisting of IPv6, IPv4 and IPv4/IPv6 hosts, interconnected in various ways.

Due to its free source code and good documentation, the Linux operating system was chosen as a test platform. The nature of the tasks to be resolved and the solutions chosen required implementation at the kernel level. The tunneling procedures in the case of 6to4 and header conversion procedures necessary for NAT-PT were implemented as kernel modules, which can be loaded/unloaded dynamically with the modutils and configured by the ifconfig utilities. These modules create pseudo network devices, which receive and send the packets they need to process using standard routing procedures. The necessary routing rules can be given with the route utility. All utilities used are part of standard Linux distributions.

Due to the lack of implementation of some quite basic features of IPv6 (which we rely on) at this stage of the development of the Linux kernel, some small patches to the kernel were also needed. Hopefully - maybe based on our suggestions - later versions of the kernel will include these changes.

Several tests were performed to find out the performance of the methods, and, additionally, the conformance and robustness of our implementation. Results and the conclusions we drew are presented.

[12] This article presents the results of a research work initiated and financed by Nokia Hungary as a joint project between the University of Szeged and Nokia Hungary.

Case study of SIRDS algorithms based autostereoscopic image

Tran Minh Son, Gyula Marosi and Dr. András Gschwindt

The environment around us is the world of three-dimension (3D); i.e. the position of everything is precisely described by its coordinates in the three-dimensional Descartes systems. However, up to now the two-dimensional (2D) presentation of objects is still deployed in most of image processing applications, which tend to simulate, to reconstruct the real world at the maximum level of reality. Obviously such methods with reduction in dimension cause data-loss, but they are still dominated because of their simplicity and portability. Recently the drastical development of computer and hardware manufacturing makes 3D displaying system (therefore application) possible at a quite "reasonable" price. A lot of prospective results emerged in this field but the question to looking for the displaying method with the best trade-off between the reality of 3D world and complexity viability is till open for researchers.

This paper provides a deep study for a displaying method based on SIRDS (Single Image Random Dots Stereogram) algorithms, which can be considered as a good solution for the mentioned question. What SIRDS algorithms is, how to display a SIRDS based image, its limit and visibility are discussed in this paper. Besides helping readers have a comprehensive view throughout different programming techniques of SIRDS implementations, it proposes a new one to overcome some limits of predecessors. Some practical trials toward a new aspect of SIRDS - applied to motion objects – are also take into account in the frame of this paper.

References

[1] Tran Minh Son: Single Image Random Dot Stereograms, Student Scientific Conference, Technical University of Budapest, Hungary, 1997.

[2] Tran Minh Son, Marosi Gyula: Guide and instructions for "Sztereoszkópikus képmegjelenítés, SIRDS alapú autosztereogramok" laboratory, Technical University of Budapest, Hungary, 1999.

[3] Horold W. Thimbleby, Stuart Inglis, Ian H. Witten: Display 3D Images: Algorithms for Single Image Random Dot Stereograms.

[4] Ian Sexton, Phil Surman: Stereoscopic and autostereoscopic display systems, IEEE Signal Processing Magazine, May 1999.

Optimal disk packings in the square

P.G. Szabó, B. Tóth and T. Vinkó

To find the densest packing for n equal and non-overlapping circles in a square, is a well-known problem in discrete geometry. An equivalent continuous global optimization problem to locate n points in a square, such that the minimal distance m between any two of them is maximal. Currently the optimal solutions of this problem are known for the 1-27 and 36 circles cases. Good packings, the best known arrangements, are available up to 200 circles (see e.g. at

`http://www.inf.u-szeged.hu/~pszabo`).

In our talk, we give a short review on the previous results of the problem and some new patterns and similar structures of packings found by a new stochastic algorithm. For all packings a corresponding minimal polynomial can be defined, the smallest positive root of which is m, and the degree of the polynomial is minimal. The degrees and coefficients of these polynomials indicate how difficult it is to find the respective solutions of the packing problem. We have calculated some new minimal polynomials of circle packing problems.

On the Scalability of Multidimensional Databases

István Szépkúti

It is commonly accepted in the practice of on-line analytical processing databases that the multidimensional database organization is less scalable than the relational one. It is easy to see that the size of the multidimensional organization may increase very quickly. For example, if we introduce one additional dimension, then the total number of possible cells will be at least doubled.

However, this reasoning does not take into account that the multidimensional organization can be compressed. There are compression techniques, which can remove all or at least a part of the empty cells from the multidimensional organization, while maintaining a good retrieval performance.

Relational databases often use B-tree indices to speed up the access to given rows of tables. It can be proven, under some reasonable assumptions, that the total size of the table and the B-tree index is bigger than a compressed multidimensional representation. This implies that the compressed array results in a smaller database and faster access at the same time.

This paper compares several compression techniques and shows when we should and should not apply compressed arrays instead of relational tables.

Parallel verification and enumeration of tournaments

László Szűcs and Gábor Pécsy

Round-robin tournaments are popular in the world of sport and they are very much discussed in informatics as well. A round-robin **tournament** is an $n \times n$ real matrix $T_n = [t_{ij}]$ $(n \geq 2)$. The elements of the main diagonal t_{ii} equal to zero and the pairs of symmetric elements $t_{ij} : t_{ji}$ give the result of the match between P_i (the i-th player) and P_j. $t_{ij} = t_{ji}$ means a draw, while $t_{ij} > t_{ji}$ means the win of P_i against P_j. The sum of the elements of the i-th row (s_i) is called the **score** of the i-th player and the vector $(s_1, \ldots, s_n)$ is called the **score vector** of the tournament. A non-decreasingly ordered vector of the scores is denoted $q =< q_1, \ldots, q_n >$ and is called the **score sequence** of the tournament. There are special subsets of tournaments defined by different constraint on the possible results of the matches. We call a set of tournaments **k-complete** if in all matrices all elements are non-negative integers and the sum of the symmetric elements $(t_{ij} + t_{ji},$ where $i \neq j)$ is always k.

The most usually discussed problems regarding tournaments include:

- **Verification of a score sequence/score vector** means the decision if there exists a tournament for a given score sequence/score vector.
- **Enumeration of score sequences** means the counting of the possible different score sequences for a given n number of players.

Several algorithms has been published to solve the above mentioned problems for different subsets of tournaments. One of the most extensively discussed subset is the 1-complete. In this article we present optimal sequential algorithms for all three problems on 1-complete tournaments. We also present their extensions to different parallel architectures including CREW PRAM, linear array, mesh and hypercube. We call a parallel algorithm **work optimal** compared to a given sequential algorithm if $S_n/(P_n * p) = O(1)$ where S_n is the runtime of the sequential algorithm, P_n is the runtime of the parallel algorithm and p is the number of processors. We will show that most of the parallel algorithms presented here are work optimal extensions of the sequential ones.

The table below summarises our results:

Problem	**Sequential**	**Linear array**	**Mesh**	**Hypercube**	**PRAM**
Score sequence	$\Theta(n)$	$\forall p \in \mathbb{N}$ $\Theta(n)$	$p = n$ $O(\sqrt{n})$	$p = \frac{n}{\log n}$ $\Theta(\log n)$ work opt.	$p = \frac{n}{\log n}$ $\Theta(\log n)$ work opt.
Score vector	$\Theta(n)$	$\forall p \in \mathbb{N}$ $\Theta(n)$	$p = n$ $O(\sqrt{n})$	$p = n$ $O(\log^2 n)$	$p = n$ $O(\log^2 n)$
				$p = n^2$ $O(\log n)$	$p = n^2$ $O(\log n)$
Enumeration of score sequences	Recursive formula with dynamic programming: $\Theta(n^4)$	$p = n$ $\Theta(n^3)$ work opt.	$p = n$ $\Theta(n^3)$ work opt.	$p = n$ $\Theta(n^3)$ work opt.	$p = n$ $\Theta(n^3)$ work opt.
		$p = n^2$ $\Theta(n^2)$	$p = n^2$ $\Theta(n^2)$	$p = n^2$ $\Theta(n^2)$	$p = n^2$ $\Theta(n^2)$
				$p = \frac{n^4}{\log n}$ $O(n * \log n)$	$p = \frac{n^4}{\log n}$ $O(n * \log n)$

Investigation of Point–Based Image Registration Methods Assuming Rigid–Body and Linear Motions

Attila Tanács

Image registration is used to match two independently acquired images.

The geometrical transformation is to be found that maps a *floating image* in precise spatial correspondence with a *reference image*. Point–based registration requires the matching of a set of 3D points in the reference image with a homologue set of 3D points in the floating image. We call these points used for registration fiducials. Fiducial points can be obtained from external fiducial markers or from internal anatomical landmarks.

Registration techniques involve searching over the space of transformations of a certain type to find the optimal one–to–one mapping for a particular problem. The major classes of 3D transformations are *rigid–body transformation* when only translations and rotations are allowed, *affine transformation*, which maps parallel lines onto parallel ones, and *nonlinear*, *curved* or *elastic transformation* i.e., which maps straight lines onto curves.

Point–based registration might find imperfect matching due to the presence of error in localizing the fiducials. There are some papers dealing with the analysis of point–based registration. Emphasis is to be put that each of these papers assumes only rigid–body transformation. Maurer et al. [8] proposed three types of measures of error: *Fiducial localization error* (FLE), which is the error in determining the positions of the fiducials, *fiducial registration error* (FRE), which is the root mean square distance between corresponding fiducials after registration, *target registration error* (TRE), which is the distance between corresponding points representing surgical targets after registration. Note that point-based registration methods minimize FRE. But using FRE as measure of registration accuracy is unreliable and may be misleading, thus investigations were focussed on TRE in the last decade [4, 9].

There are two important results concerning rigid–body registration errors:

- **Result 1.** For a fixed number of fiducials, TRE is proportional to FLE [3, 4, 7, 9],
- **Result 2.** TRE is approximately proportional to $1/\sqrt{n}$ with n being the number of fiducials [4, 6, 9].

It is obvious that TRE is sensitive to the location of target points and the location of the fiducials. Maurer et al. [9] used numerical simulations to give qualitative insight into this dependence. In addition, they realized that TRE depends on the fiducial configurations and the distances between fiducials via the examinations of four configurations of four fiducials.

Fitzpatrick et al. [4] gave exact expression for approximating TRE assuming rigid–body registration problem, thus proving **Result 1** and **Result 2**, and answering how does TRE depend on the relative positions of target and fiducial points. They also stated that TRE is inversely proportional to the scale factor used for scaling fiducials.

We investigate the following four point–based registration methods:

- RB1

  The solution of Arun et al. [1] is implemented for determining a rigid–body transformation given by a 3×3 rotation matrix and a translation vector (3×1 matrix). The rotation matrix is computed using the singular value decomposition (SVD) of the covariance matrix of the centroid–subtracted position vectors of the corresponding fiducials. The translation vector is calculated as the difference between the centroids of the two sets of points. This method is used by several authors [5, 9, 11, 14]. It is regarded as the most popular point based registration approach.

- RB2

  Our implementation utilizes the Levenberg–Marquardt technique [10] for finding the 6 parameters of rigid–body transformations. There is a possibility that this iterative minimization might fail because of local minima in the parameter space. This method is used by Zuk et al. [13], too.

- LIN
  The method proposed by Tanács et al. [12] allows more freedom. It is to find affine transformations given by the 12 unknown elements of 4×4 transformation matrices (using homogeneous coordinates).
- TPS
  We implemented the thin–plate spline interpolation proposed by Bookstein [2]. It is capable of finding nonlinear transformations given by $3 \cdot (n + 4)$ parameters with n being the number of fiducials. Note that the complexity of reslicing (i.e., applying the found transformation for the reslice image) does not depend on n in the case of the other three considered methods. It is not true for the thin–plate spline warping: the required time of reslicing is proportional to n.

Numerical simulations are made assuming both rigid–body and affine motions.

We extended the observations according to FLE and TRE for point based registration methods with different search spaces, assuming not only rigid–body but affine motions, as well. Our examinations confirm **Result 1** for all the four methods, and **Result 2** for rigid–body methods. Another result in this work is that TRE depends on the volume spanned by the fiducials:

- **Result 3.** TRE is inversely proportional to the mean distance from the centre of gravity.

We compared two rigid–body methods RB1 and RB2. Since RB2 is an iterative method, its result depends on the initial guess and numerical errors are accumulated. In some cases it may diverge. That is why using the direct method (RB1) is recommended.

If there is a precise rigid–body correspondence between the two sets of points used for registration, then any linear or nonlinear method is able to find the adequate transformation, since a rigid–body transformation is a special kind of linear or nonlinear one. It is not hold in real registration problems due to the presence of FLE. It can be stated that RB1 and RB2 (i.e., the methods restricted to rigid–body transformations) are better than LIN and TPS (i.e., the methods which allow more freedom) if a rigid–body transformation is assumed. Similarly, LIN is better than TPS if a linear motion is assumed. That is because transformations having more degrees of freedom than necessary can incorporate a wider range of fake motions induced by FLE.

Acknowledgements This work was supported by OTKA T023804 and FKFP 0908/1999 Grants.

References

[1] Arun, K.S., Huang, T.S., Blostein, S.D.: Least–Squares Fitting of Two 3–D Point Sets. IEEE Trans. on Pattern Analysis and Machine Intelligence **9** (1987) 698–700

[2] Bookstein, F.L.: Principal warps: thin–plate splines and the decomposition of deformations. IEEE Trans. on Pattern Analysis and Machine Intelligence **11** (1989) 567–585

[3] Evans, A.C., Marrett, S., Collins, D.L., Peters, T.M.: Anatomical–functional correlative analysis of the human brain using three dimensional imaging systems. In: Medical Imaging: Image Processing, vol. Proc. SPIE 1092 (1989) 236–246

[4] Fitzpatrick, J.M., West, J.B., Maurer, C.R.: Predicting error in rigid–body point-based registration. IEEE Trans. on Medical Imaging **17** (1998) 694-702

[5] Hill, D.L.G.: Combination of 3D Medical images from Multiple Modalities. Ph.D. Thesis, Image Processing Group Radiological Sciences UMDS, London (1993)

[6] Hill, D.L.G., Hawkes, D.J.: Medical image registration using knowledge of adjacency of anatomical structures. Image and Vision Computing **12** (1994) 173–178

[7] Mandava, V.R., Fitzpatrick, J.M., Maurer, C.R., Maciunas, R.J., Allen, G.S.: Registration of multimodal volume head images via attached markers. In: Medical Imaging: Image Processing, vol. Proc. SPIE 1652 (1992) 271–282

[8] C.R. Maurer, J.J. McCrory, J.M. Fitzpatrick, Estimation of accuracy in localizing externally attached markers in multimodal volume head images. In: Medical Imaging: Image Processing, vol. Proc. SPIE 1898 (1993) 43–54

[9] Maurer, C.R., Fitzpatrick, J.M., Wang, M.Y., Galloway, R.L., Maciunas, R.J., Allen, G.S.: Registration of head volume images using implantable fiducial markers. IEEE Trans. on Medical Imaging **16** (1997) 447–462

[10] Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T.: Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press, second edition (1992)

[11] Strasters, K.C., Little, J.A., Buurman, J., Hill, D.L.G., Hawkes, D.J.: Anatomical landmark image registration: validation and comparison. In: CVRMed–MRCAS'97, Lecture Notes in Computer Science, Vol. 1205. Springer-Verlag, Berlin Heidelberg New York (1997) 161–170

[12] Tanács, A., Palágyi, K., Kuba, A.: Medical image registration based on interactively defined anatomical landmark points. Int. J. Machine Graphics & Vision **7** (1998) 151–158

[13] Zuk, T.D., Atkins, M.S.: A Comparison of manual and automatic methods for registering scans of the head. IEEE Trans. on Medical Imaging **15** (1996) 732–744

[14] West, J., Fitzpatrick, M., Wang, M.Y., et al.: Comparison and evaluation of retrospective intermodality image registration techniques. In: Medical Imaging: Image Processing, Proc. SPIE 2710 (1996) 332–347

Generalised Subspace based Method for Steady State Analysis of QBD-M Processes

Hung T. Tran and Tien V. Do

The search of numerical solutions for queueing models in the performance analysis of computer systems and communication networks has practical importance and is always a hot research topic. In recent years, the extensive application of two-dimensional queueing systems which state is described by a phase and a level has been witnessed, particularly on the performance evaluation of ATM (Asynchronous Transfer Mode) systems. If the level transitions are only possible between adjacent ones, such queueing systems are called QBD (Quasi Birth-Death) processes. For the steady state analysis of this class of two-dimensional Markov chains several efficient methods have been developed and improved over recent years. For example, one can mention the method of Naoumov et al., the spectral expansion method, the ETAQA method and many others [1, 2, 3, 4].

However, there are only a few of works dealing with such two-dimensional queueing systems in which upper-bounded batches (either arrival or departure) occur, i.e. multiple jumps in level dimension are possible. Such systems (referred as QBD-M processes) can be considered as an extension of QBD processes and therefore after some manipulations (e.g. re-blocking), computation methods developed for QBD processes can be applied. Nevertheless, from the technical point of view direct methods may have some advantages over the former ones and that's why it's worth devoting research efforts in this direction.

Towards to the aim addressed before, this paper proposes a direct computational method for computing steady state distribution of QBD-M processes. This method is based on the theory of generalized invariant subspace which has been first introduced and applied by Nail Akar et al. in [3, 5] to teletraffic problems. Moreover, the capability and applicability of this method will be discussed and demonstrated through a case study and will be compared with other existing computational ones, such as Naomouv's method and spectral expansion applied after re-blocking, the iterative method proposed in [6].

References

[1] Valeri Naoumov, Udo R. Krieger, and Dietmar Warner. Analysis of a Multi-Server Delay-Loss System With a General Markovian Arrival Process. *Matrix-analytic methods in stochastic models*, 1997.

[2] R. Chakka. *Performance and Reliability Modelling of Computing System Using Spectral Expansion*. PhD thesis, University of Newcastle upon Tyne, 1995.

[3] Nail Akar and Khosrow Sohraby. Finite and Infinite QBD Chains: A Simple and Unifying Algorithmic Approach. *in Proceedings of IEEE INFOCOM*, 1997.

[4] Gianfranco Ciardo and Evgenia Smirni. ETAQA: An Efficient Technique for the Analysis of QBD-processes by Aggregation. In *Proceeding of Performance '99*, 1999.

[5] Nail Akar, Nihat Cem Oguz, and Khosrow Sohraby. Matrix-geometric Solution of M/G/1 Type Markov Chains: A Unifying Generalized State-space Approach. *in IEEE JSAC*, 16, 1998.

[6] Hung T. Tran and Tien V. Do. A new iterative method for systems with batch arrivals and batch departures. In *Proceedings of Communication Networks and Distributed Systems Modeling and Simulation, CNDS'00*, San Diego, USA, 2000.

Time Independent Invocation in Java CMS[13]

Attila Ulbert and Markus Hof

Distributed object middleware, e.g. CORBA and Java RMI, hides the network details required in order to access remote objects and to invoke their methods. It offers an easy to use framework to develop distributed object oriented applications. However, many applications require - to some extent - access to some of these details. Many of these requirements can be satisfied with the help of appropriate invocation semantics. Therefore, many middleware platforms extend their definitions to include additional semantics, e.g. interceptors in COOL or filters in Orbix. This lead also to modifications to standards as, e.g. CORBA with its messaging service, Enterprise Java Beans with transactional semantics.

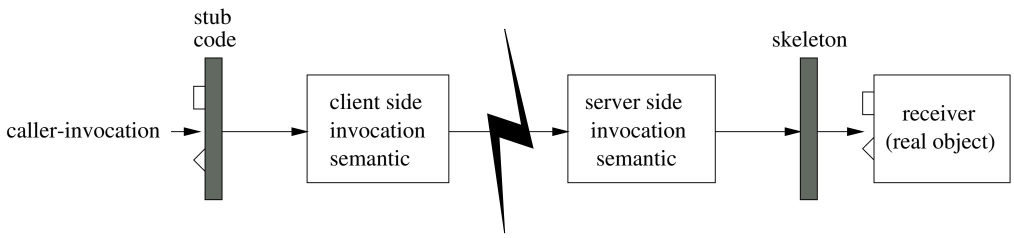

Figure 10: Client and server-side semantics

In our project we added a semantic for time independent invocations (TII) to the composable message semantics (CMS) framework. The CMS framework offers an alternative to Java RMI as a platform for remote method invocation. It allows the creation and composition of arbitrary new invocation semantics, e.g. replication, synchronization, asynchronous as well as arbitrary compositions of them. An invocation semantic consists of a server and a client part (see Figure 10). Both of these again are composed of smaller building blocks that are combined with the Decorator pattern. Every object can have its individual set of semantics. It is even possible to define more than one set of semantics for one and the same object. This allows the definition of multiple views on an object.

Other systems already define time independent invocations, e.g. the CORBA messaging service. A time independent invocation can be viewed as a fault-tolerant asynchronous invocation that is able to receive the results of the execution of the remote method at an arbitrary later point in time. This technique can help to the design and may simplify the implementation.

We have specified the implemented invocation semantics formally, as the informal specifications are often ambiguous. Our formal TII description makes the specification unambiguous and eases the design of the implementation. The complete implementation can be achieved by extending a simplified TII called oneway-TII (a fault-tolerant asynchronous invocation with parameters specifying the level of fault-tolerance).

With our CMS framework it is quite simple to use new invocation semantics. After the server instantiates and exports the object with the assigned server-side semantics, the client imports the object and assigns the desired client-side semantics. Afterwards, the methods of the imported object can be invoked as if they were methods of a local object. The following example depicts the assignment of a TII semantic to the method "set" and an invocation of that method that uses the newly assigned semantics.

Whenever the method "set" is called, the system transparently applies the assigned semantics and guarantees a fault-tolerant behaviour on the level specified by the two parameters of the TIInvocation constructor. In order to use the complete TII, we have to either supply a call-back method or a polling object. With the help of one of these two techniques we can access the results of the remote method invocation and can verify whether the remote invocation was successful or not.

```
Test testObj = new Test();

ClassInfo ci  = new ClassInfo(testObj);  // get Metainformation of class Test
```

[13]This research work was supported by Grant Nr. FKFP 0206/97

```
MethodInfo mi = ci.getMethod("set");
CallerInvocation inv = new TIInvocation(23, 100); // semantic with 23 retries
                                                  // every 100 milliseconds
mi.setCallerInvocation(inv);  // assign semantic to method "set"
testObj = (Test)Remote.get(adr, ci, "testObj");

testObj.set(....); // call methods as usual
```

Learning Decision Trees in Continuous Space

Ákos Zsiros

A lot of applications on the area of artificial intelligence and data mining leads to a similar task: constructing classification model based on our knowledge. A typical type of classification models is the decision tree, which is hierarchical system of decision rules.

The classical ID3 algorithm is based on attribute selection, where only discrete attributes can be used. Although it has a continuous extension (C4.5), it operates like ID3 using discretization of numerical values. There is an alternative way of building and pruning decision trees, in which it's not necessary to discretizate continuous values and we can handle arbitrary continuous attributes.

The geometric interpretation of classification is the following: separate the n-dimensional Euclidean space into regions and label each region with the class of its points. The C4.5-type algorithm uses only hyperplanes which are orthogonal to the axis to separate the space. In our new method we can use any other surfaces, for example hyperplanes, n-dimensional sphere, ellipsoid, etc.

The ID3-type algorithm uses special measure coming from information theory. In this method a new decision function is used, which is closely related to the fuzzy operators. The parameters of the surface is determinated by the global minimum of this decision function.

Our test shows that this new method is efficient and it generalize the original problem.

About the axiomatization of first- and second-order spatio-temporal logics [14]

Sándor Vályi

First first-order temporal logics, introduced in the sixties, root in the theory of first-order modal logics and nowadays are widely used in the theory of specification and verification of arbitrary computational systems. The temporal logic presents a tool for formulating and proving changing-through-time properties of a computational device (either software or hardware). Another possibility is writing specifications directly in a temporal logic language and allow an automated process to plan or construct an appropriate computing device. The temporal logics build on first-order logic have dramatically greater expressive power than logics based on propositional logic but the price of this power is non-axiomatizability, at least of most of these logics.
There is a diversified legion of first-order temporal logics:

1., we can decide what type of time notion is to be used.

2., as in the first-order modal logics, we have to choose what part of language is changing in time (for example, the interpretation of predicates but not the domain is changing).

3., we ca As an introduction to the methods of this paper we fill this lack for the case of $\mathbb{R}$, (we use in this proof only monadic predicate symbols which seems to be a sharpening of existing proofs in the literature) further we show that the first-order temporal logics over $\mathbb{Q}$ are axiomatizable. Reynolds [2] axiomatized some first-order temporal logics over $\mathbb{Q}$ with temporal operators $Until$ and $Since$ and proved completeness in a quite novel way. We give a short proof for the axiomatizability of first-order temporal logics with arbitrary temporal connectives over $\mathbb{Q}$. (Over $\mathbb{Q}$, $Until$ and $Since$ cannot express arbitrary temporal connectives.) A general reason of this is the ω-categoricity of the first-order theory $(\mathbb{Q}, <)$. In this way we shall show – by a rather simple and short translation – the recursive enumerability of the set of theorems of that theory. The price of simplicity is that we do not give any nice inner axiom system written using axioms and deduction rules. n choose some different temporal-logical connectives.

The first-order temporal logic using classical structures (like $\mathbb{N}, \mathbb{Z}, \mathbb{R}$) for time are usually non-axiomatizable. In [1] Garson provided some proofs of the non-axiomatizability over $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$, but his basic first-oder theory was other than classical the so-called free logic. Using classical logic, Garson's proof works only for $\mathbb{N}$ and $\mathbb{Z}$.

After that, our interest turns over temporal logics motivated by the relativistic space-time. At the present we do not know about computational devices that have relativistic features but nobody can exclude them from the future development.

Combining a result of van Benthem [3], that the first-order theory $(\mathbb{Q} \times \mathbb{Q}, L)$ is ω-categorical, with the last method, we are in the position of stating that with such a time notion, the first-order temporal logics are recursively enumerable and hence these logics are axiomatizable. The relation L is a kind of causal connectivity. This time notion (rather spa Further, we investigate the monadic second-order theory of this time notion. We prove it to be not recursively enumerable. It is surprising at first glance because the monadic second-order theory of $(\mathbb{Q}, <)$ is known to be decidable. In fact, one existential quantification over subsets is enough. This means the non-axiomatizability of the $\forall\exists$-fragment of this theory.

We select here only some references:

References

[1] James W. Garson: Quantification in modal logic. Handbook of philosophical logic Vol. II. (ed. Gabbay, Guenthner) 1984.

[2] Mark Reynolds: Axiomatising First-Order Temporal Logic: Until and Since over linear time. Studia Logica 57/1996.

[3] Johan van Benthem: The logic of time. Reidel, Synthese Library 156. 1983.

[14] Research was partly supported by the Hungarian National Foundation for Scientific Research, Grant no. 354-19341 and T30314

Using Decision Trees to Infer Semantic Functions of Attribute Grammars

Szilvia Zvada and Tibor Gyimóthy

In this paper we investigate attribute grammars from the aspect of inductive learning. More precisely, we introduce the LAG method, which is able to infer semantic functions for attribute grammars from examples.

Attribute grammars can be considered as an extension of context-free grammars, where attributes are related to the grammar symbols, and semantic functions define the values of the attributes.
The learning task is to complete the description of a given attribute grammar, where some of the semantic functions are unknown. During the learning, training examples and background knowledge are employed. The examples are words taken from the language generated by the context-free grammar. The incomplete attribute grammar is used as background knowledge. This background grammar might be a grammar in the L-attributed or S-attributed subclasses of non-circular attribute grammars.

The LAG method handles the learning problem as a classification problem and solves it by employing decision tree learning system, C4.5 [Qui93]. Building decision trees has been proved effective in attribute value learning and representing finite, discrete-valued functions. The input data of the C4.5 system expressed as attribute-value tuples are generated from the initial training example set. The decision trees created by the C4.5 are transformed back to semantic functions.

In [GyiHor97], the AGLEARN algorithm has been presented, which uses concept learning approach to infer the unknown semantic functions. Namely, the examples are positive or negative and the unknown semantic functions are generated by propositional learner. The LAG method can be viewed as an improvement of the AGLEARN algorithm. The former uses an attribute-value language instead of the propositional representation, and the background knowledge is more effectively used during the learning than in case of AGLEARN.

The LAG method is applied to the part-of-speech (PoS) tagging of Hungarian sentences. Linguistic phrases and structural information are described by attribute grammars. Based on this information, disambiguation rules for the tagging problem are produced. The main results has been presented in [AleZvaGyi99].

This approach gives rise to use attribute grammars to solve new sorts of problems.

References

[AleZvaGyi99] ALEXIN, Z., ZVADA, SZ., GYIMÓTHY, T.:
Application of AGLEARN on Hungarian Part-of-Speech Tagging. Proceedings of the Second Workshop on Attribute Grammars and their Applications (WAGA'99). Amsterdam, The Netherlands, 1999, p.133-152, INRIA Rocquencourt.

[GyiHor97] GYIMÓTHY, T.,HORVÁTH, T. Learning Semantic Functions of Attribute Grammars Nordic Journal of Computing 4, 1997, p.287-302.

[Qui93] QUINLAN, J.R. C 4.5: Programs for Machine Learning Morgan Kaufmann Publisher, 1993

List of Participants

Alexin, Zoltán: Institute of Informatics, József Attila University, Szeged, Hungary, E-mail: alexin@inf.u-szeged.hu

Arató, Mátyás: University of Debrecen, 4010, Debrecen P.O. Box 12, Hungary, E-mail: arato@math.klte.hu

Aszalós, László: University of Debrecen, 4010, Debrecen P.O. Box 12, Hungary, E-mail: aszalos@math.klte.hu

Balogh, Emese: Institute of Informatics, József Attila University, Szeged, Hungary, E-mail: bmse@inf.u-szeged.hu

Balogh, János: Institute of Informatics, József Attila University, Szeged, Hungary, E-mail: balogh@inf.u-szeged.hu

Bartha, Miklós: Department of Computer Science Memorial University of Newfoundland St. John's, NF, Canada, A1B 3X5, E-mail: bartha@cs.mun.ca

Benczúr, András: Eotvos Lorand University, Department of Information Systems 1117 Budapest, Pázmány P. s. 1/D, E-mail: abenczur@ludens.elte.hu

Bohus, Mihály: Department of Applied Informatics József Attila University, H–6701 Szeged P.O.Box 652, Hungary, E-mail: bohus@inf.u-szeged.hu

Casado, Leocadio G.: Dpto. Arquitectura de Computadores y Electrónica, Universidad de Almería 04120-Almería, Spain , E-mail: leo@miro.ualm.es

Ciric, Miroslav: University of Nis, P.O. Box 91, 18000 Nis, Yugoslavia, E-mail: mciric@archimed.filfak.ni.ac.yu

Csallner, András Erik: Department of Computer Science, Juhász Gyula Teachers Training College 6720 Szeged, Boldogasszony sgt. 6, Hungary, E-mail: csallner@jgytf.u-szeged.hu

Csendes, Tibor: Department of Applied Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: csendes@inf.u-szeged.hu

Demetrovics, János: MTA SZATKI, Budapest, Hungary, E-mail: dj@ilab.sztaki.hu

Dezső, András: Siemens - Sysdata, Budapest, Hungary, E-mail: andras.dezso@sysdata.siemens.hu

Dibuz, Sarolta: Ericsson Ltd., Budapest, Hungary, E-mail: Sarolta.Dibuz@eth.ericsson.se

Dominich, Sándor: University of Veszprém, Hungary, E-mail: dominich@almos.vein.hu

Dudásné Nagy, Marianna: Department of Computer Science, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: marcsi@inf.u-szeged.hu

Farkas, Károly: Technical University of Budapest, Hungary, E-mail: csocso@sch.bme.hu

Farkas, Lóránt: Technical University of Budapest, Hungary, E-mail: farkas@nov.mht.bme.hu

Fazakas, Antal: Nokia Hungary Ltd., Budapest, Hungary, E-mail: antal.fazakas@nokia.com

Fazekas, Gábor: University of Debrecen 4010, Debrecen P.O. Box 12, Hungary,
E-mail: fazekasg@math.klte.hu

Felső, Gábor: Technical University of Budapest, Hungary, E-mail: felso@seeger.iit.bme.hu

Felvégi, Péter: Technical University of Budapest, Hungary, E-mail: petschy@avalon.aut.bme.hu

Friedler, Ferenc: University of Veszprém, Hungary, E-mail: friedler@almos.vein.hu

Frohner, Ákos: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D,
E-mail: szamcsi@caesar.elte.hu

García, José A. Martínez: University of Almeria, Dept. of Computer Architecture and Electronics, Spain,
E-mail: jamartin@davinci.ualm.es

Garzon, G. Ester Martin: University of Almeria, Dept. of Computer Architecture and Electronics, Spain,
E-mail: ester@renoir.ualm.es

Gesztesi, Gábor: Technical University of Budapest, Hungary, E-mail: ggabor@inf.bme.hu

Gécseg, Ferenc: Department of Computer Science, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: gecseg@inf.u-szeged.hu

Gombás, Éva: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary,
E-mail: gombas@inf.u-szeged.hu

Gombos, László: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D,
E-mail: gombi@caesar.elte.hu

Gonzalez-Linares, Jose M.: University of Malaga, Spain, E-mail: gonzalez@ac.uma.es

Gőgös, Barnabás: University of Veszprém, Hungary, E-mail: barna@mail.irt.vein.hu

Gulyás, László: Computer and Automation Research Institue, Budapest, Hungary,
E-mail: laszlo.gulyas@sztaki.hu

Gyenesei, Attila: University of Turku, Finland, E-mail: gyenesei@cs.utu.fi

Gyenizse, Pál: Cygron Research and Development, Szeged, Hungary, E-mail: zsipo@cygron.hu

Hajdu, András: University of Debrecen 4010, Debrecen P.O. Box 12, Hungary,
E-mail: ahajdu@infotech.math.klte.hu

Halmai, Csongor: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary,
E-mail: halmai@inf.u-szeged.hu

Harmati, István: Technical University of Budapest, Hungary, E-mail: harmati@seeger.iit.bme.hu

Herendi, Tamás: University of Debrecen, 4010, Debrecen P.O. Box 12, Hungary,
E-mail: herendi@zuse.math.klte.hu

Heszberger, Zalán: Technical University of Budapest, Hungary, E-mail: heszi@ttt-atm.ttt.bme.hu

Hornák, Zoltán: Technical University of Budapest, Hungary, E-mail: hornak@mit.bme.hu

Horváth, Endre: Technical University of Budapest, Hungary, E-mail: horvathe@ttt-atm.ttt.bme.hu

Horváth, Zoltán: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D,
E-mail: hz@ludens.elte.hu

Hosszú, József: Ericsson Ltd., Budapest, Hungary, E-mail: Jozsef.Hosszu@eth.ericsson.se

Huszerl, Gábor: Technical University of Budapest, Hungary, E-mail: huszerl@mit.bme.hu

Imreh, Csanád: Department of Informatics, University of Szeged, Árpád tér 2, H-6720 Szeged, Hungary, E-mail: cimreh@inf.u-szeged.hu

Imreh, Balázs: Department of Computer Science, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: imreh@inf.u-szeged.hu

Jánvári, Gusztáv: E-mail: janvari@externet.hu

Jovanovic, Radmila: University of Nis, Yugoslavia, E-mail: radmilaj@orion.eknfak.ni.ac.yu

Karsai, János: SZOTE Orvosi Informatika Intézet, Szeged, Hungary, E-mail: karsai@dmi.szote.u-szeged.hu

Katsányi, István: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D, E-mail: kacsa@ludens.elte.hu

Keresztfalvi, László: E-mail: kereszt@iname.com

Kiss, Bálint: Technical University of Budapest, Hungary, E-mail: bkiss@seeger.iit.bme.hu

Kiss, Krisztián: Nokia Hungary Ltd., Budapest, Hungary, E-mail: krisztian.kiss@nokia.com

Koch, György: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: koch@inf.u-szeged.hu

Kocsor, András: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: kocsor@inf.u-szeged.hu

Kókai, Gabriella: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: kokai@informatik.uni-erlangen.de, kokai@sol.cc.u-szeged.hu

Kormos, János: University of Debrecen 4010, Debrecen P.O. Box 12, Hungary, E-mail: kormos@math.klte.hu

Kovacevic, Jelena: University of Nis, Yugoslavia, E-mail: jkovac@archimed.filfak.ni.ac.yu

Kozma, László: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D, E-mail: kozma@ludens.elte.hu

Krémer, Péter: Ericsson Ltd., Budapest, Hungary, E-mail: Peter.Kremer@eth.ericsson.se

Krész, Miklós: Department of Computer Science JGYTF, University of Szeged, H-6722 Szeged, Boldogasszony sgt. 6, Hungary, E-mail: kresz@jgytf.u-szeged.hu

László, István: E-mail: listv@rsc.fomi.hu

Markót, Mihály Csaba: Institute of Informatics, József Attila University, 6720 Szeged, Árpád tér 2, Hungary, E-mail: h430810@stud.u-szeged.hu

Maróti, György: Zenon Kft., Szeged, Hungary, E-mail: maroti@zenon.hu

Martonossy, László: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: lmarton@sol.inf.u-szeged.hu

Máté, Eörs: Department of Applied Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: mate@inf.u-szeged.hu

Milovanovic, Slavoljub: University of Nis, Yugoslavia, E-mail: smilovan@orion.eknfak.ni.ac.yu

Nagy, Antal: Department of Applied Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: nagya@inf.u-szeged.hu

Nagy, Benedek: University of Debrecen 4010, Debrecen P.O. Box 12, Hungary, E-mail: nbenedek@math.klte.hu

Nagy, Lajos: Technical University of Budapest, Hungary, E-mail: t-nagy@nov.mht.bme.hu

Németh, Zsolt: MTA SZTAKI, Budapest, Hungary, E-mail: Zsolt.Nemeth@sztaki.hu

Nyúl, László: Department of Applied Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: nyul@inf.u-szeged.hu

Ortigosa, Eva Martinez: University of Malaga, Spain, E-mail: eva@ac.uma.es

Palágyi, Kálmán: Department of Applied Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: palagyi@inf.u-szeged.hu

Pap, Gyula: University of Debrecen 4010, Debrecen P.O. Box 12, Hungary, E-mail: papgy@math.klte.hu

Pap, Zsigmond: Technical University of Budapest, Hungary, E-mail: papzs@mit.bme.hu

Pataricza, András: Technical University of Budapest, Hungary, E-mail: pataric@mmt.bme.hu

Podhorszki, Norbert: MTA SZTAKI, Budapest, Hungary, E-mail: pnorbert@sztaki.hu

Popovic, Zarko: University of Nis, Nis, Yugoslavia, E-mail: zpopovic@orion.eknfak.ni.ac.yu

Pukler, Antal: Széchenyi István College, Department of Computer Science H-9026 Győr, Hédervári út 3. Hungary, E-mail: pukler@szif.hu

Putnik, Zoran: Institute of Mathematics, Faculty of Science, University of Novi Sad, E-mail: putnik@unsim.ns.ac.yu

Radovic, Ognjen: University of Nis, Nis, Yugoslavia, E-mail: ogi@orion.eknfak.ni.ac.yu

Rapcsák, Tamás: MTA SZTAKI, 1518 Budapest, Pf 63, Hungary, E-mail: rapcsak@sztaki.hu

Recski, András: Technical University of Budapest, Hungary, E-mail: recski@math.bme.hu

Rotter, Vilmos Csaba: Nokia Hungary Ltd., Budapest, Hungary, E-mail: vilmos.rotter@nokia.com

Sára, Levente: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: lsara@inf.u-szeged.hu

Schrettner, Lajos: Department of Computer Science, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: schrettner@inf.u-szeged.hu

Sebő, Mariann: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: sebo@inf.u-szeged.hu

Selényi, Endre: Technical University of Budapest, Hungary, E-mail: selenyi@mmt.bme.hu

Son, Tran Minh: Technical University of Budapest, Hungary, E-mail: sony@sirkan.sch.bme.hu

Szabó, Péter Gábor: Department of Applied Informatics, University of Szeged, H–6701 Szeged P.O.Box 652, Hungary, E-mail: pszabo@inf.u-szeged.hu

Szabó, Szilvia: University of Debrecen 4010, Debrecen P.O. Box 12, Hungary,
E-mail: `szabosz@math.klte.hu`

Szépkúti, István: H-1149 Budapest, Bíbor u. 18. III/2., Hungary, E-mail: `szepkuti@inf.u-szeged.hu`

Szilágyi, Gyöngyi: University of Szeged, Research Group on Artificial Intelligence, Szeged, Hungary,
E-mail: `szilagyi@rgai.inf.u-szeged.hu`

Szűcs, László: Nokia Hungary Ltd., Budapest, Hungary, E-mail: `laszlo.szucs@nokia.com`

Tanács, Attila: Department of Applied Informatics József Attila University, H–6701 Szeged P.O.Box 652, Hungary, E-mail: `tanacs@inf.u-szeged.hu`

Tarnay, Katalin: Nokia Hungary Ltd., E-mail: `katalin.tarnay@nokia.hu`

Tóth, László: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary,
E-mail: `tothl@inf.u-szeged.hu`

Tran Tuan Hung: Technical University of Budapest, Hungary, E-mail: `HUNG@plan.hit.bme.hu`

Ulbert, Attila: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D, E-mail: `mormota@elte.hu`

Vajda, Szilárd: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D,
E-mail: `vsz@valerie.inf.elte.hu`

Vályi, Sándor: University of Debrecen 4010, Debrecen P.O. Box 12, Hungary,
E-mail: `valyis@math.klte.hu`

Varga, László: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D,
E-mail: varga@valerie.inf.elte.hu

Varga, László: Technical University of Budapest, Hungary, E-mail: `vargal@nimrud.eet.bme.hu`

Zsíros, Ákos: Department of Applied Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: `zsiros@inf.u-szeged.hu`

Zsók, Viktória: Eötvös Loránd University, H-1117 Budapest, Pázmány P. s. 1/D,
E-mail: `zsv@valerie.inf.elte.hu`

Zvada, Szilvia: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary,
E-mail: `zvada@sol.cc.u-szeged.hu`

Notes