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 2–5, 2008
Szeged, Hungary

Scientific Committee:

Ferenc Gécseg (Chair, SZTE)
András Benczúr (ELTE)
Hasszan Charaf (BME)
Tibor Csendes (SZTE)
János Csirik (SZTE)
József Dombi (SZTE)
Zoltán Ésik (SZTE)
Zoltán Fülöp (SZTE)
Tibor Gyimóthy (SZTE)
Zoltán Horváth (ELTE)
Zoltán Kató (SZTE)
Zoltán Kása (Sapientia EMTE)
János Kormos (KLTE)
László Kozma (ELTE)
Eörs Máté (SZTE)
Attila Pethő (KLTE)
András Recski (BME)
Endre Selényi (BME)
Tamás Szirányi (SZTAKI)

Organizing Committee:

Kálmán Palágyi, Balázs Bánhelyi, Tamás Gergely, István Matijevics

Address of the Organizing Committee

c/o. Kálmán Palágyi
University of Szeged, Institute of Informatics
H-6701 Szeged, P.O. Box 652, Hungary
Phone: +36 62 546 396, Fax: +36 62 546 397
E-mail: cscs@inf.u-szeged.hu
URL: http://www.inf.u-szeged.hu/∼cscs/

Sponsors

Novadat Bt., Győr
Polygon Publisher
Siemens PSE Kft.
University of Szeged, Institute of Informatics

Preface

This conference is the sixth 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 among 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 2008. The manuscripts will be forwarded to the proper journals.

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, June 2008

Kálmán Palágyi
Balázs Bánhelyi
Tamás Gergely
István Matijevics

Contents

Preliminary Program

Overview

Wednesday, July 2

- 09:00 - 10:15 Talks (3x25 minutes)
- 10:15 - 10:30 Break
- 10:30 - 11:20 Plenary talk
- 11:20 - 11:40 Break
- 11:40 - 12:30 Talks (2x25 minutes)
- 12:30 - 14:00 Lunch at the Gödör restaurant
- 14:00 - 15:15 Talks (3x25 minutes)
- 15:15 - 15:30 Break
- 15:30 - 16:45 Talks (3x25 minutes)
- 16:45 - 18:00 Free program
- 18:00 Dinner at the Gödör restaurant

Thursday, July 3

- 09:00 - 10:15 Talks (3x25 minutes)
- 10:15 - 10:30 Break
- 10:30 - 11:20 Plenary talk
- 11:20 - 11:40 Break
- 11:40 - 12:30 Talks (2x25 minutes)
- 12:30 - 14:00 Lunch at the Gödör restaurant
- 14:00 - 15:40 Talks (4x25 minutes)
- 15:40 - 15:55 Break
- 15:55 - 16:45 Talks (2x25 minutes)
- 16:45 - 18:00 Free program
- 18:00 Reception at the conference place

Friday, July 4

- 09:00 - 10:15 Talks (3x25 minutes)
- 10:15 - 10:30 Break
- 10:30 - 11:20 Plenary talk
- 11:20 - 11:40 Break
- 11:40 - 12:30 Talks (2x25 minutes)
- 12:30 - 14:00 Lunch at the Gödör restaurant
- 14:00 - 15:40 Talks (4x25 minutes)
- 15:40 - 16:00 Free program
- 16:00 Social program (visit the Informatic Collection and dinner)

Saturday, July 5

- 09:00 - 10:15 Talks (3x25 minutes)
- 10:15 - 10:30 Break
- 10:30 - 11:20 Plenary talk
- 11:20 - 11:40 Break
- 11:40 - 12:30 Talks (2x25 minutes)
- 12:30 - 14:00 Lunch at the Gödör restaurant
- 14:00 - 15:15 Talks (3x25 minutes)
- 15:15 Closing

Detailed program

Wednesday, July 2

Sections	**Metaprogramming**
9:00	Norbert Pataki: *A Comparative Study of C++ Standard Template Library's Formal Specification*
9:25	Áron Sisak: *Verification of UML 2 State Machines by Automated Model Transformation to the SAL Model Checker*
9:50	Gergely Dévai: *Metaprogramming on the Proof Level*
10:15	Break
10:30	**Plenary talk** Tamás Horváth: *Mining Chemical Graphs*
11:20	Break
Sections	**Testing**
11:40	Zalán Szűgyi: *Necessary Test Cases for Decision Coverage and Modified Condition / Decision Coverage*
12:05	Ferenc Bozóki: *Test Component Assignment and Scheduling in a Load Testing Environment*
12:30	Lunch at the Gödör restaurant
Sections	**Operation Research**
14:00	László Pál: *Matlab Version of the GLOBAL Optimization Method*
14:25	Levente Hunyadi: *An Identification Approach to Dynamic Errors-in-variables Systems with a Preliminary Clustering of Observations*
14:50	Attila Pásztor: *Research of Swarm Intelligence Simulation with the Help of NXT Robots*
15:15	Break
Sections	**Artificial Intelligence**
15:30	Emese Szádecky-Kardoss: *Designing a Tracking Controller for Passenger Cars with Steering Input*
15:55	Kristóf Csorba: *Searching for Similar Documents Between Mobile Devices*
16:20	Csaba Főző: *3-level Confidence Voting Strategy for Dynamic Fusion-Selection of Classifier Ensembles*
16:45	Free program
18:00	Dinner at the Gödör restaurant

Thursday, July 3

Sections	**Image Processing**
9:00	László Csernetics: *Backprojection Reconstruction Algorithm Using Order Statistic Filters in Breast Tomosynthesis*
9:25	Mihály Gara: *Determination of Geometric Features of Binary Images from Their Projections by Using Decision Trees*
9:50	Balázs Erdőhelyi: *Bone Fragment Repositioning Using Registration Methods*
10:15	Break
10:30	**Plenary talk** Márk Jelasity: *Gossip Algorithms in Peer-to-peer Computing*
11:20	Break
Sections	**Clouds, Fuzzy Sets**
11:40	Martin Fuchs: *Potential Clouds in Robust Design*
12:05	József Dániel Dombi: *Function Approximation with Fuzzy Operators*
12:30	Lunch at the Gödör restaurant
Sections	**Graph Theory, Algorithms**
14:00	Thorsten Bonato: *Lifting and Separation Procedures for the Cut Polytope*
14:25	Jan-Hendrik Prinz: *Enhanced Phasespace Sampling of Biomolecular Systems Using Metastability*
14:50	András Csernenszky: *The Chooser-Picker 7-in-a-row-game*
15:15	Csaba Legány: *Modeling Web Service Interaction Using Markovian Processes*
15:40	Break
Sections	**Branch and Bound Technique**
15:55	László Pál: *A Global Optimization Algorithm for INTLAB*
16:20	Anett Rácz: *Determining Initial Bound by "Ray-method" in Branch and Bound Procedure*
16:45	Free program
18:00	Reception at the conference place

Thursday, July 4

Sections	**Program Analysis**
9:00	Judit Jász: *Efficient Static Impact Analysis*
9:25	Tamás Nagy: *Call Graph and Data Flow analysis of Dynamic Functional Language*
9:50	Ferenc Fischer: *Calculation and Application of the Dynamic Function Level Dependencies in Java Programs*
10:15	Break
10:30	**Plenary talk** Zora Konjovic: *Geo-space Management Based on Information and Communication Technologies*
11:20	Break
Sections	**Distributed Solutions**
11:40	Máté Tejfel: *Verified Mobile Code Repository in the Intelligent Space*
12:05	József Marton: *In-memory Preprocessing of Streaming Sensory Data: a Partitioned Relational Database Approach*
12:30	Lunch at the Gödör restaurant
Sections	**Recognition, Skeletonization Algorithms**
14:00	Levente Sajó: *Enhancing Facial Features Detection with Deformable Face Templates*
14:25	Zsolt T. Kardkovács: *High Precision Pedestrian Recognition in Low Resolution Images*
14:50	Péter Kardos: *An Order-Independent Sequential Thinning Algorithm*
15:15	György Kovács: *Skeletonization Based on Neighborhood Sequences*
15:40	Free program
16:00	Social program: Visit the Informatic Collection and dinner at the Fehértó Fishermen's Tavern

Saturday, July 5

Sections	**Allocation Problems**
9:00	László Illyés: *A Special Human Resource Allocation Problem*
9:25	Tamás Németh: *Parameter Learning Algorithms in Online Scheduling*
9:50	Attila Kertész: *Adaptive Scheduling Solution for Grid Meta-brokering*
10:15	Break
10:30	**Plenary talk** Gábor Ivanyos: *The Hidden Subgroup Problem in Quantum Computing*
11:20	Break
Sections	**Theoretical Computer Science**
11:40	Izabella Stuhl: *Steiner Loops*
12:05	Ville Piirainen: *Principal Varieties of Finite Congruences*
12:30	Lunch at the Gödör restaurant
Sections	**Program Analysis and Evaluation**
14:00	Gabriella Tóth: *Analysis of Size Differences Between Static and Dynamic Slices*
14:25	István Siket: *Evaluating the Effectiveness of Object Oriented Metrics for Bug Prediction*
14:50	Lajos Jenő Fülöp: *Introducing a Benchmark for Evaluating Reverse Engineering Tools*
15:15	Closing

Lifting and Separation Procedures for the Cut Polytope

Thorsten Bonato

In this joint work with Gerhard Reinelt we describe a new separation and lifting approach for the cut polytope on arbitrary graphs exploiting known methods for dense and complete graphs respectively. Furthermore, we introduce a preliminary implementation using the branch&cut framework ABACUS.

Separation and lifting

The *max-cut problem* is one of the most studied combinatorial optimization problems. It is known to be $\mathcal{NP}$-hard and has a number of interesting applications, such as optimal design of VLSI circuits or the identification of configurations of minimal energy of so-called *spin glasses.*

Like for most of the hard combinatorial optimization problems, the technique that is typically used to solve max-cut problems to optimality is *branch&cut.* But its effectiveness heavily depends on the quality and speed of the separation routines used for the approximation of the *cut polytope*. The better part of the currently known methods are related to the max-cut problem on *dense* and *complete* graphs respectively and can only be successfully applied to relatively small problem instances.

A trivial way to exploit this knowledge for arbitrary graphs would be the artifical completion by adding zero-weighted edges. In this context the number of nodes is crucial for both basic practicability and duration of the separation.

Our approach uses a graph shrinking procedure to reduce the number of nodes prior to the artificial completion. Violated inequalities found on the resulting complete graph are then translated to the original graph by means of projection and lifting.

Implementation

A preliminary version has been implemented using the programming language C++ and the branch&cut framework *ABACUS*. It contains various separation routines, amongst others a variant of so-called *target cuts.*

References

[1] M. Jünger, G. Reinelt, G. Rinaldi. Lifting and separation procedures for the cut polytope, *Technical Report*

Test Component Assignment and Scheduling in a Load Testing Environment

Ferenc Bozóki and Levente Erős

When the construction of a network component has finished the testing phase begins, which is just as important as the implementation itself. During testing the conformance test has to be executed first. The conformance test examines whether the System Under Test (SUT) corresponds to its specifications, that is whether it generates the appropriate output and transfers to the appropriate state when induced by some kind of an input. Once the SUT has passed the conformance test, the load test begins, which stresses the SUT by the maximal load it has to be able to bear with in its latter real-life environment, while examining its behavior in this extreme situation.

However, for generating this high load we far don't have as much host computers in the test environment as the SUT will have to serve once it gets into its real-life environment. This means each host in the test environment has to emulate more hosts of the real-life environment. To carry this out these real-life hosts are represented by software entities, the so-called test components in the test environment, and each one of these test components is assigned to a host in the test environment. The question now is how these assigments have to be made. Main objectives of most of the task assignment algorithms developed so far are to maximize the number of running tasks (sometimes even overloading the hosts is permitted) and to minimize the communication costs between the tasks [1,2]. In our case the tasks (test components) don't communicate with each other and in order to simulate the real-life environment faithfully, overloading testing hosts is prohibited. This implies that a fully new approach is needed. In this paper we present a heuristic test component assignment algorithm that takes the above mentioned aspects into consideration and that is specialized for handling typical load testing traffic patterns.

The other problem we introduce a possible solution for is scheduling in a load testing environment. In the case of conformance tests where the test environment emulates a single user, scheduling can be derived from real time scheduling. Contrarily, when dealing with load tests the test environment acts as a big amount of users or hosts, and as mentioned above a testing host has to run many test components representing many users parallelly. This leads us to another problem, namely the problem of scheduling these test components inside a testing host. The essence of the problem is that if each one of the many test components assigned to a testing host run on separate threads just like in the classic case, context switching takes a dominant part of the CPU load. The solution we developed for solving the problem of the huge overload caused by context switching introduces the concept of virtual threads. A virtual thread plays the same role as a regular thread in the classical approach, that is, exactly one test component is assigned to a virtual thread. These virtual threads are then branched together into regular threads. By significantly reducing the number of regular threads this way, the CPU load used for context switching can be reduced to a near-zero level.

As our simulations have shown handling these two problems by the solutions we present in this paper results in a significant raise in the efficiency of executing load tests.

References

[1] M. Kafil and I. Ahmad. Optimal Task Assignment in Heterogeneous Distributed Computing Systems, *Heterogeneous Computing Workshop, 1997. (HCW '97) Proceedings, p135-146. 1997.*

[2] Task Assignment and Transaction Clustering Heuristics for Distributed System, *Information Sciences vol. 97. Issue 1-2, p199-219. 1997.*

The Chooser-Picker 7-in-a-row-game

András Csernenszky

Our main objective is to show how the Beck's conjecture intertwines the Chooser-Picker games with the k-in-a-row games. The k-in-a-row game deserves consideration by itself, because the last result of this topic was the famous theorem about the 8-in-a-row game (it is a blocking draw on the infinite chess-board) is more than 27 years old, see [3, 4]. Since then a lot people tried to prove similar theorem for the 7-, or 6-in-a-row-game, but up to now without success. We sketch a possible way to show that (assuming perfect play) the 7-in-a-row is also a blocking draw.

Given a hypergraph $(V, \mathcal{F})$ there are a number of games that can be played on it. In the *Maker-Maker* version, the first player and the second player take the elements of V, and the winner is who gets all elements of an $A \in \mathcal{F}$ first. In the *Maker-Breaker* version Maker wins by occupying all elements of an $A \in \mathcal{F}$, while Breaker wins by preventing Maker in doing so. Finally in the *Picker-Chooser* version Picker selects two vertices of V, Chooser takes one of those, and then the other of course goes back to Picker. Chooser wins if he occupies a whole winning set, while Picker wins if he can prevent this. When V is odd, the last element goes to Chooser.

While the Chooser-Picker games are interesting on their own, these are also useful tools to understand positional games better, see details in [2, 1]. Namely, from the result of a Chooser-Picker game we have an insight to the result of the Maker-Breaker version: these coincide several times, and it seems that Picker is *always* better off than Breaker.

In this work we prove that Picker wins the Chooser-Picker version of the 7-in-a-row game. To prove the theorem, we exhibit an appropriate tiling of the board. The tiling is constructed in such a way that if Picker wins an auxiliary game on each tiles then Picker wins the original 7-in-a-row game. The proof of this case is a medium size case study.

References

[1] A. Csernenszky, C. I. Mándity and A. Pluhár, On Chooser-Picker Positional Games, *submitted*.

[2] J. Beck, Positional games and the second moment method, *Combinatorica* **22** (2) (2002) 169–216.

[3] E. R. Berlekamp, J. H. Conway and R. K. Guy, *Winning Ways for your mathematical plays*, Volume **2**, Academic Press, New York 1982.

[4] R. K. Guy and J. L. Selfridge, Problem S.10, *Amer. Math. Monthly* **86** (1979); solution T.G.L. Zetters **87** (1980) 575-576.

Backprojection Reconstruction Algorithm Using Order Statistic Filters In Breast Tomosynthesis

László Csernetics

Breast cancer is the most common cancer type and one of the leading cause of death among women. It has been recognized over the years that preventing the disease is the most powerful weapon, and the implementation of screening mammography has had significantly reduced the death rate. However, it is also proven that conventional mammography does not detect approximately 30 percent of breast cancers. Inventing new imaging technologies for the earlier detection of breast cancer is vital and is in the center of many ongoing studies. There are several new techniques using different imaging modalities that are under investigation. The most promising is the breast tomosynthesis, an advanced x-ray application that addresses the problem of structure superimposition, one of the major deficiency of 2D mammography, by reconstructing a range of slices providing additional 3-dimensional information of the breasts.

Our goal is to investigate and develop reconstruction algorithms that fit into the new mathematical model of tomosynthesis used in mammography. In this paper we show a backprojection reconstruction technique that is especially well-suited for the problem in question. This algorithm is capable to produce contrast-enhanced slices of the breast by taking only the projections that are most probably holds the "important" information of the targeted lesions, ignoring part of the projections. This statistical approach also offers a good noise management performance, as a fortunate side-effect. After discussing the algorithm we publish the results of the comparison of this technique with other popular methods of the algorithm-family. We also look out the strict boundaries of the work done suggesting improvements of the reconstruction algorithm.

Acknowledgements

The author would like to express thanks to his mentor, Dr. Attila Kuba, who supported this work with a great deal of help in research. Without his great experience and knowledge these results could not have been born. The author also received a considerable deal of support and inspiration to prepare this paper and keep on studying the research topic from his supervisor, Dr. Kálmán Palágyi.

References

[1] American Cancer Society: Mammograms and other breast imaging procedures. ACS, Cancer Facts and Figures (2004)

[2] Claus, B.E., Eberhard, J.W.: A new method for 3D reconstruction in digital tomosynthesis. *Med. Imag. SPIE Proc.* **4684** (2002)

[3] Csernetics, L.: Melltomoszintézis: 3D Mammográfia. *Proc. 6th Conference of Hungarian Association for Image Processing and Pattern Recognition* (2007) 9–19

[4] Haaker, P., Klotz, E., Koppe, R., Linde, R., Möller, H.: A new digital tomosynthesis method with less artifacts for angiography. Med. Phys. **12(4)** (1985) 431–436

[5] Herman, G.T.: Image reconstruction from projections. *The fundamental of computerized tomography.* Academic Press, NY (1980) 1–40, 55–65, 90–147

[6] Klotz, E., Haaker, P., Koppe, R., Linde, R.: Three-dimensional angiography with digital flashing tomosynthesis. *Adv. Imag. Proc. SPIE Proc.* **804** (1987) 306–313

[7] Pisano, E.D., et al, Digital Mammographic Imaging Screening Trial (DMIST) Investigators Group: Diagnostic performance of digital versus film mammography for breast-cancer screening. *N. Engl. J. Med.* **353(17)** (2005) 1773–1783

[8] Rafferty, E.A.: Advances in imaging: Breast tomosynthesis. ASCO, Virtual Meeting (2004)

[9] Rafferty, E.A.: Breast tomosynthesis: Three-dimensional mammography. ASCO, 40th Annual Meeting (2004) 45–47

[10] Rafferty, E.A.: Tomosynthesis: New weapon in breast cancer fight. ImagingEconomics.com (2004)

[11] Smith, A.P.: Contrast-enhanced breast tomosynthesis – A promising tool for identifying breast cancer. US Special Populations (2006) 46–49

[12] Wu, T., Rafferty, E.A., Kopans, D.B., Moore, R.H.: Contrast-enhanced breast tomosynthesis. DBT for better breast cancer diagnosis. RT Image **17(41)** (2004)

Searching for Similar Documents Between Mobile Devices

Kristóf Csorba and István Vajk

This paper proposes a system helping mobile device users to search for documents which are similar to the ones already stored on the user's device locally. The goal is to create a background process monitoring available documents in a peer-to-peer environment and notifying the user if a new document of possible interest gets retrievable. As the system is designed for mobile device environment, it has to satisfy strict conditions according processor and memory consumption. The program running in background may not cause significant performance degradation, otherwise the user interface might get slower which decreases usability. This is accomplished by employing very fast and simple algorithms for topic representation and comparison. Beside the resource consumption, the communication traffic created by the mobile device has to be kept low as well, because traffic usually costs a fee. Users would probably not like to pay huge bills just because background processes are transmitting document topic representations between the mobile devices. The compact topic representation is achieved using topic specific keyword lists and mask vectors indicating the presence or absence of a given keyword in the document. These topic representations are then compared using a simple similarity measure based on the number of common keywords. The key idea behind the compact topic representation is an iterative selection of the best matching keyword list, that is, the keyword list having the most common keywords with the document. After this keyword list is selected, all the contained keywords get a binary flag indicating their presence or absence in the document. As the keyword lists are globally available, topic representations contain only a keyword list identifier number and a single bit for all keywords in it in a form of a binary mask vector. If a keyword list has not more than 128 keywords and the keyword list identifier is 16 bit, the topic representation is 16+128 bit = 18 byte for every document. Using the keyword lists, these mask vectors can be mapped into a global space of all possible keywords and compared using simple scalar product resulting the number of common keywords between the observed documents. As topic specific keywords are relative rare, there are many documents not containing any of the keywords which makes the topic unidentifiable. To overcome this limitation, a cascaded selection procedure is employed using multiple levels of document selection: after documents similar to the local ones are retrieved, the keyword lists are extended with additional keywords (usually with lower quality). This allows the retrieval of further documents, although with lower precision. If the user is allowed to choose the allowed amount of additional keywords, a trade-off between precision and recall may be defined by the user. This paper presents the techniques described above in detail, together with experimental results investigating multiple aspects of the proposed systems performance.

Metaprogramming on the Proof Level

Gergely Dévai

This research is about the usability of metaprogramming techniques for construction of program correctness proofs.

In metaprogramming one uses two languages: a base language and a meta language. The compilation is done in two phase: first a precompiler processes the meta language constructs and generates a result written purely in the base language. Examples vary from macro assemblers to template metaprogramming in C++. In these classical applications metaprogramming is used to generate programs.

The goal of formal programming methods is to produce verified programs. These methods try to increase reliability of software by formally proving its properties. In order to achieve this goal, one has to construct correctness proofs of programs. There are two main approaches: writing the program first and proving its properties afterwards (verification), or deriving the program from its specification using rules that guarantee soundness (correctness by construction).

We use the second approach. In our system the programmer writes specification first, then refines it by more detailed specification statements. This process is called stepwise behavioral refinement and it results in a correctness proof of the algorithm. Our system checks this proof and generates the program automatically, which is then correct by construction.

We investigate how to support proof construction using metaprogramming techniques. That is, the base language is the proof language and metaprogramming elements are used to generate proof fragments. Let us summarize the advantages of this approach.

- *Decreased proof-length.* If we identify often used proof fragments and generalize them to proof templates we can decrease the length of the proof by calling the templates instead of repeating the proof fragment several times.

- *Templates make the system extensible.* In our system we specify only a small subset of instructions. The proof for programming constructs like conditional statements, loops, procedure calls are generated by templates. This means that the verification conditions for these constructs are not built into our system. If one needs a new kind of loop that is not supported yet, can develop a template for the new construct. This is not possible in systems where the supported language elements are hard-wired in the system core.

- *Meta programming is safe at the proof level.* Primitive metaprogramming constructs, like macros sometimes generate inefficient or erroneous program fragments. This can also happen when metaprogramming is used to generate proof. Fortunately the generated proof is checked afterwards and all the errors are reported to the programmer already in compile time.

- *The trusted base can be minimal.* In every verification framework we have to trust the system: its proof engine, the rules about different language constructs etc. In our system this trusted base is minimal: it consists of the module that checks the validity of proofs and the specification of primitive instructions. All other features are provided by templates, and, according to the previous point, erroneous templates can not corrupt the soundness of the system.

In this paper we present a set of metaprogramming constructs that we have found useful in proof generation. These include different types of proof-templates, conditions, passing proof fragments as parameters and generation of template definitions by templates. We also present how to use these features to implement proofs for different programming language constructs.

Function Approximation with Fuzzy Operators

József Dombi and József Dániel Dombi

Numerical analysis is the area of mathematics and computer science that creates, analyzes and implements algorithms for solving numerical problems of continuous mathematics. One of the main subfields of numerical analysis is interpolation. Interpolation is a method, where we determine a function (in most cases it is a polinom) which fits to the given data points and using this result we can determine the function value if new data points are given. In science in most cases we obtain samplings to determine the relations between inputs and outputs, which is called curve fitting, because usually we do not require the exact fit, only the approximation. We can say interpolation is a specific case of curve fitting, which case the function must go exactly through data points. There are lots of interpolations for example: linear, polynomial, spline and trignometric. Curve fitting can be done by minimizing the error function that measures the misfit between the function for any given value and the data points. One simple and widely used error function is the sum of the squares of the errors between the predictions for each data point and the corresponding target values, so that we minimize the energy function. The problems of the curve fitting and interpolation are that we do not understand the meaning of the parameters. In our approach we drop the traditional concept. We define effects as a basic units, which have meaning. In our system all function defined on [0,1] interval. In order to working in [0,1] interval we have to transform the real function to [0,1]. This can be done by using a linear function or using the Sigmoid function. Our new novel technique for function creation use Sigmoid function, Dombi operator, Conjunction operator and Aggregation operator [1],[2]. First we need natural effect, which is created by two Sigmoid functions, and the conjunction operator. We define positive effect, which means that this effect will increase the function value and we define negative effect, which is the opposite of the positive effect (ie. decrease the function value). The last step of function creation is to aggregate these effects. In function approximation we know the values. Our task is to divide real values into effects. We use global optimization for this problem. The critical point of the optimization is the initial values of the effects. If we define good initial points we have a high chance to find the global optimum. We develop a Java program to analyze our approach. In this system we can create any arbitrary funcion, and we can test the capabality of function approximation.

References

[1] Dombi J.: A general class of fuzzy operators, the De Morgan class of fuzzy operators and fuzziness measures induced by fuzzy operators, Fuzzy Sets and Systems.

[2] Detyniecki M..: Fundamentals on Aggregation Operators, asturias

Bone Fragment Repositioning Using Registration Methods

Balázs Erdőhelyi

The surgical intervention of complex bone fractures has to be planned very carefully, especially for such a complex bone as the pelvic ring. The computer aided surgical planning is done before the actual surgery takes place and its main purpose is to gather more information about the dislocation of the bone fragments and to arrange the surgical implants to be inserted. With the help of finite element analysis even the biomechanical stability of the whole plan can be predicted.

To create such a plan the following steps are performed (as described in [1]). First, the CT dataset of the patient is segmented, which enables us to treat the different bones and broken fragments separately. Next the surface of these bones is determined and presented in a 3D environment. Since the fragments may have moved during the fracture, it is essential to move and rotate them back to their original or healthy position. Without this repositioning step, no implants can be inserted, since the final locations of the fragments are unknown. Previous solutions to the fracture reduction problem included moving the fragments with the mouse, or a special 3D haptic device. The former is not intuitive to use since the mouse is only 2D, and the later is expensive and still requires learning.

In this paper we present a semiautomatic method for the repositioning problem. It is semiautomatic because the user has to select the corresponding surface pairs on two different fragments which belong to the same fracture by clicking on them with the mouse. For a simple fracture, where the healthy bone was split into two fragments, there is only one surface pair to select, but a complex case can have up to 8 or 10. Assuming we have collected all the surface pairs, our algorithm registers them one by one. The registration is basically a search for the three translational and the three rotational parameters to move and rotate one fragment to match the other. We formulated this search as a function minimalization problem where our cost function is the sum of the distances of all the points on the first surface to their nearest neighbors on the second surface. To reduce the 6-dimensional search space we included constraints on both the translational and rotational components: we permit only 2 cm movement and 20 degrees rotation. In fact, the dislocation can be much higher than the above mentioned value, but we start the process by translating the centroids together to cancel out the initial distance. The function optimization method we used was a Evolutionary Algorithm with a population size of 400 and lack of improvement served as stop condition.

The described method matched a single surface pair with high accuracy. Because the selection was made by hand, the user himself introduced minor errors besides previously present inaccuracies caused by improper segmentation or surface simplification. These errors accumulate if the number of dependent surfaces pairs is high, especially with models containing circles in their object graph. Therefore a global optimization is necessary to finalize the transformations. In this global optimization all surface pairs are considered simultaneously which means the dimensionality of the search space grew to $(n-1)*6$ for n bone fragments. The cost function was the same as above, but the constraints were stronger assuming there are only small corrections to make.

Using this method the fragment repositioning task can be done faster than with the mouse in particular for complicated cases. However human verification of the result is still required.

References

[1] Balázs Erdőhelyi, Endre Varga, and Attila Kuba. Surgical Planning Tool with Biomechanical Simulation. In *Proceedings of the International Conference on Computer Assisted Radiology and Surgery (CARS)*, volume 2 of International Journal of Computer Assisted Radiology and Surgery, pages S262-S263, 2007.

Calculation and Application of the Dynamic Function Level Dependencies in Java Programs

Ferenc Fischer

Dynamic dependencies are dependencies occurring during program runs on inputs. These dependency sets are smaller than the static dependency sets calculated from the source code, hence they are more informative and they characterize better the behaviour of the analyzed program than their static counterparts if the inputs are good.

We can define a distance based on the dynamic call graph (Dynamic Function Coupling - DFC) [1]. By putting the functions that are farther and farther from the selected one into the dependency set we can approximate the ExecuteAfter set generalized by us. After ER, EA and EB relations used in the definition of DFC we can compute $ImpactSet^{(d)}$, $EAfterSet^{(d)}$ set and $EBeforeSet^{(d)}$ set. These sets contain the impacting and/or impacted functions in a given direction and distance with the cut-off value d for a set of functions. $ImpactSet^{(d)}$ can be useful in change impact analysis, $EAfterSet^{(d)}$ in regression testing and $EBeforeSet^{(d)}$ in debugging.

These approximate sets can be easier computed than the static and dynamic statement level dependency sets. The intuition behind our approach is that the 'closer' the execution of a function is to the execution of another function in some of the runs of the program, the more likely it is that they are really dependent on each other. We used Jadys developed at our department to produce statement level slices. These slices were raised to function level to validate the precision of our method (precise slice). We examined how the precision and the recall of the approximate dependency sets change compared to those of the precise slices. Our results show that the precision of the sets that contain only close functions (with distance 1) is significantly (twice) larger than the precision of the sets created by the original conservative method.

References

[1] Árpád Beszédes, Tamás Gergely, Szabolcs Faragó, Tibor Gyimóthy, Ferenc Fischer. The Dynamic Function Coupling Metric and Its Use in Software Evolution, *Proceedings of 11th European Conference on Software Maintenance and Reengineering (CSMR'2007)*, pages 103-112, 2007.

3-level Confidence Voting Strategy for Dynamic Fusion-Selection of Classifier Ensembles

Csaba Főző and Csaba Gáspár-Papanek

Using *Classifier Ensembles* in data mining seems to be the easiest and most widespread way of further increasing classification accuracy nowdays. There are different stages of using multiple classifiers, such as the meta-classifier stage treating the ensemble members as black boxes, and the stage on which the functionality of members is also important. Furthermore, the method of combining ensemble members can be done by fusion or selection of classifiers. In this paper, we propose a novel procedure for constructing *Multiply Classifier Systems* (MCSs) [2] on the meta-classifier stage with an *Oracle* of a 3-level voting strategy. We use a dynamic, half fusion-half selection type method for ensemble member combination, which is a midway procedure between the extremes of fusion and selection. The MCS members are weighted and combined with the help of an Oracle. Classification is done by selecting the MCS's members that has weights higher than a predefined weight limit, and afterwards, fusing the remaining single classifier outputs. Due to this, the key role in our method is played by the Oracle, which is founded on a voting strategy of three levels: The explicit global (1), the implicit semi-global (2), and the local confidence (3). The first is calculated using the validation part of the labelled training data, and it is used to obtain the general accuracy of the single classifier. The second confidence segment is derived during the classifiers' construction via the implicit knowledge gathered simultaneous with training. Since this strongly depends on the internal operation of the classifier, it can not always be obtained, in example, when using some particularly complex classification methods. And finally, the third and most important part of the confidence triple is created with the help of the unlabelled object yet to be classified. Due to this, all of the MCS members have the global (1) and semi-global (2) confidence attributes, that can be counted in construction time, but the local one (3) must be derived in classification time. The procedure is similar to [1], with the difference that we are interested in the precise local confidences instead of the best single classifier. First, the k nearest neighbour of the unlabelled object (UO) is located by the help of an appropriate distance metric over data objects. Next, we select those, that has a Multiply Classifier Behaviour (MCB) similar to the UO. This is done by comparing the UO's vector with k decision vectors, whose elements are the decisions of MCS members for the given data object. Finally, the remaining n objects' MCBs are used to calculate the ratio of the correct decisions and n for each single classifier, resulting the local confidence. Combining the triplet of "general accuracy", the "self confidence", and the "expertise on the particular field" of the object yet to be labelled leads to a greatly increased accuracy in classification.

References

[1] Giorgio Giacinto, Fabio Roli. Dynamic classifier selection based on multiple classifier behaviour, *Pattern Recognition 34 (2001) 1879–1881*.

[2] Csaba Főző. State of the Art: Classifier Ensemble Methods.

Potential Clouds in Robust Design

Martin Fuchs and Arnold Neumaier

The goal of robust design is to safeguard reliably against worst-case scenarios while seeking an optimal design. An engineer typically faces the task to develop a product which satisfies given requirements formulated as design constraints. Output of the engineer's work should be an optimal design with respect to a certain design objective. In many cases this is the cost or the mass of the designed product. An algorithmic method for design optimization functions as decision making support for engineers. The attempt of autonomous design has been made trying to capture the reasoning of the system experts. For more complex kinds of structures, e.g., a spacecraft component or a whole spacecraft, the design process addresses several different engineering fields, so the design optimization becomes multidisciplinary. An interaction between the involved disciplines is necessary. The resulting overall optimization process is known as multidisciplinary design optimization (MDO). Particularly MDO benefits from autonomous optimization methods for decision support that bridge the gap between different technical backgrounds. The difficulties arising during design optimization can be of most complex nature: multiobjective, multilevel, mixed integer nonlinear programming (MINLP) optimization problems with discontinuities or strong nonlinearities are involved. Standard optimization techniques cannot be used to solve such problems.

In many cases, in particular in early design phases, it is common engineering practice to handle uncertainties by assigning intervals, or safety margins, to the uncertain variables, usually combined with an iterative process of refining the intervals while converging to a robust optimal design. The refinement of the intervals is done by experts who assess whether the worst-case scenario, that has been determined for the design at the current stage of the iteration process, is too pessimistic or too optimistic. How to assign the intervals and how to choose the endpoint of the assigned intervals to get the worst-case scenario is usually not computed but assessed by an expert. The goal of the whole iteration includes both optimization of the design and safeguarding against uncertainties. The achieved design can thus be qualified as robust. Apart from interval assignments there are further methods to handle uncertainties in design processes: e.g., fuzzy clustering, simulation techniques like Monte Carlo.

Real life applications of uncertainty methods disclose various problems. The dimension of many uncertain real life scenarios is very high which causes severe computational problems, famous as the curse of dimensionality. Even given the knowledge of the multivariate probability distributions the numerical computation of the error probabilities becomes very expensive, if not impossible. Often standard simulation techniques are used to tackle the dimensionality issue, as the computational effort they require seems to be independent of the dimension. Advancements have been made based on sensitivity analysis, or on α-level optimization.

Frequently, especially in early design phases, data are scarce, though a large amount of data would be required to use traditional methods to estimate high dimensional probability distributions. Simulation techniques like Monte Carlo also require a large amount of information to be reliable, or unjustified assumptions on the uncertainties have to be made. However, mostly there are only interval bounds on the uncertain variables, sometimes probability distributions for single variables without correlation information. The lack of information typically causes standard simulation based methods to underestimate the effects of the uncertain tails of the probability distribution. Similarly, a reduction of the problem to an interval analysis after assigning intervals to the uncertain variables as described before (e.g., 3 σ boxes) entails a loss of valuable uncertainty information which would actually be available, maybe unformalized, but is not at all involved in the uncertainty model.

To overcome the problem of high dimensions in real life applications and the problem of incomplete information we make use of the concept of potential clouds. Potential clouds combine ideas from p-boxes, fuzzy sets, interval and optimization methods. The clouds can be weaved

into an optimization problem formulation as confidence regions constraints. The computational effort is still tractable in higher dimensions. Remarkably potential clouds even enable an a posteriori information update for experts, even if an expert is unable to give a formal description of his knowledge. Unformalized knowledge is available, e.g., if an expert does not know correlations exactly, but can formulate a statement like 'if variable a has a large value then variable b cannot have a low value'. Thus he is able to exclude irrelevant scenarios, although he is unable to give a formal description. This can be performed in a graphical user interface as an interaction between the uncertainty modeling and the optimization phase. The new information can be processed by means of potential clouds.

The basic concept of our methodology can be summarized in three essential steps within an iterative framework. First, the expert provides the underlying system model, given as a black-box model, and all a priori available uncertainty information on the input variables of the model. Second, the information is processed to generate a potential cloud. Parameterized by given confidence levels, the clouds provide a nested collection of regions of relevant scenarios affecting the worst-case for a given design and thus produce safety constraints for the optimization. Third, optimization methods minimize a certain objective function (e.g., cost, mass) subject to the functional constraints which are represented by the system model, and subject to the safety constraints from the cloud. To this end we have developed heuristic optimization techniques. The results of the optimization are returned to the expert, who is given an interactive possibility to provide additional uncertainty information a posteriori and to rerun the procedure, adaptively improving the uncertainty model.

References

[1] D. Dubois and H. Prade. *Possibility Theory: An Approach to Computerized Processing of Uncertainty*. New York: Plenum Press, 1986.

[2] S. Ferson. *Ramas Risk Calc 4.0 Software: Risk Assessment with Uncertain Numbers*. Lewis Publishers, U.S., 2002.

[3] M. Fuchs and A. Neumaier. Potential based clouds in robust design optimization. *Journal of statistical theory and practice, special issue on imprecision*, 2008. accepted, preprint available on-line at: `http://www.martin-fuchs.net/publications.php`.

[4] B. Moeller and M. Beer. Fuzzy randomness: uncertainty in civil engineering and computational mechanics. *Springer-Verlag Berlin Heidelberg*, 2004.

[5] A. Neumaier. Clouds, fuzzy sets and probability intervals. *Reliable Computing 10*, pages 249–272, 2004. `http://www.mat.univie.ac.at/ neum/ms/cloud.pdf`.

Introducing a Benchmark for Evaluating Reverse Engineering Tools

Lajos Jenő Fülöp, Péter Hegedűs and Rudolf Ferenc

Useful information can be discovered from source code with the examination of software systems. First step of the examination is to analyze the source code and making an *abstract representation*. In the second step different structures and problems can be discovered from the abstract representation, e.g. design patterns, code duplications and coding rule violations. These structures and problems are discovered by different kind of *reverse engineering tools*.

Reverse engineering tools present their results in different formats, which makes them very difficult to compare. In our previous works [5][6] we presented a tool called DEEBEE(DEsign pattern Evaluation BEnchmark Environment) which supports the evaluation and comparison of design pattern miner tools only. We realized that DEEBEE can be extended to support the evaluation and comparison of every kind of reverse engineering tools as well. In this paper we introduce a benchmark - called BEFRIEND (BEnchmark For Reverse engInEering tools workiNg on source coDe) - with which the outputs of reverse engineering tools can be evaluated and compared easily and efficiently. The extension of DEEBEE into a general benchmark BEFRIEND is motivated by other works which try to evaluate and compare code duplication finder tools [3][4] and rule violation checker tools[1][7] as well.

BEFRIEND supports different kinds of tool families, programming languages and software systems, and it enables the users to define their own evaluation criteria. Furthermore, it is a web-application open to the community and freely available [2]. We hope it will be accepted and used by the community in the future to evaluate and compare their tools with others.

References

[1] N. Ayewah, W. Pugh, J. D. Morgenthaler, J. Penix, and Y. Zhou. Evaluating static analysis defect warnings on production software. In *PASTE '07: Proceedings of the 7th ACM SIGPLAN-SIGSOFT workshop on Program analysis for software tools and engineering*, pages 1–8. ACM, 2007.

[2] DEEBEE and BEFRIEND Homepage. `http://www.inf.u-szeged.hu/designpatterns/`.

[3] S. Bellon, R. Koschke, G. Antoniol, J. Krinke, and E. Merlo. Comparison and Evaluation of Clone Detection Tools. In *IEEE Transactions on Software Engineering*, volume 33, pages 577–591, 2007.

[4] E. Burd and J. Bailey. Evaluating clone detection tools for use during preventative maintenance. In *Proceedings of the 2th International Workshop on Source Code Analysis and Manipulation (SCAM 2002)*, pages 36–43. IEEE Computer Society, 2002.

[5] L. J. Fülöp, R. Ferenc, and T. Gyimóthy. Towards a Benchmark for Evaluating Design Pattern Miner Tools. In *Proceedings of the 12th European Conference on Software Maintenance and Reengineering (CSMR 2008)*. IEEE Computer Society, Apr. 2008.

[6] L. J. Fülöp, A. Ilia, A. Z. Végh, P. Hegedűs, and R. Ferenc. Comparing and Evaluating Design Pattern Miner Tools. In *ANNALES UNIVERSITATIS SCIENTIARUM DE ROLANDO EÖTVÖS NOMINATAE Sectio Computatorica*. Department of Computer Algebra, Eötvös Loránd University, 2008.

[7] S. Wagner, F. Deissenboeck, M. Aichner, J. Wimmer, and M. Schwalb. An evaluation of two bug pattern tools for java. In *Proceedings of the 1st IEEE International Conference on Software Testing, Verification and Validation (ICST 2008)*, pages 1–8. ACM, 2008.

Determination of Geometric Features of Binary Images from Their Projections by Using Decision Trees

Mihály Gara and Péter Balázs

Binary tomography produces two-dimensional cross-sections of three-dimensional homogeneous objects from their projections. Several algorithms developed for solving this task. Most of them presume that the binary image to be reconstructed has some special geometric features. In order to choose the appropriate reconstruction algorithm it is necessary to have a priori information of the image to be reconstructed. In this way we can improve the speed and reduce the ambiguity of the reconstruction.

Our work is concerned with the problem of retrieving geometrical information from the projections themselves. We investigate whether it is possible to evolve geometric features of binary shapes if only their projections are known. Most of the reconstruction algorithms based on geometrical information presuppose hv-convexity or connectedness about the image to be reconstrcuted. We investigate those properties in detail, and also the task of separating 4- and 8-connected shapes. We generate experimental images, then we try to retrieve the proper information just from the projections of the generated images.

For the classification we use decision trees. We decided to apply this learning method for three main reasosns. Our goal is to improve the speed of the reconstruction by finding the appropriate reconstruction algorithm. Thus, the determination of the feature must not take much time. Furthermore, in practice usually just a limited number of training data is available, so the learning must be able to work well even with relatively few training data. Finally, we want to exploit that classifications given by decision trees can be expressed by if-then rules.

We show that the separation of hv-convex images from randomly generated ones just by studying their projections is quite an easy task. Surprisingly, we obtain good results for classifying hv-convex and "almost" hv-convex images as well. Those experiments help us to gain useful information concerning the characterization of the discussed and other geometrical properties.

An Identification Approach to Dynamic Errors-in-variables Systems with a Preliminary Clustering of Observations

Levente Hunyadi

Errors-in-variables models are statistical models in which not only dependent but also independent variables are observed with error, i.e. they exhibit a symmetrical model structure. The application field for these models is diverse including computer vision, image reconstruction, speech and audio processing, signal processing, modal and spectral analysis, system identification and astronomy [1].

One particular application of errors-in-variables models is to parameter estimation of discrete-time dynamic linear systems. In this configuration, only the noise-corrupted input and output signals are observable, the original noise-free signals are not. As far as the additive noise is concerned, in most cases, a white noise model is assumed, which corresponds to noise due to measurement error. Given this system model, the goal of system identification is to derive process as well as noise parameters using the observable noise-contaminated input and output signals.

Provided that the ratio of input and output noise variances is a priori known, the task of deriving process and noise parameters is a classical system identification problem. In contrast, a situation where no such information is available is recognized as a more difficult one. In fact, it turns out that under the assumption of Gaussian distributed input and noise signals, the system is not identifiable by second order properties only (i.e. the autocorrelation function in time domain or the power spectrum in frequency domain). In other words, identification results in many indistinguishable solutions, so that for the identification to produce a unique result, additional restrictions are necessary [2].

One possible restriction is to assume that observations can be clustered into two sets which expose different characteristics. Provided that such a clustering is possible, identification can be broken down into parameter estimation over two separate sets of data. As a result, the estimation procedure produces two sets of process parameter estimates, the distance of which can be measured using some distance metrics. Consequently, varying the ratio of input and output noise variances, and deriving process parameter estimates for each case, minimizing the distance metrics yields an estimate for the noise variance ratio, resolving the identifiability issue.

The feasibility of the approach is demonstrated by a simulation example. Clustering is performed by principal component analysis [3] and estimation is based on a generalized Koopmans-Levin method [4]. The performance of the approach is also compared to the Cramér–Rao statistical lower bound [5], which gives the theoretical value for the best possible estimate. The algorithmic aspects of the simulation environment are implemented in MatLab with a visual front-end realized in C#.

References

[1] Sabine Van Huffel and Philippe Lemmerling, editors. TLS and EIV modeling. *Kluwer*, 2002.

[2] Torsten Söderström. Errors-in-variables methods in system identification. *Automatica*, 43:939–958, 2007.

[3] Chris Ding and Xiaofeng He. K-means clustering via principal component analysis. In *Proceedings of the 21st Int. Conf. on Machine Learning*, page 29, New York, NY, USA, 2004. ACM.

[4] István Vajk. Identification methods in a unified framework. *Automatica*, 41(8):1385–1393, 2005.

[5] Erlendur Karlsson, Torsten Söderström, and Petre Stoica. The Cramér-Rao lower bound for noisy input-output systems.*Signal Processing*, 80(11):2421–2447, 2000.

Cohesion and Balance in a Human Resource Allocation Problem

László Illyés

The problem discussed occurred when many construction workers were taken to Germany from Romania for labor in construction projects[1]. Managers have to make independent groups of workers from some categories, like carpenters, brick layers etc. They have to discover their collaborative attitudes. One method is that every worker scores the others how much he wants to work with them. There are other methods like personality tests but in this case that approach was expensive and time consuming. No scoring was necessary inside of same trade of workers. The objectives were to form groups of workers with high compatibility and to minimize the difference of compatibility scores between the most fitted and the least fitted group. The problem becomes more interesting in making software collaborative groups where there are database administrators, operating system specialists, security specialists and so on. One have to prospect also the level of knowledge, the possibility and ability to increase this knowledge and the overlap between the trade groups of workers, for example the database administrator may also be a good Linux specialist and we can exploit both aptitudes inside a group. This paper resumes to the problem of construction workers where there is no overlap between trades and the level of knowledge is not in the universe of discourse. We propose a Greedy and a genetic algorithm approach and we compare these methods. In [2] we treat the problem when we have no categories and every student has to be allocated in a group. We consider that the Greedy algorithm found is a very interesting one because it follows both objectives at the same time.

Acknowledgments

This work was partially supported by the Department of Mathematics and Informatics of the Faculty of Business and Humanities, Sapientia University, Csíkszereda, Romania

References

[1] Fábián, Cs. B., Private conversations, 2003-2006.

[2] Illyés L., (2007), Balanced students groups forming for university projects using genetic algorithm, The proceedings of the eight international conference on informatics in economy, ASE Printing House, Bucharest,pp. 554-559.

Efficient Static Impact Analysis

Judit Jász

Impact analysis has a main role in many software engineering tasks such as software maintenance, regression testing, debugging. We present an alternative way to determine the impact sets of methods by applying the *Static Execute After (SEA)* relation among them. The methods f and g are in SEA relation if any part of the method g could execute any part of the method f. This technique is suitable for real size programs with over thousands lines of code where the application of a slicer, which is considered as the most precise solution in impact analysis, cannot be carried out. This way, the technique could be widely used in the field of impact analysis.

To compute the SEA relations we introduce the *ICCFG* graph, which is a more detailed representation of the program than a simple call graph, but more compact than the system dependence graph, which is the base of slicing. We present a suitable traversal on the ICCFG graph which determines the SEA set of a particular method and, in addition, we give an algorithm that computes the sets of these relations for each method in the same time.

We prove with experimental results that the computed sets can approximate the sets of sliced methods computed by a precise slicer, and this way, we can approximate the real impact sets as well. Since the introduced algorithms are based on much fewer dependencies than the slicer, and they work on a more compact graph representation, they are more convenient to use especially in the case of large, real size programs. At the same time, the precision declines very slightly, by some 4% on average.

An interesting spinoff of our experimental results was the fact that the dependencies computed by the nowadays used slicers can be more conservative than it would be rational. The most important consequence of this behavior is that in some cases the forward slice is not the dual representation of the backward slice, which is anyway an expected property. We also investigate this problem.

Keywords: *slicing, execute after relation, impact analysis*

High Precision Pedestrian Recognition in Low Resolution Images

Zs.T. Kardkovács, Cs. Gáspár-Papanek, G. Szabó, E. Sipőcz and G. Pécsi

Proper recognition of objects in images is one of the most important tasks of next generation intelligent systems. For example, identification of pedestrians plays a key role in building intelligent vehicles, surveillance robots, etc. To this end, NiSIS (Nature Inspired Smart Information Systems) announced a competition in 2007 which was aimed at boosting researches in this specific area. During the competition a large amount of low resolution (36x18 pixels) 8-bits grayscale still images were available based on DaimlerChrysler (DC) pedestrian classification benchmark [1]. The problem consists in the detection of pedestrians against a background or other pedestrian-like objects. Our solution proved to be the best among 17 participants from 14 countries.

DC pedestrian benchmark contains five sets of non-correlated still images. That is, neither pedestrians nor objects appear in more than one set, however, pedestrians may appear in several images mirrored or shifted by 2-4 pixels both vertically and horizontally within the same set. NiSIS made the benchmark more interesting by clearing (replacing adequate color codes by 205) a part of some images as if they were broken. In order to fulfill criteria of the benchmark our solution consists of four main modules: data preprocessing algorithms, image preprocessing techniques, classifiers, and ensemble methods.

Data preprocessing algorithms (DPA) are responsible for standardizing images as much as possible, i.e. all next processing techniques should not depend on mirroring, shifting, and other possible transformations. DPA discovered that images show pedestrians are common in having similar color values near the head and „shoe" regions. Nevertheless, the difference between common pedestrian and non-pedestrian images indicated that the real distinction between images can be captured near the sky region. We assume this was produced by the lens and auto-focusing methods of the camera since certain pixels have the same (reference) color codes in almost all images.

Image preprocessing techniques are responsible to identify common characteristics of object and filter foreground object from background. We assumed four approaches can help in solving data analysis discovered subproblems: we used an edge detection algorithm, a normalized skeletonization method, image based Z-normalization, and singular value decomposition (SVD). We used edge detection to find silhouettes of objects in images. We hoped skeletons determine basic shape of an object, and provide us hints how to transform them into a well-identified pedestrian's skeleton. Skeletons were also inputs of well-known neural network classifiers. Since neural networks are usually sensitive for input shifting, we normalized the skeletons in the sense that the skeleton element p closest to the pixel $p_0 = <9;\ 18>$ is shifted to p_0, Z-normalization is widely applied for time series where intensities, trends, vertical shifting should be avoided. We found a 5–10% improvement on Z-normalized data results. SVD is used as a geometric transformation which determines proper orthonormal basis for each of the images. SVD is invariant for horizontal (and vertical) shifting, thus feature matrices of images were the most important input of classifiers.

Three classifiers were applied our solution: k nearest neighbor (kNN, with $k = 11$), a feed-forward multi layer Adaline network (two hidden layers, 5–5 sigmoid neurons in each), and an extreme learning machine (ELM) with Gaussian weights (a single layer network with 300 hidden neurons).

Our classifiers provide confidence like metrics between 0 and 1. We enhanced our model to evaluate both possibilities independently by each of the classifiers, and classifiers must predict confidences of both evaluations. One of the ensemble method is firmly used the sum of confidence weighted prediction for images. Since all classifiers perform very similarly, additional weighting was not necessary. While this voting strategy is very effective confidence values cannot be compared in general hence they have no common absolute scale or common

background. Therefore, we also applied a feed-forward neural network to evaluate classifier confidence values, judgment and some other parameters. Since judgments of neural network based ensemble overperformed the other by 3-5%, we agreed on its judgment for pedestrians.

Our method was trained on the train data set given by the downloaded data. We used labeled test data for both validating and testing our model by splitting the set into two equal size parts. That is, we had a train data set denoted by S_1, a test set S_2 and a validation S_3. Each performance of the applied methods is seen in Table 1 where performance was calculated by the formula:

$$P(c) = \frac{\text{correct judgements by classifiers}}{\text{the number of images}}.$$

Our solution was found to be the most accurate (96.04%) on the NiSIS competition therefore we received a special award.

	SVD + kNN	Adaline	ELM	Simple ensemble	Neural ensemble
no modification	84%	84%	83%	91%	93%
edge detection	74%	74%	72%	78%	81%
skeletonization	76%	75%	73%	80%	83%

Table 1: Performances of classifiers trained on $S_1 + S_2$, tested on S_3

Acknowledgements

Our work was sponsored by Mobile Innovation Centre through the Asbóth Oszkár Programme of the Hungarian State.

References

[1] S. Munder, D. M. Gavrila. An Experimental Study on Pedestrian Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 11, November 2006.

An Order-Independent Sequential Thinning Algorithm

Péter Kardos and Gábor Németh

Skeletons are region-based shape descriptors which summarize the general form of objects/shapes. An illustrative definition of the skeleton is given using the prairie-fire analogy: the object boundary is set on fire and the skeleton is formed by the loci where the fire fronts meet and extinguish each other [1]. Thinning is a frequently used method for making an approximation to the skeleton in a topology–preserving way [2]. It is based on a digital simulation of the fire front propagation: the border points of a binary object that satisfy certain topological and geometric constraints are deleted in iteration steps. The entire process is then repeated until only the "skeleton" is left. Most of the existing thinning algorithms are parallel as the fire front propagation is by nature parallel, but numerous sequential ones were also proposed [3].

A simple point is an object point whose deletion does not alter the topology of the picture [2]. Sequential thinning algorithms use operators that delete some simple points which are not end points, since preserving end–points provides important geometrical information relative to the shape of the objects. Unfortunately, the existing algorithms produces various results for different (say forward and backward) scans of the points.

We propose a new sequential thinning algorithm based on the notion of critical pairs of points. Let p and q be two 4-neighboring object points. We define the $\{p, q\}$ set as critical pair if the following conditions are satisfied:

- $C(p) = C(q) = 1$,
- $3 \leq B(p) \leq 6$ and $3 \leq B(q) \leq 6$,
- $C(p, q) = 2$,

where $C(p)$ is the number of black components in the neighborhood of p , $B(p)$ denotes the number of black points in the 8-neighborhood of p , and $C(p, q)$ is the number of black components in the 8-neighborhood of $\{p, q\}$.

Using a special classification of critical pairs, some statements can be made about neighbouring object points. With the help of these rules, we define the pixel removal conditions of our algorithm. Furthermore, we use an additional matrix to store the previously deleted points in the current iteration. This information also helps us to decide if a point is erasable or not.

It is proven that our algorithm is topology preserving for $(8, 4)$ binary pictures and order independent. Results on two test pictures produced by the proposed algorithm are presented in Fig. 1.

References

[1] H. Blum, A transformation for extracting new descriptors of shape, *Models for the Perception of Speech and Visual Form*, MIT Press,1967, 362–380.

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

[3] L. Lam, S.-W. Lee, C.Y. Suen, Thinning methodologies – A comprehensive survey, IEEE Trans. *Pattern Analysis and MachineIntelligence* 14, 1992, 869–885.

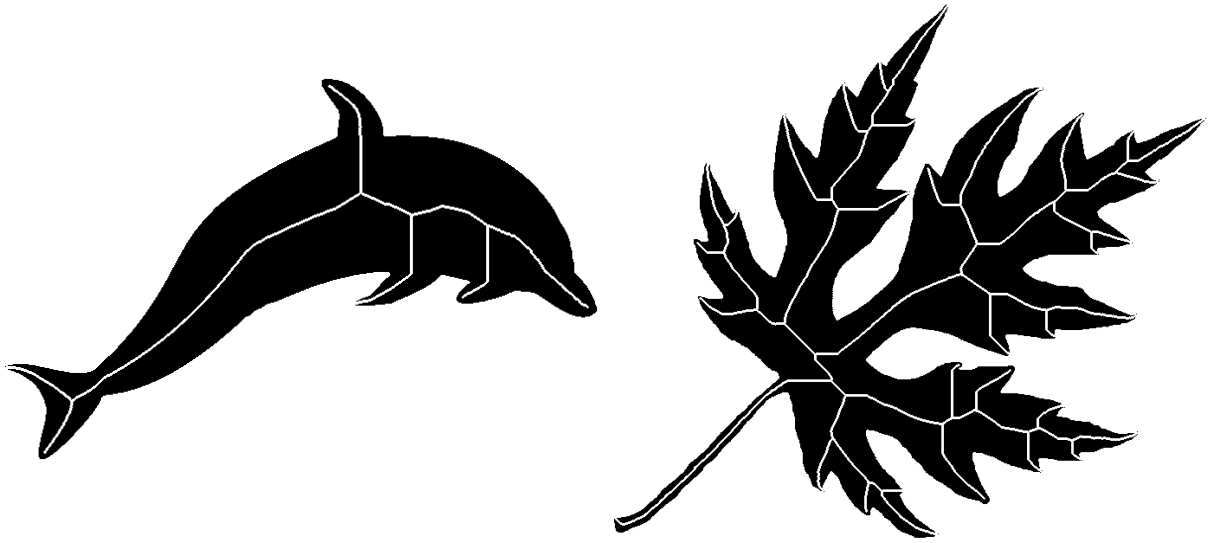

Figure 1: A 612×398 image of a dolphin (left) and a 700×633 image of a leaf (right). The "skeletons" produced by our algorithm are superimposed on the original images.

Adaptive Scheduling Solution for Grid Meta-brokering

Attila Kertész, József Dániel Dombi and József Dombi

Since the management and beneficial utilization of highly dynamic grid resources cannot be handled by the users themselves, various grid resource management tools have been developed, supporting different grids. To ease the simultaneous utilization of different middleware systems, researchers need to revise current solutions. User requirements created certain properties that resource managers have learned to support. This development is still continuing, and users already need to stress themselves to distinguish brokers and migrate their applications, when they move to a different grid. Interoperability problems and multi-broker utilization have emerged the need for higher level brokering solutions. The meta-brokering approach means a higher level resource management by enabling automatic and simultaneous utilization of grid brokers. The Grid Meta-Broker [1] is a high-level middleware service that uses meta-data about user request and broker capabilities to find a proper execution environment.

Scheduling at this level requires sophisticated approaches, because high uncertainty presents at all stages of grid resource management. This work presents an adaptive scheduling algorithm that enhances the matchmaking service of the Meta-Broker and enables it to react to unreliable broker properties and load unbalances. The algorithm uses an initial teaching phase to gather reliability information of the interconnected brokers. We keep the first phase of the original matchmaking algorithm of the Meta-Broker: this is used to filter out brokers that are lacking capabilities required by the user. The extension appears in the second phase, where a fitness function is used to calculate the goodness of the brokers. Different environments may require different weights in the fitness function of the algorithm. We developed a method to monitor the behavior of the scheduling algorithm and modify the appropriate weights to find the optimal fitness function. After defining the weights for the actual environment, we are able to calculate the goodness values. This is done with the help of a random generator function [3],[4], which takes into account dynamic historical performance data of the brokers.

We experimented in a simulated grid environment built on top of the GridSim toolkit [2]. It supports modeling and simulation of heterogeneous grid resources, users, applications, brokers and schedulers in a grid computing environment. It provides primitives for the creation of jobs, mapping these jobs to resources and managing them, therefore resource schedulers can be simulated with this tool to study scheduling algorithms. In our general simulation architecture resources can be defined with different grid middleware types. Resources consist of more machines, to which workloads can be set. On top of this simulated grid infrastructure we can set up resource brokers. Brokers are extended GridUser entities: they can be connected to one or more resources, they report to the IS Grid load database (which has a similar purpose as a grid Information System), different properties can be set to these brokers (agreement handling, co-allocation, advance reservation, etc.), properties can be marked as unreliable, different scheduling policies can be set for each broker and resubmission is used, when a job fails due to resource failure. The Simulator is an extended GridSim entity: it can generate a requested number of jobs with different properties, start and run time. It is connected to the Grid Meta-Broker through its web service interface and able to call its matchmaking service. Before the Meta-Broker is used for broker selection, it has to be configured with the capabilities of the interconnected brokers. Our evaluation results show that the Meta-Broker performs better with the enhanced adaptive algorithm: the total makespan of the submitted jobs can be significantly reduced.

Acknowledgements

This research work is partly supported by the FP7 Network of Excellence S-Cube project funded by the European Commission (Contract FP7/2007-2013).

References

[1] A. Kertész, P. Kacsuk: Meta-Broker for Future Generation Grids: A new approach for a high-level interoperable resource management, *CoreGRID Workshop on Grid Middleware in conjunction with ISC'07 conference*, Dresden, Germany, June 25-26, 2007.

[2] R. Buyya and M. Murshed: GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing, *Concurrency & Computation: Prac. & Exp.*, Nov-Dec 2002.

[3] I. Deák: Véletlenszám-generátorok és alkalmazásuk, *Akadémiai Kiadó*, 1986, Budapest.

[4] J.P.C. Kleijnen, B. Annik: Pseudorandom number generators for supercomputers and classical computers? A practical introduction, *European Journal of Operation Research*, 63 (1992) 76-85.

Skeletonization Based on Neighborhood Sequences

György Kovács and Gábor Németh

Skeletonization provides shape features that are extracted from binary image data [1]. It can be used in raster–to–vector conversion, topological analysis, shape simplification, or feature tracking. In the 2D Euclidean space, the skeleton consists of the locus of the centers of all the maximal inscribed disks [2].

Mathematical morphology is a powerful tool for image processing and image analysis [6]. Its operators can extract relevant topological and geometrical information from binary (and grey-scale) images by using structuring elements with various sizes and shapes. The discrete skeletons can be characterized morphologically: the centers of all maximal inscribed disks can be expressed in terms of erosions and dilations.

The morphological skeleton of set $X \subseteq \mathbb{Z}^2$ by structuring element $Y \subseteq \mathbb{Z}^2$ is defined by

$$S(X,Y) = \bigcup_{k=0}^{K} S_k(X,Y) = \bigcup_{k=0}^{K} (X \ominus Y^k) - ((X \ominus Y^{k+1}) \oplus Y) \,,$$

where $\ominus$ and $\oplus$ denote respectively the erosion and the dilation, and

$$Y^k = \begin{cases} \{(0,0)\} & \text{if } k = 0 \\ \{(0,0)\} \oplus Y = Y & \text{if } k = 1 \\ Y^{k-1} \oplus Y & \text{otherwise} \end{cases},$$

and K is the last step before X is eroded completely:

$$K = \max\{\ k \ \mid \ X \ominus Y^k \neq \emptyset\ \}.$$

The formulation states that $S(X,Y)$ is obtained as the union of the skeletal subsets $S_k(X,Y)$. It can be readily be seen that the set $S_k(X,Y)$ contains all points $p \in X$ such that x is the center of a maximal "disk" included in X. Note that the limitation of the morphological skeleton is that its construction is based on "disks" of the form Y^k. Hence the morphological skeleton does not provide a good approximation to the Euclidean skeleton.

In the digital space $\mathbb{Z}^2$, two types of motions are considered [5]. The cityblock motion (denoted by $\mathbf{1}$) allows horizontal and vertical movements only, while in the case of chessboard motion (denoted by $\mathbf{2}$) one can diagonal movements, as well. The octagonal distances can be obtained by the mixed use of these motions. The sequences of cityblock and chessboard motions are called neighborhood sequences [3]. Some of them generate metrics on the digital space $\mathbb{Z}^2$ [4].

In order to cut the shortage of the morphological skeleton, we propose a new type of skeleton that is based on neighborhood sequences.

Let $A = (A(i))_{i=1}^{\infty}$ be a $2D$ neighborhood sequence (where $A(i) \in \{\mathbf{1}, \mathbf{2}\}$) and let $\mathcal{Y} = (Y(i))_{i=1}^{\infty}$ be the sequence of structuring elements in which $Y(i)$ corresponding to $A(i)$ is defined by

$$Y(i) = \begin{cases} \{(0,0),(-1,0),(1,0),(0,-1),(0,1)\} & \text{if } A(i) = \mathbf{1} \\ \{(0,0),(-1,0),(1,0),(0,-1),(0,1),(-1,-1),(-1,1),(1,-1),(1,1)\} & \text{if } A(i) = \mathbf{2} \end{cases}.$$

The sequence skeleton of set $X \subseteq \mathbb{Z}^2$ by sequence of structuring elements $\mathcal{Y} \subseteq \mathbb{Z}^2$ is defined by

$$\mathbb{S}(X,\mathcal{Y}) = \bigcup_{k=0}^{K} (X \ominus \mathcal{Y}^k) - ((X \ominus \mathcal{Y}^{k+1}) \oplus Y(k+1)) \,,$$

where

$$\mathcal{Y}^k = \begin{cases} \{(0,0)\} & \text{if } k = 0 \\ \{(0,0)\} \oplus Y(1) = Y(1) & \text{if } k = 1 \\ \mathcal{Y}^{k-1} \oplus Y(k) & \text{otherwise} \end{cases}$$

and

$$K = \max\{k \mid X \ominus \mathcal{Y}^k \neq \emptyset\}.$$

Note that the sequence skeleton can be defined in arbitrary dimensions. It is easy to see that

$$\mathcal{S}(X, \mathcal{Y}) = S(X, Y)$$

if $\mathcal{Y} = (Y, Y, \ldots)$. Hence the conventional morphological skeleton is a special case of sequence skeletons.

A novel method for quantitative comparison of skeletonization algorithms is also proposed. According to our experiment, sequence skeletons can provide much closer approximations to the Euclidean skeleton than conventional morphological skeletons do.

References

[1] H. Blum. A transformation for extracting new descriptors of shape, *Models for the Perception of Speech and Visual Form*, MIT Press,1967, 362–380.

[2] L. Calabi. A study of the skeleton of plane figures, Technical report 60429, Parke Mathematical Laboratories, 1965.

[3] P.P. Das. Lattice of octagonal distances in digital geometry, *Pattern Recognition Letters* 11, 1990, 663–667.

[4] A. Fazekas, A. Hajdu, L. Hajdu. Metrical neighborhood sequences in $\mathbb{Z}^n$, *Pattern Recognition Letters* 26, 2005, 2022–2032.

[5] A. Rosenfeld, J.L. Pfaltz. Distance functions on digital pictures, *Pattern Recognition* 1, 1968, 33–61.

[6] J. Serra. Image anaysis and mathematical morphology, Academic Press, London, 1982.

Modeling Web Service Interaction Using Markovian Processes

Csaba Legány, Raluca Popa and Márk Kaszó

Nowadays E-Market forms a crucial part of our life; there are a lot of E-Commerce sites available worldwide. Customers of an E-Commerce site interact with requests in a session. A session identifies a navigation pattern through the sites. These customers use different navigation patterns, thus they make different workload for E-Commerce systems. Performance is one of the main challenges in designing an E-commerce or E-business application. To identify the performance for these systems, workload has to be determined. Our aim is to characterize typical user workload to reproduce user behavior before the site is developed, even during a modeling process. This article proposes mathematical models describing E-Commerce Systems along with a new method to estimate one important parameter of system workload, the average visit length. Several techniques can be applied to characterize system workload. One of these techniques is to analyze the Customer Behavior Model Graph while another approach is to use Markov Chains. A simple transformation has to be applied to transform CBMG graphs to stable Markov chains. Further methods include using mathematical models like Phase-type distributions or QBD (Quasi Birth Death Process) processes. In order to use these methods a transformation has to be applied to estimate the intensity matrix belonging to the CBMG graph. Finally typical E-Commerce systems are modeled using MAP / PH / 1 processes because service process can be modeled as Phase-type distribution and MAP is the one of the most general arrival processes. Average queue-length, waiting time and other characteristics of these E-Commerce systems can be calculated using these mathematical models. It is illustrated with measurement results that average visit length converges to the stationary distributions of the Markov chain representations.

Acknowledgements

This paper is part of the Performance Monitoring project of the Department of Automation and Applied Informatics of Budapest University of Technology and Economics.

References

[1] R.Jain: The art of computer systems performance analysis. New York: Wiley, 1991

[2] Daniel A. Menasce: Scaling for E-Business. *Prentice Hall* 2000 ISBN: 0-13-086328-9

[3] A. Horvath, M. Telek: Approximating heavy tailed behaviour with Phase type distributions, in *Advances in Algorithmic Methods for Stochastic Methods, G. Latouche and P. Tayler*, pages 191-214, Notation Publication, 2000

[4] A. Horvath, G. Rozsa, M. Telek: A MAP Fitting Method to Approximate Real Traffic Behavior, in *Proceedings of IFIP Workshop on Performance Modelling and Evaluation of ATM & IP networks*, UK, 2000.

[5] Cathy H. Xia, Zhen Liu, Mark S. Squillante, Li Zhang, N. Mabuch: Traffic Modeling and Performance Analysis of Commercial Web Sites, 2001

[6] J. Cao, M. Andersson, Ch. Nyberg, M. Kihl: Web Server Performance Modeling Using an M/G/1/K*PS Queue, IEEE, 2003

[7] A. N. Dudin, A.V. Kazimirsky, V.I. Klimenok: The Queueing Model MAP/PH/1/N with Feedback Operating in a Markovian Random Environment, *Austrian Journal of Statistics*, Volume 34, nr. 2, 101-110, 2005

[8] G. Latouche, V. Ramaswami: Introduction to Matrix Analytic Methods in Stochastic Modeling.

In-memory Preprocessing of Streaming Sensory Data Partitioned Relational Database Approach

József Marton and Sándor Gajdos

As the application area of sensors broadens amount of streaming sensory data to be processed is also increasing. Analytical systems based on heterogeneous sensory data require efficient preprocessing techniques hence integrated view of different data sources must be provided in order to calculate data in a short time period. In this paper, we present a partitioned database approach - i.e. a composite of memory-resident and disc-resident partition. This approach can provide two to five-times performance improvement in the data preprocessing stage. It can adapt to hardware resources efficiently, i.e. adding more CPU cores and random access memory increases performance in a near-linear fashion. This is especially the case when data streams to be integrated are independent of each other - which is a common premise.

Off-line transformations have at least two drawbacks. Temporary storing of data at some staging area not only wastes I/O and communication bandwidth but also decreases usefulness as data loose their freshness. That's why, along with recent research and development, in this paper we focus on near real-time transformation and integration of data into the database (repository). Near real-time preprocessing demands for high computational capacity which can be provided using several different approaches. Our proposal follow a resource-sensitive way.

Architecture of the proposed database backend, as seen on Figure 1, is a two-tier one. A traditional disc-resident database management system provides the long-term storage. On the top of this 1^{st} layer works a memory resident database. This means that the working set of the current data and a recent portion of the historical data reside in the physical memory. Application and transformation logic layers see a coherent view of the database and can neglect the disc-resident and memory-resident partitioning scheme applied.

As fresh data arrive into the database, it is stored in the memory-resident partition. This way it is available in the memory address space of the application driving the preprocessing. It is guaranteed by the partitioning scheme that no additional I/O cost arise during the transformation. This means that the cost of this phase solely depends on the function applied. The result of the transformation is appended to the repository of data used for instant monitoring and prediction. The proposed partitioning scheme also guarantees that data frequently queried for is available in the memory-resident portion thus enabling near real-time analysis.

A test system has been implemented to measure the performance gain of the proposed database architecture. We found that using a consumer class PC the proposed scheme was able to process data at 22% of the specified performance. To provide an easily comparable measure, we have replaced the partitioned database in our implementation by a solely disc-resident database tier, and found that it worked at a performance of 10% utilizing the same hardware. In contrast, the original 3rd party implementation that ran on a server platform processed just at a rate of 5%.

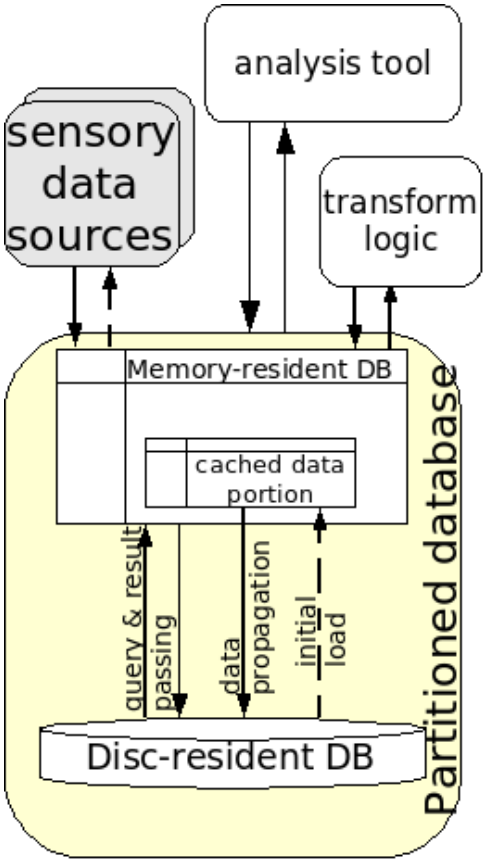

Figure 1: Architecture of the proposed partitioned database in its context.

Call Graph and Data Flow Analysis of a Dynamic Functional Language[1]

Tamás Nagy, Zoltán Horváth, László Lövei, Melinda Tóth and Anikó Nagyné Víg

Refactoring is about improving the design of existing program code: making changes to the source code which preserves the meaning of the program in order to improve non-functional characteristics of the code like readability or maintainability.

In refactoring, the highest amount of work is usually the precondition checking, which makes sure that the refactoring does not change the behaviour of the system. Compared to precondition checking an uncomplicated transformation is almost straightforward. To check whether the preconditions are met, the type system of the language can provide very useful information. Our primary research areas are those functional languages which do not have a static type system. Our aim is to find static analysis techniques which can provide enough information to check whether preconditions are met. Two examples of such techniques are call graph and data flow analysis.

Call graph analysis aims to give the exact function dependency relations in a given amount of code. This means that the wider the scope of the analysis the more accurate the result will be, but the smaller scope's call graph can never be invalidated by the new data, just new edges can appear.

Data flow graph on the other hand aims to give back the flow of data through the functions [6]. In other words it shows how parameters, global variables are used, passed around the system.

In Erlang to have an accurate call graph data flow analysis has to be done as well. This is the result of the highly dynamic nature of the language, and because the way functions are treated. For example it is possible to call functions from data we receive from different parts of the system using the `erlang:apply` built-in function. The source code of such data does not clearly show that it will be used in a function call. In fact in different parts of the system it can be used for a different purpose. This is because functions are identified by `atoms`. `Atoms` can be created dynamically, for example with the `erlang:list_to_atom` function.

There are a subset of function calls which do not need data flow analysis. These are the static function calls. Where every element of the function call are known at compile time.

The dynamic calls – where the called function is not known at compile time – can be further categorized based on how much information is present at compile time. Naturally the less information given the harder the analysis is. There are some edge cases where the analysis is impossible. In these cases the analysis' aim is to limit the possible functions which the call could refer to.

The static analysis aims to create the function call graph of the static calls. This analysis is straightforward if we use the results of the semantical analysis which is incorporated in our refactoring system [3].

Collecting the function calls, then sorting based on which function they are in and which function they call, is essentially the work that has to be done to create the call graph for the static calls.

While building the static call graph is a relatively lightweight job, the dynamic call graph building takes significantly more time and resources. Of course the dynamic analysis possibly adds more data to the graph resulting in a more accurate call graph.

The analysis method is based on the Observer design pattern; which means we have entities (variables, atom, tuples etc.) that are loosely connected to each other. During the analysis new connections and entities are created as well. Connection represents the dependency between certain entities.

[1]Supported by ELTE IKKK, Ericsson Hungary

When an entity finds out more information about itself (for example a variable about its possible values) it sends this information to the other entities which are connected to it. Because connections can be created to an entity after its analysis is completed, the entity is not deleted after its analysis is finished.

Entities are modeled with Erlang processes, and data propagation with message sending. This approach creates the opportunity of parallel computation, because there is no strong order to how the entities should be processed. There is one further advantage which is recomputation after changes happen. If there are changes in the underlying code, the graph can be adjusted by creating and deleting entities and edges. In other words there is no need to recompute the whole graph.

To start the analysis we need initial entities which will be further analysed, these are the dynamic calls unknown values. For example the values of the `erlang:apply/3` function's parameters. Further analysis is done by investigating entity types and surroundings. This could result in new edges and new entities which need to be further analysed. When the value of the initial entities is found out with the analysis it is made available through the `call_graph` applications interface in the same way as the static call graph data.

In general the result of the call graph and data flow can be widely used by the precondition checks of the different refactorings [1, 2, 4, 3, 5].

By creating a different interface to retrieve the existing data of the static analysis, the data collection part for automatic detection of refactoring opportunities can be easier. Of course this interface will be used to retrieve the dynamic function call data as well. By providing a common interface for the two different data we further ease data collection complexity.

References

[1] C# Refactory homepage. `http://www.xtreme-simplicity.net/`.

[2] Eclipse Project homepage. `http://www.eclipse.org/`.

[3] Erlang Official homepage. `http://erlang.org/`.

[4] H. Li, S. Thompson, and C. Reinke. The Haskell Refactorer, HaRe, and its API. *Electronic Notes in Theoretical Computer Science*, 141(4):29–34, 2005.

[5] R. Szabó-Nacsa, P. Diviánszky, and Z. Horváth. Prototype environment for refactoring Clean programs. In *The Fourth Conference of PhD Students in Computer Science (CSCS 2004), Vol. of extended abstracts*, page 113, Szeged, Hungary, July 2004. (full paper: `http://aszt.inf.elte.hu/~fun_ver/`,10 pages).

[6] G. Naumovich. Using the observer design pattern for implementation of data flow analyses. *SIGSOFT Softw. Eng. Notes*, 28(1):61–68, 2003.

Parameter Learning Algorithms in Online Scheduling

Tamás Németh and Csanád Imreh

The effeciency of online scheduling algorithms is usually measured by the competitive analysis, which is a worst case bound on the ratio of the cost of the online algorithm, and the optimal offline cost. Only a few result exists which investigate the average behavior of the algorithms on real inputs or on randomly generated inputs. Such results are presented for the multiprocessor makespan minimizing scheduling problem in [1], where it is shown that the simple List scheduling algorithm is not worst from the average case point of view than the much more difficult algorithms with better competitive ratio. In some scheduling model parametrized algorithms are developed and such parameter value is used which yields the best worst case bound for the algorithm. In this paper we investigate the question whether is it possible to improve the average efficency by an algorithm which learns the value of the parameter which results the smallest cost. We consider the problem of online multiprocessor scheduling with rejection which problem can be defined as follows.

There are jobs, and each job has a processing time and a penalty. The problem is online, the jobs arrive one by one and when a job arrives the decision maker either has to schedule it on one of the machines or to reject the job. The cost of the schedule is the sum of the makespan and the total penalty of the rejected jobs. This problem was defined in [2], and a class of algorithm depending on a parameter α is given for the solution of the problem. The optimal value (considering the competitive ratio) of the parameter is determined for any fixed number of machines. We define the following algorithm for the solution of the problem. The algorithm works in phases, and in each phase it uses a fixed value of the parameter. After the phase it approximates the optimal value of the parameter by the known part of the input, and it uses this new value of the parameter for the next phase. We compare this new algorithm to the algorithm which uses the optimal worst case parameter setting (defined in [2]) on several test cases. We use randomly generated tests and also real data to analyse the algorithms. We also present the extension of the algorithm for the two sets scheduling model of [3] which can be considered as a generalization of the multiprocessor scheduling problem with rejection.

Acknowledgements

This research has been supported by the Hungarian National Foundation for Scientific Research, Grant F048587.

References

[1] S. Albers, B. Schröder: An Experimental Study of Online Scheduling Algorithms. *ACM Journal of Experimental Algorithms* **7**, (2002), article 3

[2] Y. Bartal, S. Leonardi, A. Marchetti-Spaccamela, J. Sgall, L. Stougie, Multiprocessor scheduling with rejection, *SIAM Journal on Discrete Mathematics* **13** (2000) 64–78.

[3] Cs. Imreh, Scheduling problems on two sets of identical machines, *Computing*, **70**, (2003), 277–294.

Matlab Version of the GLOBAL Optimization Method

László Pál

Global optimization aims to find a solution point satisfying all given constraints and for which the objective function reaches its smallest value, the global minimum.

This talk presents the stochastic global optimization algorithm GLOBAL [3], as a derivative free version of the method of Boender et al. [2]. We introduce a new Matlab based implementation of the original Fortran and C code together with some algorithmic improvements. The latter include a new, BFGS local search procedure, a new Matlab implementation of the UNIRANDI local search method [6], an extended capability for larger dimensional problems, and further changes to improve the efficiency of the procedure. Special attention has been devoted to increasing the reliability, while keeping the low computational complexity of the original implementation.

We illustrate the efficiency of the method on a set of standard test problems as well as on hard global optimization problems [1, 7, 8] and compare the numerical results with the old version and with a new direct search procedure, C-GRASP [5]. The results [4] are encouraging, and the new Matlab version of the algorithm GLOBAL should be available soon for academic and nonprofit purposes at (`www.inf.u-szeged.hu/~csendes/reg/regform.php`).

References

[1] Banga, J.R., C.G. Moles, and A.A. Alonso: Global Optimization of Bioprocesses using Stochastic and Hybrid Methods. In: *C.A. Floudas and P. Pardalos (eds.): Frontiers in Global Optimization*, Springer, Berlin, 2003, 45–70

[2] Boender, C.G.E., A.H.G. Rinnooy Kan, G.T. Timmer, and L. Stougie: A stochastic method for global optimization, *Mathematical Programming* 22(1982) 125–140

[3] Csendes, T.: Nonlinear parameter estimation by global optimization – efficiency and reliability. *Acta Cybernetica* 8(1988) 361-370

[4] Csendes, T., Pál, L., Sendin, J.O.H., Banga, J.R.: The GLOBAL Optimization Method Revisited. Accepted for publication In the *Optimization Letters*

[5] Hirsch, M.J., C.N. Meneses, P.M. Pardalos, and M.G.C. Resende, Global optimization by continuous GRASP, *Optimization Letters*, vol. 1, no. 2, 201-212, 2007

[6] Järvi, T.: A random search optimizer with an application to a max-min problem, *Publications of the Institute for Applied Mathematics, University of Turku*, No. 3, 1973.

[7] Moles C.G., J.R. Banga, and K. Keller: Solving nonconvex climate control problems: pitfalls and algorithm performances. *Applied Soft Computing* 5(2004) 35–44

[8] Moles C.G., G. Gutierrez, A.A. Alonso, and J.R. Banga: Integrated process design and control via global optimization – A wastewater treatment plant case study. *Chemical Engineering Research & Design* 81(2003) 507-517

A Global Optimization Algorithm for INTLAB

László Pál

This talk considers an algorithm for the bound constrained global optimization problem implemented in Matlab. It is based on the methods investigated in the past ([1] and [4]), which were first developed from the global optimization algorithm of the Numerical Toolbox [3]. The algorithm uses interval arithmetic operations and automatic differentiation offered by the INTLAB [5] package. INTLAB is a Matlab toolbox supporting real and complex interval scalars, vectors and matrices. It is designed to be fast, and every computation is rigourously verified to be correct.

Interval analysis is a powerful tool which allows to implement branch-and-bound algorithms able to solve global optimization problems. Our algorithm is also based on this technique using the most common accelerating devices: the cutoff test, the concavity test, the monotonicity test, and the interval Newton step. Beyond natural interval extension (based on naive interval arithmetic), a simple centered form inclusion function is also applied. Once the inclusion of the gradient is available, the intersection of these inclusion functions proved to be a good quality estimation of the range of the objective function. We use also multisection and advanced subdivision direction selection.

We have completed a numerical test, and compared the efficiency and the results of the INTLAB implementation to that of a C-XSC procedure. For the comparison we used the standard global optimization problems. The main indicators were the number of function, gradient and Hessian evaluations, the number of iterations, and the memory complexity.

Summarizing our numerical results, we can state the new implementation is as efficient as the old one, with the exception of the CPU time. The CPU time needed is as a rule by at least two order of magnitude higher for the INTLAB version (as it can be anticipated regarding the interpreter nature of Matlab). In spite of the lower speed, the new method is easy to use, and suitable for finding verified solutions of bound constrained global optimization problems.

References

[1] Csendes, T.; *New subinterval selection criteria for interval global optimization*. J. *Global Optimization* 19(2001) 307-327

[2] Csendes, T., L. Pál, J.O.H. Sendín, and J.R. Banga; The GLOBAL Optimization Method Revisited. Accepted for publication in the *Optimization Letters*.

[3] Hammer, R., M. Hocks, U. Kulisch, and D. Ratz; *Numerical Toolbox for Verified Computing I.* Springer-Verlag, Berlin, 1993.

[4] Markót, M.C., J. Fernandez L.G. Casado, and T. Csendes; New interval methods for constrained global optimization. *Mathematical Programming* 106(2006) 287-318

[5] Rump, S.M.; Intlab – Interval Laboratory. In: T. Csendes (ed.): Developments in Reliable Computing, *Kluwer Academic Publishers*, Dordrecht, 1999, pp. 77-104

Research of Swarm Intelligence Simulation with the Help of NXT Robots

Attila Pásztor

Nowadays a „relatively" new scientific research field of Artificial Intelligence is the SWARM INTELLIGENCE. Several researchers study the attitudes and habits of the animals that live in colony in the wild and researchers try to adopt their notes into robot groups. In order to reach a kind of goal the autonomous robots change into a multi agent system, they communicate and cooperate with each other like the animals.

This article is about how we got from the simple programmable mobile robots communicating tasks to the simulation of robots which imitating some animals' food collecting habits at the Kecskemét College faculty of GAMF at the department of Information Technology. In this experiment we used the food collecting habits of the African Desert Ants'. With the help of simple NXT robots in a simulated area the robots as a group collect objects (foods). They communicate with each other via blue tooth and use some sensors (touch, light and ultrasonic) to do the task. The first robot examines the territory and memorizes coordinates of the objects then returns back to the „anthill" and gives them the coordinates. After that the robots then convert the Descartes coordinates into Polar ones and start to the right directions. The robots fulfil the tasks together rapidly and more effectively.

This simulation is the first step in the series of possible projects. Improving new sensors and perfect communicating channels more and more robot colonies can be step up. These researches can be starting points of solving many problems or tasks, for example rescue humans and objects from polluted areas or solve collective cartographic issues.

References

[1] Blum, C., Sampels, M.,: An Ant Colony Optimalization Algorithm for Shop Shedulings Problems. J. Math. Model. *Algorithms* 3 (2004) 285-308

[2] Bonebeau, E., Dorigo, M., Thereulaz, G.,: Swarm Intelligence: From Natural to Artificial Systems. *Oxford University Press*, New York (1999)

[3] Shervin Nouyan, Dorigo, M., : Chain Based Path Formation in Swarms of Robots 5th International Workshop, ANTS 2006, Procedings 120-132

[4] http://bricxcc.sourceforge.net/nbc/nxcdoc/NXC_Guide.pdf

[5] Miller,P.,: Clever swarm, National Geographic (2007. 06)

[6] http://index.hu/cyberia/swarm, (2008.02.28)

A Comparative Study of C++ Standard Template Library's Formal Specification

Norbert Pataki and Gergely Dévai

The C++ Standard Template Library (STL) is an essential library that is used in C++ programs. The STL is built up according to the *generic programming* paradigm. It provides many useful containers (for example `vector` or `set`, etc.), iterators, algorithms (for instance `for_each()` or `sort()`) and functors. These work together and they are designed to be highly extensible.

The generic programming results in a special layout. Algorithms are fairly irrespective of the used container, because they work with iterators. For instance, we can use the `for_each()` algorithm with all containers. As a result of this layout the complexity of the library is greatly reduced and we can extend the library with new containers and algorithms simultaneously.

However, the STL itself is defined in the C++ Standard in an informal way. It may lead to misunderstood in some special cases and it does not help when one needs to formally prove the correctness of a safety-critical application. Unfortunately, ambigous specification can be found in the Standard too.

Formal techniques are needed to overcome the previous problems. Many formal specifications have been developed for the C++ STL and the generic programming paradigm. Various techniques have been worked out from the extension of well-known Hoare's method to the debugger-engined proof execution. Software tools also have been implemented to write safer STL-based code fragments.

In this paper we give an overview about formal techniques and tools that are used in C++ STL's specification. We examine the advantages and disadvantages of the known methods.

References

[1] Dévai, D., Pataki, N.: A Tool for Formally Specifying the C++ Standard Template Library, 2007, Annales (to appear)

[2] Meyers, S.: *Effective STL*, Addison-Wesley, 2001.

[3] Pataki, N., Porkoláb, Z., Istenes Z.: Towards soundness examination of the C++ standard template library, In *Electronic Computers and Informatics*, pages 186–191, 2006

[4] B. Stroustrup: *The C++ Programming Language*, Addison-Wesley, special edition, 2000.

Principal Varieties of Finite Congruences

Ville Piirainen

The classification of formal languages is one of the central problems in automata theory. For string languages the basic theory is quite well-established, but due to an exponential increase in complexity and the diverse uses of trees in many applications, there are still many questions in the classification theory for tree languages which are open for debate and discovery. Our view of trees and tree languages is strictly algebraic, and we are mainly interested in classes of tree languages of combinatorial nature; for example the varieties of the definite, the reverse definite, or the piecewise testable tree languages.

The general pattern of constructing a variety of tree languages consists of defining a chain or otherwise ordered set of suitable congruences of term algebras whose congruence classes then generate the languages. This set of congruences can be regarded as the generating set of the class, but proving the variety properties of this class using only the basic definitions can be relatively tedious compared to the fact that the result usually is completely obvious. However, each congruence by itself generates a principal variety of finite congruences, and we have developped a few simple criteria which guarantee that if the congruences satisfy them, the union of these principal varieties, which is the same set that the congruences generate together, is indeed a variety. Hence, we can omit many of the technicalities we would face using the original definitions of varieties.

Given a class of tree languages, in addition to the corresponding class of congruences of term algebras, we are also interested in the class of algebras or tree recognizers which recognize the class. The class of such algebras corresponding to a principal variety of finite congruences has an interesting special property: it is the finite part of a locally finite variety of algebras. This correspondence is actually bijective, and might provide interesting new results on tree language varieties.

Enhanced Phasespace Sampling of Biomolecular Systems Using Metastability

Jan-Hendrik Prinz

One of the major problems in the simulation of molecular systems such as protein is the limitation of the simulation time. Since high frequency modes restrict the integration timestep to about 1 fs, it is not possible to reach the biological relevant timescale of several ms. This issue is often referred to as the sampling problem. Common approaches to alleviate this difficulty is to use immense amounts of computational resources or step back to coarse-grained or other simplified models.

A recent idea is to not simplify the model, but constrain the interest to some predefined system properties, which may be of thermodynamic (e.g. equilibrium constants), kinetic (e.g. transition rates), or structural type (e.g. the transition state ensemble). This allows to split the integration process into a controlled series of small simulations, which are selected in such a way that a maximum of statistical information is gathered with respect to these predefined system properties.

Our method uses an adaptive technique to construct a suitable metastable and markovian model, which reduces the complexity to a graph representation. This picture allows for a convenient and fast way to compute errors of system properties and predict initial conditions for additional simulations to collect the maximum of statistical information.

We applied the adaptive method to a small five residue peptide (MR121-GSGSW) and obtained several results indicating that the method can speed up the rate of convergence for the chosen target property, e.g. the estimated energy difference between metastable conformations, the mean first passage time or the overall number of existing metastable states in the system.

Acknowledgments

This work was supported by the German Research Foundation (DFG), the International Graduiertenkolleg IGK 710 "Complex processes: Modeling, Simulation and Optimization".

References

[1] P. Deuflhard and M. Weber. Robust perron cluster analysis in conformation dynamics. *ZIB Report*, 03-09, 2003.

[2] Frank Noe, Illia Horenko, Christof Schutte, and Jeremy C. Smith. Hierarchical analysis of conformational dynamics in biomolecules: Transition networks of metastable states. *The Journal of Chemical Physics*, 126(15):155102, 2007.

[3] Nina Singhal and Vijay S. Pande. Error analysis and efficient sampling in markovian state models for molecular dynamics. *J. Chem. Phys.*, 123(20):204909, 2005.

Determining Initial Bound by "Ray-method" in Branch and Bound Procedure

Anett Rácz

It is very hard to solve an optimization problem if the problem contains integrality constraints. Such problems are integer programming problems, which can be solved by various algorithms. The most widespread algorithm is the Branch and Bound algorithm, which similarly to other methods, builds a search tree. Following from the structure of the problem, sometimes these methods obtain a result only at the expense of building a large tree. It may also happen that it take a lot of time to recognise that the problem does not have an integer solution. Therefor, the demand to reduce the size of the tree built by the algorithms is understandable. The paper aims to develop a reliable method, that can reduce the size of the search tree. The idea is based on the results of the theoretical research of the Computing Centre of the Russian Academy of Sciences.[1] It is difficult to adapt the main theoretical results into a computer environment. As a result these methods in original form are not applicable yet. Based on the theorems, that have been worked out recently, a new method, which can be relatively easily implemented has been developed, which has proved to be efficient on the basis of tests. By using the solution of the relaxation problem, the algorithm gives a bound for the value of the objective function of the problem with integer conditions, thereby creating an initial bound for the solvers, wich build a search tree. The key to the efficiency of the algoritms based on the ray-method is to create a feasible integer solution relatively quickly. One characteristic of the tree is that the optima get worse as we move down on its branches. If we find a feasible integer solution on one of the nodes to the original problem, we use the objective value of this solution as a bound. If a node has to be branched, but its objective value is already worse then our current bound we do not have to perform this operation, and we say this node is fathomed. However, we do not have such a bound at the starting node. The algorithms based on the ray-method aim to create a initial bound before we start to build the search tree. As a result, we get the opportunity to cut the unnecessary branches as soon as possible. This means that we do not have to build a big part of the tree in practice. It is essential to be able to implement these methods, since they serve to accelerate an applied method, namely Branch and Bound. It is obvious that, if calculating the initial bound takes more time then building the part of the tree, which can be cut with this bound, then we have not improved the efficiency. The method presented in this paper is relatively simple, easy to implement, which has proved to be efficient during the tests so far.

Acknowledgements

My special thanks are due to my supervisor Dr. Erik Bajalinov for his invaluable comments. I thank the Department of Applied Mathematics and Probability Theory for the technical background and other useful support.

References

[1] Khachaturov, V.R., Combinatorial methods and algorithms for solving problems of discrete optimization with large dimensionality, Nauka, 2000.

Enhancing Facial Features Detection with Deformable Face Templates

Levente Sajó

In research of multi-modal human-computer interaction, an important problem is to extract information from human faces. For this, in every case, the first step is to localize faces, then the localized face can be used in further processing steps, like gender, age, gesture recognition, facial feature extraction.

Because color images do not carry much more information, detection steps are made on grayscale images. Faces and facial features are complex shapes having high variability in scale and texture therefore their localization is not a simple task. Many different object detection techniques have been published in the last few years, the most successful detectors are template based and appearance based detectors [1]. Viola and Jones have developed a template based object detector (called Boosted Cascade Detector - BCD) which has a similar detection rate then the other detectors (based on SVM, Neural Networks) but needs much less computational time [2].

Our aim was to detect faces and facial features in real time, on video streams, so BCD was a plausible choice. BCD was used for localizing faces and on the localized face individual feature detectors were used to localize facial features (e.g. mouth and eyes corners). Face Detection showed us a convincing performance, but since facial features contains less information then the whole face, individual feature detectors seemed to be unreliable, therefore some further processing needed.

For enhancing the reliability of feature detectors two additional steps have been implemented. First, the search area of every individual feature detector has been restricted. In the second step, the detected feature points are checked whether fulfill a predefined shape constraint. For this, a face template has been defined, which contains template points corresponding to the facial features (we used 16 points, 4x3 point from eyebrows and eyes and 4 points from mouth). The relation between template-points is described with a one or more template rules. A rule defines the relative position of two or more template points. E.g.: one rule can define the symmetry of the face; the center of mouth points should lay on the perpendicular bisector of two eyes.

Every rule has a kernel function and three parameters: estimated value, tolerance, increment. We currently use two types of kernel functions, every type having two versions: one operates on x coordinate, the other on y coordinate of a point. One kernel function calculates the distance between two points (1); the other calculates the distance between mean of two points and a third point (2).

$$\begin{aligned} xDistance(P1, P2) &= P2.x - P1.x, \\ yDistance(P1, P2) &= P2.y - P1.y \end{aligned} \tag{1}$$

$$\begin{aligned} xMiddle(P1, P2, P3) &= (P1.x + P2.x)/2 - P3.x, \\ yMiddle(P1, P2, P3) &= (P1.y + P2.y)/2 - P3.y \end{aligned} \tag{2}$$

The penalty value is calculated as a combination of the return value of the kernel function and the other three rule parameters. If the kernel-function value is in the neighbor of the estimated value, the return value is 0, otherwise penalty value is calculated using increment value.

if (funcvalue is between [estimated value - tolerance, estimatedValue + tolerance])
then penaltyValue = 0;
else penaltyValue = abs(funcvalue - estimatedValue) - tolerance) * increment

The sum of penalty values gives the overall penalty value of a set of feature point. Every combination of the first few feature points found by individual detectors are matched to the face template and that set of feature points is selected which has the smallest penalty value. On the selected set of points, using the template, those points which has still large penalty value, it is possible to calculate a new (better) position, which together with other points forms a better face template.

In this stage of the project, the parameters of the rules are defined manually. Later a learning algorithm can be implemented, which uses a face database with manually defined face points and automatically defines the rules and their parameter.

References

[1] M. H. Yang, D. J. Kriegman, N. Ahuja. Detecting Faces in Images: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 24, January 2002.

[2] P. Viola, M. Jones. Robust Real-time Object Detection, Technical Report, *Cambridge Research Laboratory*, 2001.

[3] D. Cristinacce. Automatic Detection of Facial Features in Grey Scale Images, PhD Thesis, *University of Manchester*, 2004.

Evaluating the Effectiveness of Object Oriented Metrics for Bug Prediction

István Siket

Although a lot of object-oriented metrics were defined and published in the last decades their usage is still not widespread. The main reason of this is that the relationship between object-oriented metrics and software qualities (e.g. maintainability or reusability) were rarely confirmed with experimental results [1].

In our experiment we examined the relationship between object-oriented metrics and fault proneness of classes. We analyzed a large open source software called Mozilla, calculated 58 object-oriented metrics for the classes of Mozilla [2], collected the reported and corrected bugs from the bug tracking system of Mozilla and associated them with its classes. We applied logistic regression to examine which metrics could be used to predict the faultiness of the classes. We found that 17 of the 58 object-oriented metrics were useful predictors but in different rate. CBO (Coupling Between Object classes) metric was the best but it was only a slightly better than NOI (Number of Outgoing Invocations) and RFC (Response Set for a Class) which were proven useful as well.

We also examined the metrics by their categories and we obtained that coupling metrics were the best predictors for finding bugs but the complexity and size metrics also gave good results. On the other hand, all inheritance metrics were statistically insignificant.

We tried to achieve better result by using more predictors (multivariate logistic regression) but the result was almost the same as the result of CBO, which was the best metric.

Keywords: compiler wrapping, object-oriented metrics, logistic regression, fault prediction

References

[1] Gyimóthy, Tibor and Ferenc, Rudolf and Siket, István. Empirical Validation of Object-Oriented Metrics on Open Source Software for Fault Prediction. In *IEEE Transactions on Software Engineering*, volume 31, pages 897-910, IEEE Computer Society, October 2005.

[2] Ferenc, Rudolf and Siket, István and Gyimóthy, Tibor. Extracting Facts from Open Source Software. In *Proceedings of the 20th International Conference on Software Maintenance (ICSM 2004), pages 60-69*. IEEE Computer Society, September 2004.

Verification of UML 2 State Machines by Automated Model Transformation to the SAL Model Checker

Áron Sisak

Nowadays, Model Driven Software Development aims at offering a framework to automate the synthesis and verification of software. This paper aims at presenting the transformation of UML 2.0 state machine diagrams to a model checking language (i.e., a mathematical model), which is suitable for formal verification. The approach allows to perform early investigation of software models defined in a very widespread modeling language with rich capabilities.

UML 2.0 State Machines provides the de facto standard for modeling reactive, state-based system behavior. It offers both a graphical representation for intuitive, visual system modeling and a semantics that aims to serve as a base for both code generation and verification and validation activities. However, besides the powerful concepts included in the State Machine language, the semantics contained by the standard contains several ambiguities as well. The Precise Statechart Semantics introduced in [1] offers remedy for these problems. The semantics is defined in a *declarative* manner, which suits the declarative model checking paradigm very well. The transformation is based upon Precise Statecharts (PSC) .

Formal verification aims to prove that a system model satisfies certain properties. Model checking is a widespread formal verification approach. It performs the exhaustive exploration of the model to verify the desired properties. The SAL model checking framework [2] is widely used to perform model checking, mostly because its capability of coping with large model state spaces. The PSC semantics is based on the Kripke Transition System formalism, which is semantically very close to the foundations of the SAL module language, which makes possible to find a well-established mapping to transform PSC constructs into the SAL language.

My prototype transformation tool is implemented on the top of the Java API provided for the Precise Statechart Semantics, which makes possible to process UML2 statechart models produced by EMF-based UML modeling tools, with access to the special constructs defined by the PSC semantics (sets and relations). The tool implements a template-based solution, using the Velocity template engine. Metamodel-level constructs are transformed directly without any knowledge about the actual model being processed; e.g., initializing a statechart or firing a transition are mapped into a generic SAL module. States, regions, triggers and other elements are mapped to enumerations based on the actual model being transformed. Guards and effects are implemented as parameterized generic SAL code. Arbitrary SAL code can be used in guard and effect implementations. The actual run-time data consist of a set of boolean arrays representing the active states in the statechart structure, and a set of SAL variables used in the guards and effects. One of the most difficult aspect of implementing the transformation engine is to decide on proper SAL constructs that both capture the semantics properly and avoid state space explosion as much as possible, e.g., using simple boolean arrays instead of records can reduce the state space with multiple orders of magnitude.

Besides support for basic state machine constructions discussed above, verification of interacting state machines is also possible, as event queue models are supported. Basic support to map standard UML variable types to SAL types is also present in the current implementation. After implementing the full semantics, various abstraction possibilities are to be investigated in order to cope with even bigger state spaces.

References

[1] G. Pintér. Model Based Program Synthesis and Runtime Error Detection for Dependable Embedded Systems, *PhD Thesis*, BME MIT, 2007

[2] L. de Moura et al. SAL 2, in *R. Alur and D. Peled, editors, CAV 2004*, LNCS 3114, pp. 496–500

Steiner Loops

Izabella Stuhl

If the order of any product of two different translations of a finite Steiner quasigroup of size $n > 3$ is odd, then the group G generated by the translations of the corresponding Steiner loop S of order $n + 1$ contains the alternating group of degree $n + 1$. For extensions L of the group of order 2 by the Steiner loop S the group generated by all translations of L is an extension of an elementary abelian 2-group Θ by G. The group Θ has order 2^n, respectively 2^{n-1} depending whether n is odd or even.

Moreover, we thoroughly study the relations between such extensions L and oriented Steiner triple systems in order to obtain more detailed knowledge about these loops L, about the structure of their automorphism groups and the isomorphism classes.

Designing a Tracking Controller for Passenger Cars with Steering Input

Emese Szádeczky-Kardoss

The results presented in the paper were motivated by a practical application. The goal was to build a parking assist system for a passenger car with a human driver. Such a system is able to map the environment of the vehicle and to detect the existence of an accessible parking place on the map where the vehicle can park into. Once the parking place is identified, a feasible path geometry has to be designed and the tracking of this reference is realized by controlling the steering wheel.

This system was realized on a Ford Focus type vehicle equipped with an Electronic Power Assist Steering (EPAS) but without automatic gear. In practice this means that our tracking controller can only influence one of the vehicle inputs. (Generally, the car has two control inputs: the longitudinal velocity v and the angle of the steering wheels φ.) In our case the steering wheel angle φ can be influenced through the EPAS, but the velocity must be generated by the driver who handles the pedals (throttle, brake and clutch). This implies that we face a novel theoretical control problem: the tracking controller should be designed such that the controller can only generate the angle of the steered wheels φ, and the longitudinal velocity of the vehicle is an external signal which cannot be influenced by the controller but can be measured. (This velocity is denoted by v_{car} to indicate that it is not a controllable input.)

The kinematic model of the car is used to carry out the calculations. We supposed that the movement of the car can be well estimated at sufficiently low velocities by the motion equations of a bicycle which is fitted on the longitudinal symmetry axis of the vehicle. There exists several methods in the literature which control such a car model if both the longitudinal velocity v and the angle of the steering wheels φ are the outputs of the controller (e.g. [1]), but the one-input case has not been addressed until now.

Our solution for the single-input control problem is based on a new time-scaling scheme. This means that the path planning method generates a reference path paramterized by a virtual time τ then a time-scaling function is used which maps this virtual time τ into the real time t. Roughly speaking, this time-scaling function depends on the measured car velocity v_{car} and on the tracking error along the path and its result is a reference path according to t which gives the reference signals for the tracking controller.

To obtain the time-scaling function we transform the one-input model of the car into a two-input model using again time-scaling such that a new (scaling) input u_s is created for the system evolving according to the time τ. Using this concept, the tracking controller has again two outputs. In other words, if we are not able to control the velocity of the car, we influence the time distribution of the reference path instead.

Our tracking control method with time-scaling is based on the flatness property of the two-input kinematic car model [2]. We implemented the method using Matlab and Simulink. Several simulations were performed to verify the functioning of the closed loop system. Our method is able to eliminate initial position and orientation errors by decelerating the reference until the convergence of the real path is achieved.

We also tested our tracking controller in the real Ford Focus type car. We used the fast prototyping environment of dSPACE (AutoBox and ControlDesk) for the implementation and monitoring. The real test results are similar to the simulations. Our method is able to ensure accurate tracking of the reference path such that the longitudinal velocity is generated by the driver and the controller influences only the angle of the steered wheels and the time distribution along the reference path.

References

[1] F. Cuesta, A. Ollero. Intelligent Mobile Robot Navigation, *Springer Tracts in Advanced Robotics*, vol. 116, Springer, Heidelberg, 2005.

[2] M. Fliess, J. Lévine, P. Martin, P. Rouchon. Flatness and Defect of Nonlinear Systems: Introductory Theory and Examples, *International Journal of Control*, vol. 61, no. 6, pp. 1327–1361, 1995.

Necessary test cases for Decision Coverage and Modified Condition / Decision Coverage

Zalán Szűgyi and Zoltán Porkoláb

Coverage refers to the extent to which a given verification activity has satisfied its objectives. Coverage measures can be applied to any verification activity, although they are most frequently applied to testing activities. Appropriate coverage measures give the people doing, managing, and auditing verification activities a sense of the adequacy of the verification accomplished. [1]

The code coverage analysis contains three main steps [2], such as: finding areas of a program not exercised by a set of test cases, creating additional test cases to increase coverage and determining a quantitative measure of code coverage, which is an indirect measure of quality. Optionally it contains a fourth step: identifying redundant test cases that do not increase coverage.

Different types of code coverage analysis requires different set of test cases. In this paper we concern to Decision Coverage (DC), and Modified Condition / Decision Coverage (MCDC) testing methods. The DC only requires that every lines of a subprogram must be executed and every decisions must be evaluated to true and to false. The MCDC is more strict. It contains the requirements of DC and it demands to show that every condition in a decision independently affects the outcome. It is clear there are more test cases are needed to satisfy the requirements of MCDC than DC. But it is not so trivial how much can be spared when testing by DC instead of MCDC.

We analyzed several projects in subprogram level, and estimated how many test cases are needed to satisfy the 100% DC and MCDC coverage. These projects was written in Ada programming language, and some of them were open source and some of them were developed in the industry. We analyzed them in several aspects: McCabe metrics, nesting, maximal argument number in decision etc, and examined how these aspects affected the difference of necessary test cases. At last we answered to the question: how many test cases need more to satisfy MCDC then DC.

References

[1] Kelly J. Hayhurst, Dan S. Veerhusen, John J. Chilenski, Leanna K. Rierson: A Practical Tutorial on Modified Condition/Decision Coverage

[2] `http://www.bullseye.com/coverage.html` 2008.03.01

[3] Tibor Gyimóthy, Rudolf Ferenc and István Siket: Empirical Validation of Object-Oriented Metrics on Open Source Software for Fault Prediction. *In IEEE Transactions on Software Engineering*

[4] `http://www.antlr.org/` 2008.03.02

[5] Oliver Kellogg: `http://www.antlr.org/grammar/ada` 2008.03.02

[6] `https://lemon.cs.elte.hu/site/` 2008.03.09

[7] `http://www.gnu.org/software/glpk/` 2008.03.09

Verified Mobile Code Repository in the Intelligent Space

Máté Tejfel, Zoltán Istenes and László A. Jeni

The Intelligent Space (iSpace) [6] is a space (room or corridor) containing several communicating Distributed Intelligent Network Devices (DINDs) which share their information about a human environment. A DIND consists of three basic elements, the sensor to monitor the dynamic environment of the DIND (which contains people, vehicles and robots, etc.), the processor (computer) to process the captured information and the communication device to provide the cooperation of different DINDs through a network. The iSpace also contains mobile robots which handle real objects in the human environment to support the people in it. The mobile robots can be controlled with mobile code technology, by dynamically downloading on the robots the tasks needed to be executed by them. In some earlier work the concept how the safe mobile code technology can be applied in the case of mobile robots and how it integrated into the iSpace environment was introduced [2, 3, 4, 5]. In this model the mobile code is created, verified, stored and transmitted to the robot using the CPPCC architecture [1]. Robots contain explicit and formally expressed security requirements (for example that each task has to be started from, and stop in a given state or the robot may not go to dangerous places, lift, stairway etc.). A formal verification system can verify the mobile code properties correspondence against the robots requirements. The robot refuses to execute those mobile code tasks violating its requirements. For the execution of the correspondence analysis formally verified properties of the mobile code have to be attached to the mobile code. This paper describes a method for formalisation and verification of properties of mobile code and provides a repository of mobile codes with attached verified properties.

Acknowledgements

This work was supported by Ferenc Deák scholarship.

References

[1] Horváth, Z., Kozsik, T. Safe mobile code - CPPCC: Certified Proved- Property-Carrying Code. *G. Czajkowski and J. Vitek, Resource Management for Safe Languages (in: ECOOP 2002 Workshop Reader, LNCS 2548/2002, Springer-Verlag)*, 2002, pp. 8-10.

[2] Istenes, Z., Kozsik, T., Hoch, Cs., Tóth, L. A. Proving the correctness of mobile Java code, *6th Joint Conf. on Math. and Comp. Sci.*, July 12-15., 2006, Pécs, Hungary, accepted to *Pure Mathematics and Applications*, SAAS Ltd.-SAAS Publishing, Budapest, Hungary, Vol. 17 (2006), No. 3-4,.

[3] Istenes, Z., Kozsik, T. Commanding a robot in a safe way *Proceedings of the 10th Symposium on Programming Languages and Software Tools (SPLST 2007)*, Budapest, 2007, pp. 167-177., ISBN 978 963 463 925 1

[4] Jeni, L. A., Istenes Z., Korondi, P., Hashimoto H. Hierarchical Reinforcement Learning for Mobile Robot Navigation using the iSpace Concept, *11th IEEE International Conference on Intelligent Engineering Systems*, 2007.

[5] Jeni, L. A., Istenes Z., Tejfel M. Safe Mobile Code in the Intelligent Space, Submitted to *Proceedings of the 2nd Symposium of Intelligent Systems (IRFIX 2007)*, Budapest, 2007

[6] Lee, J.-H., Hashimoto, H. Inteligent Space - concept ant contents, *Journal of Advanced Robotics*, Brill Academic Publishers, Vol. 16, Number 3, May, 2002, pp. 265-280.

Analysis of Size Differences Between Static and Dynamic Slices

Gabriella Tóth

Program slicing provides great support in executing software engineering tasks, eg. debugging errors, maintenance, testing, reengineering, thus the purpose of my paper is a more sophisticated analysis of the two kind of slicing: dynamic and static slicing.

The backward program slice is a subset of the program which contains the statements that have direct or indirect effect on a certain variable occurrence of a certain program point(criteria). The forward program slice is a subset of the program which contains the statements that have direct or indirect dependence on a certain program point.

In the case of static slicing, the dependencies occurring during all the possible program executions appear. Dynamic slicing, which is exact for a given testcase, results in realized dependencies belonging to a concrete execution. Compared to the static slice, the number of dependencies are lower. Union slicing, which is the union of dynamic slices for the same criteria in different testcases, provides a solution for this problem.

On the basis of the data calculated with the help of the Jadys union dynamic slicer developed by the Software Engineering Department of Szeged and the Indus static slicer developed by a research group of Kansas State University the received statement level slices are compared. In a previous article[1] only the sizes of the slices was measured, but in this paper I examine the causes of differences between the sizes of the slices through concrete examples.

The examination brings up two major causes as an explanation for these differences: firstly, there are statements which could appear in the dynamic slice by adding a new testcase, secondly, there are the statements which are included in the static slice because of the conservative approach of static slicing. The goal is to separate these causes as much as possible and with the help of this to give recommendations for the further improvement of slicing methods.

References

[1] Attila Szegedi, Tamás Gergely, Árpád Beszédes, Tibor Gyimóthy, Gabriella Tóth. Verifying the Concept of Union Slices on Java Programs,CSMR 2007

List of Participants

Bonato, Thorsten: University of Heidelberg, Department of Computer Science, INF 368, D-69120 Heidelberg, Germany, E-mail: thorsten.bonato@informatik.uni-heidelberg.de

Bozóki, Ferenc: Software Engineering Group, Ericsson Hungary Ltd., H-1300 Budapest P.O. Box 107, Hungary, Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: bozoki@tmit.bme.hu

Balázs, Péter: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: pbalazs@inf.u-szeged.hu

Csernenszky, András: OTP Bank, Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: csernenszkya@gmail.com

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

Csorba, Kristóf: Budapest University of Technology and Economics, H-1111 Budapest, Goldmann György tér 1., Hungary, E-mail: kristof@aut.bme.hu

Dévai, Gergely: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: deva@elte.hu

Dombi, József: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: dombi@inf.u-szeged.hu

Dombi, József Dániel: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: dombijd@inf.u-szeged.hu

Erdőhelyi, Balázs: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: ber@inf.u-szeged.hu

Erős, Levente: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: eros@tmit.bme.hu

Ferenc, Rudolf: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: ferenc@inf.u-szeged.hu

Fischer, Ferenc: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: fischerf@inf.u-szeged.hu

Fuchs, Martin: University of Vienna, Faculty of Mathematics, A-1010 Vienna, Dr.-Karl-Lueger-Ring 1, Austria, E-mail: martin.fuchs@univie.ac.at

Fülöp, Lajos Jenő: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: flajos@inf.u-szeged.hu

Főző, Csaba: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: fozo@tmit.bme.hu

Gajdos, Sándor: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: gajdos@db.bme.hu

Gara, Mihály: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: h370371@stud.u-szeged.hu

Gáspár-Papanek, Csaba: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: gaspar@tmit.bme.hu

Gergely, Tamás: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: gertom@inf.u-szeged.hu

Hegedűs, Péter: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail:

Horváth, Zoltán: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: hz@inf.elte.hu

Hunyadi, Levente: Budapest University of Technology and Economics, H-1111 Budapest, Goldmann György tér 1., Hungary, E-mail: hunyadi@aut.bme.hu

Illyés, László: Sapientia University, 510104 Piata Libertatii nr. 1, Miercurea Ciuc, jud Harghita, Romania, E-mail: laszlo6783@yahoo.com

Imreh, Csanád: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: cimreh@inf.u-szeged.hu

Istenes, Zoltán: Department of Software Technology and Methodology, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: istenes@inf.elte.hu

Jász, Judit: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: jasy@inf.u-szeged.hu

Jeni, László, Attila: Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail:

Kardkovács, Zsolt Tivadar: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: kardkovacs@tmit.bme.hu

Kardos, Péter: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: kardos.peter@dk.hu

Kaszó, Márk: Budapest University of Technology and Economics, H-1111 Budapest, Goldmann György tér 1., Hungary, E-mail: mkaszo@aut.bme.hu

Kertész, Attila: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: keratt@inf.u-szeged.hu

Kovács, György: Faculty of Informatics, University of Debrecen, H-4032 Debrecen, Egyetem tér 1., Hungary, E-mail: gykovacs@inf.unideb.hu

Legány, Csaba: Budapest University of Technology and Economics, H-1111 Budapest, Goldmann György tér 1., Hungary, E-mail: legcsabi@aut.bme.hu

Lövei, László: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: lovei@inf.elte.hu

Marton, József: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: maron@db.bme.hu

Nagy, Tamás: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: n_tamas@inf.elte.hu

Nagyné Víg, Anikó: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: viganiko@inf.elte.hu

Németh, Gábor: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: gnemeth@inf.u-szeged.hu

Németh, Tamás: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: tnemeth@inf.u-szeged.hu

Neumaier, Arnold: University of Vienna, Faculty of Mathematics, A-1010 Vienna, Dr.-Karl-Lueger-Ring 1, Austria E-mail: Arnold.Neumaier@univie.ac.at

Pataki, Norbert: Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: patakino@elte.hu

Pál, László: Sapientia University, 530170 Piata Libertatii, Nr. 1, Miercurea-Ciuc, Romania, E-mail: pallaszlo@sapientia.siculorum.ro

Pásztor, Attila: Kecskemét College Faculty of Mechanical Engineering and Automation, H-6000 Kecskemét, Izsáki u. 10., Hungary, E-mail: pasztor.attila@gamf.kefo.hu

Pécsi, Gábor: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: pecsi@tmit.bme.hu

Piirainen, Ville: Turku Centre for Computer Science/University of Turku, FI-20014 Turku, Finland, E-mail: ville.piirainen@utu.fi

Porkoláb, Zoltán: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: gsd@elte.hu

Popa, Raluca: Budapest University of Technology and Economics, H-1111 Budapest, Goldmann György tér 1., Hungary, E-mail: praluca@aut.bme.hu

Prinz, Jan-Hendrik: University of Heidelberg, Interdisciplinary Center for Scientific Computing, D-69120 Heidelberg, Neuenheimer Feld 368, Germany, E-mail: Jan-Hendrik.Prinz@iwr.uni-heidelberg.de

Rácz, Anett: Faculty of Informatics, University of Debrecen, H-4032 Debrecen, Egyetem tér 1., Hungary, E-mail: anett.racz@inf.unideb.hu

Sajó, Levente: Faculty of Informatics, University of Debrecen, H-4032 Debrecen, Egyetem tér 1., Hungary, E-mail: sajolevente@inf.unideb.hu

Sisak, Áron: Budapest University of Technology and Economics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: asisak@mit.bme.hu

Siket, István: Institute of Informatics, University of Szeged, H-6701 Szeged P.O. Box 652, Hungary, E-mail: siket@inf.u-szeged.hu

Sipőcz, Eszter: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: sipocz@tmit.bme.hu

Stuhl, Izabella: Institute of Mathematics and Informatics, University of Debrecen, H-4010, Debrecen, Hungary, E-mail: stuhl@math.klte.hu

Szabó, Gergely: Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: szabo@tmit.bme.hu

Szádeczky-Kardoss, Emese: Budapest University of Technology and Economics, H-1117 Budapest, Magyar Tudósok krt. 2., Hungary, E-mail: szadeczky@iit.bme.hu

Szűgyi, Zalán: Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: lupin@ludens.elte.hu

Tejfel, Máte: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: matej@inf.elte.hu

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

Tóth, Melinda: Department of Programming Languages and Compilers, Eötvös Loránd University, H-1117 Budapest, Hungary, E-mail: toth_m@inf.elte.hu

Vajk, István: Budapest University of Technology and Economics, H-1111 Budapest, Goldmann György tér 1., Hungary, E-mail: vajk@aut.bme.hu

Notes

Notes

Notes