Helsinki University of Technology Laboratory for Theoretical Computer Science
Annual Report 2001
Teknillisen korkeakoulun tietojenkäsittelyteorian laboratorion vuosiraportti 2001
Espoo 2002 HUT-TCS-Y2001

ANNUAL REPORT FOR THE YEAR 2001

Kimmo Varpaaniemi (Ed.)

TEKNILLINEN KORKEAKOULU
TEKNISKA HÖGSKOLAN
HELSINKI UNIVERSITY OF TECHNOLOGY
TECHNISCHE UNIVERSITÄT HELSINKI
UNIVERSITE DE TECHNOLOGIE D'HELSINKI

Helsinki University of Technology Laboratory for Theoretical Computer Science
Annual Report 2001
Teknillisen korkeakoulun tietojenkäsittelyteorian laboratorion vuosiraportti 2001
Espoo 2002 HUT-TCS-Y2001

ANNUAL REPORT FOR THE YEAR 2001

Kimmo Varpaaniemi (Ed.)

Helsinki University of Technology
Department of Computer Science and Engineering
Laboratory for Theoretical Computer Science

Teknillinen korkeakoulu
Tietotekniikan osasto
Tietojenkäsittelyteorian laboratorio

Distribution:
Helsinki University of Technology
Laboratory for Theoretical Computer Science
P.O.Box 5400
FIN-02015 HUT, Finland
Tel. +358-9-451 1
Fax. +358-9-451 3369
E-mail: lab@tcs.hut.fi

Printing: Picaset Oy,
Helsinki 2002

Abstract: This report describes the educational and research activities of the Laboratory for Theoretical Computer Science at Helsinki University of Technology during the year 2001. In the PDF version of this report, URL addresses are links to those addresses. For example, you get to the home page of the laboratory by clicking http://www.tcs.hut.fi/index.html.

CONTENTS

1 Personnel **1**

1.1 University staff . . . 1
1.2 Docents . . . 2
1.3 Main teachers of the active courses of the year 2001 . . . 2
1.4 Teaching assistants in the active courses of the year 2001 . 3
1.5 Researchers and research assistants . . . 5

2 Educational activities **6**

3 Research activities **9**

3.1 Formal methods in distributed systems . . . 9
3.2 Constraint programming based on default rules . . . 11
3.3 Computational methods in coding theory and discrete mathematics . . . 14
3.4 Generative string rewriting . . . 15
3.5 The `ANNA-MARIA` project . . . 15
3.6 Analysis and development of cryptosystems . . . 16
3.7 Security in mobile ad-hoc networks . . . 16

4 Conferences, visits and guests **17**

4.1 Conferences . . . 17
4.2 Visits . . . 21
4.3 Guests . . . 22

5 Publications **23**

5.1 Journal articles . . . 23
5.2 Articles in collections . . . 24
5.3 Conference papers . . . 24
5.4 Reports . . . 26
5.5 Licentiate's theses . . . 27
5.6 Master's theses . . . 28
5.7 Software . . . 29

1 PERSONNEL

1.1 University staff

Niemelä, Ilkka, D.Sc. (Tech.) (cf. 1.2, 1.3)	Professor in Computer Science. Head of the Laboratory.
Orponen, Pekka, D.Phil. (cf. 1.3)	Professor in Theoretical Computer Science since November 1. Professor (pro tem) during September 1 – October 31.
Husberg, Nisse, D.Sc. (Tech.) (cf. 1.2, 1.3)	Professor (pro tem).
Janhunen, Tomi, D.Sc. (Tech.) (cf. 1.3)	Professor (pro tem) until September 1. Teaching Researcher (on leave until September 1).
Kari, Hannu H., D.Sc. (Tech.)	Professor (pro tem) since October 1.
Lipmaa, Helger, PhD (cf. 1.3)	Professor (pro tem) since August 1.
Ojala, Leo, Lic.Sc. (Tech.) (cf. 1.3)	Professor Emeritus
Kangasniemi, Ulla	Secretary of the Laboratory.
Heljanko, Keijo, Lic.Sc. (Tech.) (cf. 1.3, 1.5)	Senior Assistant (on leave).
Lassila, Eero, Lic.Sc. (Tech.) (cf. 1.5)	Laboratory Manager (on leave).
Varpaaniemi, Kimmo, D.Sc. (Tech.) (cf. 1.4)	Senior Assistant.

1.2 Docents

Husberg, Nisse, D.Sc. (Tech.) (cf. 1.1, 1.3)	Docent in Verification.
Lilius, Johan, D.Sc. (Tech.)	Docent in Reactive Systems since February 1. Professor in Computer Science and Engineering (Åbo Akademi University).
Niemelä, Ilkka, D.Sc. (Tech.) (cf. 1.1, 1.3)	Docent in Logic and its Applications in Computer Science and Engineering.
Ukkonen, Esko, D.Phil.	Docent in Theoretical Computer Science. Professor in Computer Science (University of Helsinki and the Academy of Finland).
Östergård, Patric R.J., D.Sc. (Tech.) (cf. 1.5)	Docent in Coding Theory. Professor in Information Theory (HUT, Department of Electrical and Communications Engineering, on leave until August 1).

1.3 Main teachers of the active courses of the year 2001

Haanpää, Harri, Lic.Sc. (Tech.) (cf. 1.5)	Tik-79.161	Combinatorial Algorithms (spring)
Heinonen, Rauno, Lic.Sc. (Tech.)	Tik-79.148	Introduction to Theoretical Computer Science (spring)
Heljanko, Keijo, Lic.Sc. (Tech.) (cf. 1.5)	Tik-79.186	Reactive Systems (spring)
Herttua, Ilkka, Stud.Tech.	Tik-79.232	Safety-Critical Systems (spring)
Huima, Antti, M.Sc. (Tech.) (cf. 1.5)	Tik-79.159	Cryptography and Data Security (spring)
Husberg, Nisse, Professor (pro tem), D.Sc. (Tech.), Docent (cf. 1.1, 1.2)	Tik-79.149 T-79.185 Tik-79.193	Discrete Structures (spring) Verification (autumn) Formal Description Techniques for Concurrent Systems (spring)
Janhunen, Tomi, Professor (pro tem), D.Sc. (Tech.) (cf. 1.1)	T-79.144 T-79.154 Tik-79.230	Logic in Computer Science: Foundations (autumn) Logic in Computer Science: Special Topics II (autumn) Foundations of Agent-Based Computing (spring)

Kettunen, Esa, M.Sc. (Tech.)	Tik-79.179	Parallel and Distributed Digital Systems (spring)
Lipmaa, Helger, Professor (pro tem), PhD (cf. 1.1)	T-79.510 T-79.511	Seminar on Cryptography and Security Protocols (autumn) Special Course on Cryptology (autumn)
Niemelä, Ilkka, Professor, D.Sc. (Tech.), Docent (cf. 1.1, 1.2)	Tik-79.146 T(ik)-79.189 Tik-79.194 T-79.240 T(ik)-79.295 T(ik)-79.298	Logic in Computer Science: Special Topics I (spring) Student Project in Theoretical Computer Science Seminar on Theoretical Computer Science (spring) Special Course in Computational Complexity (autumn) Individual Studies Postgraduate Course in Digital Systems Science
Ojala, Leo, Professor Emeritus, Lic.Sc. (Tech.) (cf. 1.1)	Tik-79.157 T-79.192	Formal Description and Verification of Computing Systems (spring) Special Course in Theoretical Computer Science (autumn)
Orponen, Pekka, Professor, D.Phil.	T-79.149	Discrete Structures (autumn)
Tynjälä, Teemu, M.Sc. (Tech.) (cf. 1.5)	T-79.231	Parallel and Distributed Digital Systems (autumn)

1.4 Teaching assistants in the active courses of the year 2001

Honkola, Jukka, Stud.Tech. (cf. 1.5)	Tik-79.179 T-79.231	Parallel and Distributed Digital Systems (spring) Parallel and Distributed Digital Systems (autumn)
Junttila, Tommi, Lic.Sc. (Tech.) (cf. 1.5)	T-79.240 Tik-79.298	Special Course in Computational Complexity (autumn) Postgraduate Course in Digital Systems Science (spring)
Jussila, Toni, Lic.Sc. (Tech.) (cf. 1.5, 5.5)	T-79.144 Tik-79.148	Logic in Computer Science: Foundations (autumn) Introduction to Theoretical Computer Science (spring)
Kaski, Petteri, M.Sc. (Tech.) (cf. 1.5, 5.6)	Tik-79.161	Combinatorial Algorithms (spring)
Keinänen, Misa, M.A. (cf. 1.5)	T-79.144	Logic in Computer Science: Foundations (autumn)

Latvala, Timo, M.Sc. (Tech.) (cf. 1.5)	Tik-79.148 Tik-79.186	Introduction to Theoretical Computer Science (spring) Reactive Systems (spring)
Mäki, Silja, M.Sc. (cf. 1.5)	Tik-79.148	Introduction to Theoretical Computer Science (spring)
Oikarinen, Emilia, Stud.Tech. (cf. 1.5)	T-79.144	Logic in Computer Science: Foundations (autumn)
Parviainen, Elina, Stud.Tech. (cf. 1.5)	Tik-79.148 T-79.192	Introduction to Theoretical Computer Science (spring) Special Course in Theoretical Computer Science (autumn)
Pääkkönen, Rauni, M.Sc. (Tech.) (cf. 1.5, 5.6)	Tik-79.148	Introduction to Theoretical Computer Science (spring)
Syrjänen, Tommi, M.Sc. (Tech.) (cf. 1.5)	Tik-79.148 T-79.154 Tik-79.230	Introduction to Theoretical Computer Science (spring) Logic in Computer Science: Special Topics II (autumn) Foundations of Agent-Based Computing (spring)
Tauriainen, Heikki, M.Sc. (Tech.) (cf. 1.5)	Tik-79.146 Tik-79.148	Logic in Computer Science: Special Topics I (spring) Introduction to Theoretical Computer Science (spring)
Varpaaniemi, Kimmo, D.Sc. (Tech.) (cf. 1.1)	Tik-79.157 T-79.298	Formal Description and Verification of Computing Systems (spring) Postgraduate Course in Digital Systems Science (autumn)

1.5 Researchers and research assistants

Beaver, Harriet	Stud.Tech.	Research Assistant (until July 1)
Candolin, Catharina	Stud.Tech.	Research Assistant (since August 1)
Falck, Emil	Stud.Tech.	Research Assistant
Haanpää, Harri	Lic.Sc. (Tech.) (cf. 1.3)	Researcher
Heljanko, Keijo	Lic.Sc. (Tech.) (cf. 1.1, 1.3)	Researcher
Hietalahti, Maarit	M.Sc. (Tech.) (cf. 5.6)	Research Assistant (until July 1) Researcher (since July 1)
Honkola, Jukka	Stud.Tech. (cf. 1.4)	Research Assistant
Huima, Antti	M,Sc. (Tech.) (cf. 1.3)	Researcher (since August 1)
Junttila, Tommi	Lic.Sc. (Tech.) (cf. 1.4)	Researcher
Jussila, Toni	Lic.Sc. (Tech.) (cf. 1.4, 5.5)	Researcher
Järvisalo, Matti	Stud.Tech.	Research Assistant (since July 1)
Kallenbach, Jan	Stud.Tech. (cf. 4.3)	Trainee (August 6 – October 5)
Karilinna, Timo	Stud.Tech.	Research Assistant (March 1 – June 30)
Kaski, Petteri	M.Sc. (Tech.) (cf. 1.4, 5.6)	Research Assistant (until October 1) Researcher (since October 1)
Keinänen, Misa	M.A. (cf. 1.4)	Researcher (since May 1)
Lassila, Eero	Lic.Sc. (Tech.) (cf. 1.1)	Researcher
Latvala, Timo	M.Sc. (Tech.) (cf. 1.4)	Researcher
Mäkelä, Marko	Lic.Sc. (Tech.)	Researcher (since March 1)
Mäki, Silja	M.Sc. (cf. 1.4)	Researcher
Oikarinen, Emilia	Stud.Tech. (cf. 1.4)	Research Assistant (since June 1)
Parviainen, Elina	Stud.Tech. (cf. 1.4)	Research Assistant
Pyhälä, Tuomo	Stud.Tech.	Research Assistant (since June 1)
Pääkkönen, Rauni	M.Sc. (Tech.) (cf. 1.4, 5.6)	Research Assistant

Ruusu, Henna	Stud.Tech.	Research Assistant (June 1 – August 31) (November 1–30)
Syrjänen, Tommi	M.Sc. (Tech.) (cf. 1.4)	Researcher
Taurianen, Heikki	M.Sc. (Tech.) (cf. 1.4)	Researcher
Tynjälä, Teemu	M.Sc. (Tech.) (cf. 1.3)	Researcher
Östergård, Patric R.J.	D.Sc. (Tech.) (cf. 1.2)	Senior Fellow of the Academy of Finland (until August 1)

2 EDUCATIONAL ACTIVITIES

The aim of the education at the undergraduate level is to give students basic insight into theoretical computer science and parallel and distributed digital systems, as well as learning in applying the theoretical results to practice. At the post-graduate level knowledge in the aforementioned areas will be completed further, especially in some particular theoretical questions. During the year 2001, the following courses were active, i.e. arranged as lectures, seminars or projects.

T-79.144 Logic in Computer Science: Foundations
(autumn, 2 credits; main teacher: Tomi Janhunen)

Contents: Propositional and predicate calculus, their syntax, semantics and proof theory. Applications of logic in computer science.

Tik-79.146 Logic in Computer Science: Special Topics I
(spring, 2 credits; main teacher: Ilkka Niemelä)

Contents: Basics of modal logic. Current applications in computer science.

Tik-79.148 Introduction to Theoretical Computer Science
(spring, 2 credits; main teacher: Rauno Heinonen)

Contents: Basics of the theory of formal languages and automata. Theory of computation. Fundamental limitations of computers.

Tik-79.149 Discrete Structures
(spring, 2 credits; main teacher: Nisse Husberg)

Contents: The basic mathematics underlying formal descriptions, especially (universal and heterogeneous) algebra and category theory.

T-79.149 Discrete Structures
(autumn, 2 credits; main teacher: Pekka Orponen)

Contents: The fundamental techniques of enumerative combinatorics, with applications to the analysis of algorithms. Emphasis on the methodology of generating functions, especially their formal construction, and the exact and asymptotic analysis of coefficient sequences using e.g. tools from complex function theory.

T-79.154 Logic in Computer Science: Special Topics II
(autumn, 2 credits; main teacher: Tomi Janhunen)

Contents: Efficient implementation methods for propositional logic. Logical foundations and implementation techniques of rule-based systems. Current applications.

Tik-79.157 Formal Description and Verification of Computing Systems
(spring, 2 credits; main teacher: Leo Ojala)

Contents: The use of net theoretic models in the exact modelling of distributed algorithms and the efficient verification of the models. Applications primarily to communication protocols.

Tik-79.159 Cryptography and Data Security
(spring, 3 credits; main teacher: Antti Huima)

Contents: Cryptographic algorithms and protocols. Modern symmetric and asymmetric encryption. Digital signatures. Protection of integrity. Cryptographic hash functions. Authentication protocols. Key exhange and identification protocols. Design and analysis of cryptographic protocols. Electronic commerce. Steganography. New directions in data security.

Tik-79.161 Combinatorial Algorithms
(spring, 2 credits; main teacher: Harri Haanpää)

Contents: Basic algorithms and computational methods for combinatorial problems. Combinatorial structure generation (e.g. permutations). Search methods. Graph algorithms and combinatorial optimization. Algorithm complexity.

Tik-79.179 Parallel and Distributed Digital Systems
(spring, 3 credits; main teacher: Esa Kettunen)

Contents: Modelling and analysis of parallel and distributed digital systems. Concurrency. Basics of Petri nets and process algebra (CCS). Using computer-aided methods for the analysis and verification of telecommunication systems, especially communication protocols.

T-79.185 Verification
(autumn, 3 credits; main teacher: Nisse Husberg)

Contents: Verification and analysis of parallel and distributed systems using tools. Applications to telecommunication protocols. Practical verification methods, e.g. partial reachability analysis. Introduction to current research problems.

Tik-79.186 Reactive Systems
(spring, 2 credits; main teacher: Keijo Heljanko)

Contents: Specification and verification of reactive systems with temporal logic. Basics of computer-aided verification methods and their algorithms.

T(ik)-79.189 Student Project in Theoretical Computer Science
(3 credits; main teacher: Ilkka Niemelä)

Contents: Independent student project on a subject from the field of theoretical computer science or digital systems.

T-79.192 Special Course in Theoretical Computer Science
(autumn, 2 credits; main teacher: Leo Ojala)

Contents: Current applications in theoretical computer science.

Tik-79.193 Formal Description Techniques for Concurrent Systems
(spring, 2 credits; main teacher: Nisse Husberg)

Contents: Validation, testing and analysis methods for large concurrent systems, embedded systems and real-time software.

Tik-79.194 Seminar on Theoretical Computer Science
(spring, 2 credits; main teacher: Ilkka Niemelä)

Contents: Current trends and research problems in theoretical computer science.

Tik-79.230 Foundations of Agent-Based Computing
(spring, 3 credits; main teacher: Tomi Janhunen)

Contents: Structure of software agents. Rational and intelligent agents. Architectures, implementation technologies and applications of agent-based systems.

T-79.231 Parallel and Distributed Digital Systems
(autumn, 3 credits; main teacher: Teemu Tynjälä)

Contents: Modelling and analysis of parallel and distributed digital systems. Concurrency. Basics of Petri nets and process algebra (CCS). Using computer-aided methods for the analysis and verification of telecommunication systems, especially communication protocols.

Tik-79.232 Safety-Critical Systems
(spring, 2 credits; main teacher: Ilkka Herttua)

Contents: Safety-critical systems. The use of formal methods in the specification, modelling and verification of systems.

T-79.240 Special Course in Computational Complexity
(autumn, 3 credits; main teacher: Ilkka Niemelä)

Contents: NP-completeness. Randomized algorithms. Cryptography. Approximation algorithms. Parallel algorithms. Polynomial hierarchy. PSPACE-completeness.

T(ik)-79.295 Individual Studies
(1–10 credits; main teacher: Ilkka Niemelä)

Contents: Individual studies on a subject from the field of theoretical computer science or digital systems.

T(ik)-79.298 Postgraduate Course in Digital Systems Science
(10 credits; main teacher: Ilkka Niemelä)

Contents: Insight into current research problems in theoretical computer science.

T-79.510 Seminar on Cryptography and Security Protocols
(autumn, 3 credits; main teacher: Helger Lipmaa)

Contents: This is an advanced undergraduate or graduate level seminar on the subfield of data security. The goal of the students is to write and present a research paper in English. This course can be seen as a graduate level and more cryptography-oriented continuation of T-110.501 (Seminar on Network Security).

T-79.511 Special Course on Cryptology
(autumn, 2 credits; main teacher: Helger Lipmaa)

Contents: This is a graduate level course that every semester concentrates on one concrete area of cryptology.

3 RESEARCH ACTIVITIES

A major part of the research has been funded by the Academy of Finland with substantial support from Helsinki Graduate School in Computer Science and Engineering (HeCSE). More details on this research is given in Sections 3.1, 3.2, 3.3, and 3.4. For more applied research funding has been awarded by the National Technology Agency of Finland as well as companies and other non-academic partners. This research is described in Sections 3.5, 3.6, and 3.7.

3.1 Formal methods in distributed systems

This subsection describes research which during the year 2001 was carried out by Ilkka Niemelä (the leader), Harrier Beaver, Keijo Heljanko, Tomi Janhunen, Tommi Junttila, Leo Ojala, Elina Parviainen, Olli-Matti Penttinen, Rauni Pääkkönen, Heikki Tauriainen, and Teemu Tynjälä.

This is a basic research project on analysis and design methods of parallel and distributed systems. It focuses on model checking, symmetries, agent-based computing, and quantum computing. Below more details on the research are given.

Prefix-based model checking (*Keijo Heljanko*)

Research concentrated on using symbolic methods to alleviate the state explosion problem in model checking. The main approach used is a method called *complete finite prefixes* originally devised by McMillan. We have implemented a state based linear temporal logic (LTL) model checker based on finite complete prefixes called `unfsmodels` [56]. This work is described in the publications [19, 33]. We have also worked on efficient implementation methods for prefix generation. The main result is the parallelisation of a prefix generation procedure [34].

Testing implementations of algorithms for translating linear time temporal logic formulae into Büchi automata
(*Heikki Tauriainen and Keijo Heljanko*)

Automata-theoretic model checking tools for linear time temporal logic (LTL) use algorithms which translate LTL properties into Büchi automata. These algorithms have to be implemented very carefully to ensure the correctness of model checking results in practice. In this research we have devised methods for detecting errors in LTL-to-Büchi translation algorithm implementations by (i) checking for known relationships between a pair of automata obtained from two complementary LTL formulae and (ii) comparing the model checking results obtained using independent LTL-to-Büchi translation algorithm implementations. Incorrect implementations are identified with the help of a restricted LTL model checking algorithm for single computation paths. Most of the test methods have been integrated into the `lbtt` software package [63].

Symmetries in verification (*Tommi Junttila*)

The symmetry reduction method is a way to alleviate the state space explosion problem occurring in the state space analysis of concurrent systems. It exploits the symmetries (automorphisms) of the state space by considering only one representative state per each set of mutually symmetric states. Thus a potentially much smaller set of states have to be considered during the state space analysis. Our work is concentrated on the application of the symmetry reduction method to Petri nets, both high- and low-level level nets. During the year 2001, we published results concerning the computational complexity of the sub-problems appearing in the method when applied to a class of low-level nets, namely Place/Transition nets [7]. In addition, the work still in progress has been concentrated on the core algorithms needed in the symmetry reduction method, namely ones comparing whether two states are symmetric or not and producing a representative state for a given state.

Agent-based framework (*Tomi Janhunen and Rauni Pääkkönen*)

The aim of this research is to create a framework for agent-based computing where agents have communication and coordination capabilities in order to operate in a heterogenous and distributed environment, and agents perform non-trivial reasoning tasks. We have previously brought forth *agent programs* as declarative specifications of for multi-agents systems. Such programs were obtained by synthesizing two existing formalisms: (i) Petri nets which have been introduced as models of parallel and distributed systems and (ii) logic programs that capture rule-based knowledge representation and reasoning. A particular attention was paid to keep the formalism computationally feasible. In 2001, we concentrated on tuning/generalizing the formalism and its implementation [48] under the Linux Debian operating system. The implementation consists of an interpreter which operates according to the agent description that it receives as its input. Our prototypical agent architecture enables distributed coordination of processes in the Linux environment.

Modeling Feynman's quantum computer using high-level Petri nets (*Harriet Beaver, Leo Ojala, Elina Parviainen, Olli-Matti Penttinen, and Teemu Tynjälä*)

Petri Nets have been successfully used to model systems based on classical physics. The aim of our study is to model Feynman's quantum computer, one of the first quantum computers suggested, using high-level Petri Nets. Feynman's quantum computer is based on a circuit of quantum logic gates in a similar way as classical computers are based on boolean gates and circuits. At the time of invention, Feynman could not give a time bound for the completion of his computer's computation; a periodical measuring procedure was needed. The periodical measurement allows us to use a computational approach. We model the use and operation of Feynman's computer, in [27] using predicate/transition nets and in [26] using stochastic high-level Petri nets.

3.2 Constraint programming based on default rules

This subsection describes research which during the year 2001 was carried out by Ilkka Niemelä (the leader), Keijo Heljanko, Tomi Janhunen, Tommi Junttila, Toni Jussila, and Tommi Syrjänen.

The goal of the research is to develop a novel constraint programming paradigm based on default rules and to study its applications. The project has focused on logic program type rules and the stable model semantics. We have developed a C++ implementation of the approach, the `Smodels` system [31], which is among the leading systems in the area and used in dozens of research groups all over the world. `Smodels` is available via http://www.tcs.hut.fi/Software/smodels/index.html. In 2001, we have extended the basic language of `Smodels`, studied expressivity issues and investigated various applications areas. More details are given below.

Extending the rule language (*Ilkka Niemelä and Tommi Syrjänen*)

In many applications normal logic program rules lack expressivity to handle cardinalities, weights and optimization. We have developed an extended rule language which allows for cardinality and weight constraints and optimization capabilities and devised a generalization of the stable model semantics for it. The extended rule language allows also the use of logical variables, function symbols and built-in arithmetic. In order to keep the language decidable the rules are required to be domain-restricted such that the domain of each variable is defined using a domain predicate. We have devised a method for allowing recursive definitions of domain predicates and studied the expressivity and computational complexity of the resulting rule language [29]. We have also developed general methods for automating different forms of nonmonotonic reasoning more general than normal logic programs with stable model semantics [15] and studied in detail the relationship between different semantics of disjunctive logic programs [3].

Expressive power analysis of rule-based languages
(*Tomi Janhunen*)

This research continues our earlier work on classifying non-monotonic logics by their expressive powers. Our classification method is based on analyzing the existence of polynomial, faithful and modular (PFM) translations between non-monotonic logics under consideration. We have applied an analogous framework for studying the expressiveness of classes of logic programs under the stable model semantics proposed by Gelfond and Lifschitz. In 2001, certain syntactic classes of disjunctive logic programs were taken into consideration. In particular, the role of default negation was examined [23]. It was established that (i) default negation can be removed from the heads of rules using a PFM translation if and only if default negation is allowed in the bodies of rules. Moreover, (ii) there is no PFM translation for removing default negation from the bodies of rules. These relationships form the basis for the expressive power hierarchy (EPH) of disjunctive logic programs.

Software configuration management (*Tommi Syrjänen*)

Modern software products are large and complex and they may contain hundreds or thousands of interacting components. Also, software products are generally volatile in the sense that new versions of individual components are introduced regularly. The aim of software configuration management research is to find new methods for representing configuration knowledge and constructing valid configurations that satisfy user requirements. The current software configuration research in the laboratory concentrates on developing high-level rule-based methods for expressing configuration knowledge using the stable model semantics of normal logic programs as a formal framework. In 2001 the problem of handling complex version spaces where each component may have numerous different variants was addressed in [30].

Product configuration (*Ilkka Niemelä*)

Together with the product data management group at Helsinki University of Technology (Timo Soininen, Juha Tiihonen, Reijo Sulonen) we have developed general methodology for product configuration [28]. It has turned out that the new types of rules supported by `Smodels` play an important role in representing configuration knowledge in a compact and maintainable form. Moreover, `Smodels` provides a promising inference engine on top of which intelligent automatic configurators can be built.

Bounded model checking (*Keijo Heljanko and Ilkka Niemelä*)

Bounded model checking has been recently introduced as a memory efficient way of locating errors in reactive systems. We have continued to work on bounded model checking using both the `BCSat` [57] and the `Smodels` system developed in the laboratory as the underlying NP-solvers. The work with `BCSat` has concentrated on efficiently using the parallelism present in the model to speed up bounded model checking [20] and has been implemented in the `punroll` [54] reachability and deadlock checker. The work based on `Smodels` has resulted in an improved reachability property checker [21], as well as a new LTL model checking translation [22]. Both of these translations are implemented in a tool called `boundsmodels` [55].

Bounded Model Checking for Concurrent Programs
(*Toni Jussila and Ilkka Niemelä*)

In this work bounded model checking techniques developed in the project for verifying 1-safe Petri Nets [20, 21, 22] have been applied and extended to concurrent software. A parallel programming language, called `SPINB`, has been developed together with its operational semantics. The basis for applying bounded model checking techniques has been given by devising a translation from valid `SPINB` programs to propositional logic such that given a bound the models of the propositional translation of a `SPINB` program correspond to the bounded length prefixes of the executions of the program [42]. It is then shown how bounded model checking of reachability and safety properties can be handled using the translation [42].

Boolean circuit satisfiability checking
(*Tommi Junttila and Ilkka Niemelä*)

Propositional satisfiability (SAT) checking can be seen as a special case of stable model computation for logic program type rules. As this case appears frequently in applications, special purpose methods for it have been developed using ideas from the implementation techniques for stable model computation developed in the project. Most state of the art SAT checkers require that the input must be transformed into conjunctive normal form (CNF) and the algorithms are based on working with CNF formulae. We decided to study an alternative approach where Boolean circuits are used as the input format for the SAT checker. Boolean circuits provide a natural and compact way of encoding problems allowing structure sharing. A tableau algorithm for solving satisfiability problems has been developed. It works directly on Boolean circuits without any CNF transformation.

During the year 2001, a new, more efficient version of the C++ implementation of the algorithm, the `BCSat` system, was released. It is available via http://www.tcs.hut.fi/%7etjunttil/bcsat/index.html During the year, the system has also been applied to bounded model checking problems [20].

3.3 Computational methods in coding theory and discrete mathematics

This subsection describes research which during the year 2001 was carried out by Patric R.J. Östergård (the leader), Harri Haanpää, and Petteri Kaski. During the year 2001, the research project contributed to the publications [1, 2, 4, 5, 6, 8, 11, 12, 13, 14, 35, 44].

The aim of the research is the study of existence and enumeration problems in coding theory and discrete mathematics using computational methods, and enhancing these by algebraic and combinatorial results. The methods are developed in a general framework, and have been applied to numerous discrete structures, such as codes, designs, and graphs, just to mention a few. They have also been applied to a variety of practical problems, many of which are related to telecommunications. Both exhaustive and stochastic methods are used.

The stochastic methods used include simulated annealing, tabu search, and evolutionary algorithms; however, almost without exception, tabu search has turned out to yield the best performance for the problems under study.

As for exhaustive methods, the main focus has been on orderly generation of discrete structures. Using these, classification results have been obtained for various structures, including balanced incomplete block designs (BIBDs), resolvable and almost resolvable BIBDs, whist tournaments, covering arrays, covering codes, linear codes, etc. Structures for which other (algebraic, combinatorial, and computational) methods have been applied include Bhaskar Rao designs, complete caps, constant-composition codes, constant weight codes, error-correcting codes, etc. One of the main results obtained is a computer-aided classification of the Steiner triple systems of order 19.

Many of the computational results have been obtained with very CPU-intensive computations, some of which have been distributed using the distributed batch system `autoson` over the entire computer network of the laboratory and, in classifying the STS(19)s, dozens of Linux PCs of the HUT Computing Centre.

3.4 Generative string rewriting

This subsection describes research that during the year 2001 was carried out by Eero Lassila. During the year 2001, the research project contributed to the publication [36].

The aim of this research is to enable the parallelization of context-sensitive rewriting operations without disrupting the semantics of the string under rewriting. Specifically, it should be possible to freely adjust the degree of parallelism in the rewriting process without any need to modify the rewriting rules. Such an adjustment is allowed to change the structure but not the semantics of the output. Even if the introduction of parallelism in one hand dampens optimization, it on the other hand enhances maintainability of the rewriting rules.

The concrete short-term goal of this research is to devise a general formal model, and the long-term one is to apply the model to practical tasks like optimizing code generation.

3.5 The `ANNA–MARIA` project

This subsection describes research which during the year 2001 was carried out by Nisse Husberg (the leader), Annikka Aalto, Emil Falck, Jukka Honkola, Timo Karilinna, Timo Latvala, Marko Mäkelä, Henna Ruusu, Teemu Tynjälä, and Kimmo Varpaaniemi. During the year 2001, the research project contributed to the publications [9, 10, 24, 25, 26, 27, 32, 37, 40, 53, 58, 59, 60, 61].

The one-year-project `ANNA–MARIA` was a sequel to the `MARIA` project that during the years 1998–2000 developed the verification and debugging tool `MARIA` for industrial size concurrent systems. The `ANNA–MARIA` project was funded by the National Technology Agency of Finland, Nokia Research Center, Nokia Networks, EKE Electronics, and Genera.

The `MARIA` tool was improved in many aspects. Model checking of linear time temporal logic formulas was improved by taking advantage of the observation that classical finite automata are more practical than Büchi automata in detection of counterexamples to safety properties. State space explosion was alleviated by making the tool recognise and utilise symmetries that occur in system descriptions. A prototype of an SDL Z.100 front-end to `MARIA` was achieved, and the TNSDL front-end `EMMA` to the analysis tool `PROD` was revised to be a front-end to `MARIA`. Interconnections between `MARIA` and other tools were created. Experimental versions of graphical user interfaces were implemented.

RLC, a UMTS radio network layer protocol, was modelled and analysed. In order to alleviate state space explosion, much effort was put in restricting the behaviour of the model, using the idea that as long as errors can be caught and fixed one at a time, representing more than one error at a time is not a primary goal. In another case study, graphical operator panel application software was modelled and analysed. It turned out that it is not totally unmotivated to try to find errors from a system of the kind by using `MARIA`, but it might be a good idea to try to find and use some other tools as well.

There was also some basic research that was indirectly connected to the other research. The stubborn set method can be used e.g. for constructing an LTS (a labelled transition system) that is CFFD- or CSP-equivalent to a parallel composition of a given collection of LTSs. The method tries to alleviate combinatorial explosion. It was found out how to optimise the method in such a way that for each state of the LTS being constructed, the least possible number (w.r.t. certain computation conditions) of immediate successor states becomes generated.

3.6 Analysis and development of cryptosystems

This subsection describes research that during the year 2001 was carried out by Ilkka Niemelä (the leader), Helger Lipmaa, Maarit Hietalahti, and Silja Mäki. During this project, we studied the security of different cryptographic primitives—that is, of basic algorithms that are used for composing more complex cryptographic protocols starting from SSL and IPSec and ending with digital cash and e-voting. We evaluated the security of existing cryptosystems. Our studies resulted in (i) an overview of the security of different public key cryptosystems, and (ii) computing of differential properties of several simple yet hard to evaluate algebraic functions.

3.7 Security in mobile ad-hoc networks

This subsection describes research that during the year 2001 was carried out by Ilkka Niemelä (the leader), Hannu H. Kari, Silja Mäki, Catharina Candolin, and Maarit Hietalahti. During the year 2001, the research project contributed to the publications [18, 43].

An ad hoc network is a collection of nodes that do not need to rely on a predefined infrastructure to establish and maintain network communications. Nodes may have varying mobility rate, ranging from fully static nodes to frequently moving nodes. Furthermore, the network may be dynamic, that is, nodes enter and leave the network on frequent basis. Securing such networks becomes difficult, because the nodes may not have any prior knowledge of each other, the network medium tends to be wireless and is thus more vulnerable to attacks, and the dynamic nature of the network render many solutions for problems in traditional networks unsuitable. One example is trust management: the main problem in traditional networks is establishing trust, whereas the main problem in ad hoc networks is maintaining trust. Ad hoc networks are also heavily dependent on their environment, for example, the requirements specified for small sized meeting room networks, hot spot networks, disaster site networks, and hostile environment networks, are very different.

The focus of the project is to develop security solutions for mobile ad hoc networks in a hostile environment. This includes securing network functions, such as routing, mobility management, and network management. Hostility refers to the fact that the environment consists of malicious entities that actively try to interfere with network communications, e.g. by launching denial-of-service attacks or inducing false information to protocols.

In 2001, the project focused its research on developing an efficient key agreement protocol for groups [43], dynamic group management, and incomplete trust management. Furthermore, the project was active on related issues, such as addressing security issues of IPv6 [18].

4 CONFERENCES, VISITS AND GUESTS

Every personified activity mentioned in Section 4.1 is such that the person in question was a member of the laboratory at the time of the activity. Every bibliographic pointer from Section 4.1 to Section 5.3 is such that the mentioned presenter of the paper in question was a member of the laboratory at the time of the presentation.

4.1 Conferences

January

HeCSE Winter School, Valkeala, Finland, January 8–9. A talk given by Harri Haanpää (*Sets in Abelian Groups with Distinct Sums of Pairs*).

February

Research Seminar of the Finnish Defence Forces Technical Research Centre, Riihimäki, Finland, February 8. Participants: Maarit Hietalahti, Tomi Janhunen, Silja Mäki, and Ilkka Niemelä.

March

EWSCS (The 6th Estonian Winter School in Computer Science), Palmse, Estonia, March 4–9. Talks given by Maarit Hietalahti (*Key Establishment in Ad-hoc Networks*) and Silja Mäki (*On Long-Lived Public-Key Traitor Tracing. First steps.*). http://www.cs.ioc.ee/yik/schools/win2001/

FME (The 10th International Symposium of Formal Methods Europe: Formal Methods for Increasing Software Productivity), Berlin, Germany, March 12–16. Participants: Nisse Husberg and Toni Jussila.
http://gsyc.escet.urjc.es/lists/brunello/msg00083.html

FEmSys (Workshop on Formal Design of Safety Critical Embedded Systems), Munich, Germany, March 21–23. Participant: Kimmo Varpaaniemi. http://ais.gmd.de/%7eap/femsys/

AAAI Spring Symposium on Answer Set Programming: Towards Efficient and Scalable Knowledge Representation and Reasoning, Stanford CA, USA, March 26–28. Talks given by Ilkka Niemelä ([21] and *Smodels: A System for Answer Set Programming*). A session chaired by Niemelä. http://www.cs.nmsu.edu/%7etson/ASP2001/

April

ETAPS (European Joint Conferences on Theory and Practice of Software), Genova, Italy, April 2–6. Participant: Timo Latvala. http://www.disi.unige.it/etaps2001/

DGNMR (The 5th Dutch-German Workshop on Nonmonotonic Reasoning Techniques and their Applications), Potsdam, Germany, April 4–6. A talk given by Tomi Janhunen (*On the Effect of Default Negation on the Expressiveness of Disjunctive Rules*). http://www.cs.utwente.nl/data/amast/mail/2000/10/msg00039.html

International Workshop on Security Protocols, Cambridge, UK, April 25–27. A talk given by Silja Mäki (*Towards a Survivable Security Architecture for ad-hoc Networks*). http://homepages.feis.herts.ac.uk/%7ecomqjam/2001SPW-running-order-v03.htm

May

The 8th International SPIN Workshop on Model Checking of Software, Toronto, Canada, May 19–20. A talk given by Keijo Heljanko [19]. http://www.cis.ksu.edu/santos/spin2001/

UC Berkeley – HeCSE Summer School on Telecommunications Architectures, Berkeley CA, USA, May 29–30. A poster presented by Silja Mäki (*Towards survivable membership management in ad-hoc groups*). http://www.cs.helsinki.fi/u/kraatika/Courses/TSASummerSchool.html

3Gwireless (International Conference on Third Generation Wireless and Beyond), San Francisco CA, USA, May 30 – June 2. Participant: Silja Mäki. http://delson.org/3gwireless01/

June

ICATPN (The 22nd International Conference on Application and Theory of Petri Nets), Newcastle upon Tyne, UK, June 25–29. Talks given by Timo Latvala [24] and Marko Mäkelä [25]. The MARIA tool demonstrated by Marko Mäkelä. Other participants: Nisse Husberg, Leo Ojala, and Elina Parviainen. http://www.cs.ncl.ac.uk/conferences/2001/pn/

ICACSD (The 2nd International Conference on Application of Concurrency to System Design), Newcastle upon Tyne, UK, June 25–29. Participants: Nisse Husberg, Timo Latvala, Marko Mäkelä, Leo Ojala, and Elina Parviainen. http://www.cs.ncl.ac.uk/conferences/2001/acsd/

Workshop on Concurrency in Dependable Computing, Newcastle upon Tyne, UK, June 26. Participant: Elina Parviainen. http://www.cs.ncl.ac.uk/conferences/2001/condepend/

The 10th SDL Forum, Copenhagen, Denmark, June 26–29. Participant: Teemu Tynjälä. http://www.sdl-forum.org/Events/10thsdl.htm

July

FMICS (The 6th International Workshop on Formal Methods for Industrial Critical Systems), Paris, France, July 16–17. Participant: Kimmo Varpaaniemi. http://www.dsse.ecs.soton.ac.uk/FMICS2001/

CAV (The 13th International Conference on Computer-Aided Verification), Paris, France, July 18–23. Participant: Keijo Heljanko. http://www.lsv.ens-cachan.fr/cav01/

SoftMC (Workshop on Software Model Checking), Paris, France, July 23. Participant: Keijo Heljanko. http://www.cs.sunysb.edu/%7estoller/softmc01/

SCI / ISAS (The 5th World Multiconference on Systems, Cybernetics and Informatics; The 7th International Conference on Information Systems, Analysis and Synthesis), Orlando FL, USA, July 22–25. A talk given by Teemu Tynjälä [27]. http://www.iiis.org/sci2002/menusci2001.htm

August

IJCAI (The 17th International Joint Conference on Artificial Intelligence), Seattle WA, USA, August 4–10. Participants: Tomi Janhunen and Tommi Syrjänen. http://www.ijcai-01.org/

IJCAI Workshop on Configuration, Seattle WA, USA, August 6. A talk given by Tommi Syrjänen [30]. http://www.soberit.hut.fi/pdmg/IJCAI2001ConfWS/

ESSLLI (The 13th European Summer School in Logic, Language and Information), Helsinki, Finland, August 13–24. A plenary talk given by Ilkka Niemelä (*Answer-Set Programming: a Declarative Knowledge Representation Paradigm*). A session chaired by Niemelä. Niemelä is a member of the programme committee of the Student Session. http://www.helsinki.fi/esslli/

CRYPTO (The 21st Annual International Cryptology Conference), Santa Barbara CA, USA, August 19–23. Participants: Maarit Hietalahti, Helger Lipmaa, and Silja Mäki. http://www.iacr.org/conferences/c2001/

The 2nd Modes of Operation Workshop, Goleta CA, USA, August 24. Participant: Helger Lipmaa. http://csrc.nist.gov/encryption/modes/workshop2/

CONCUR (The 12th International Conference on Concurrency Theory), Aalborg, Denmark, August 21–24. A talk given by Keijo Heljanko [20]. http://concur01.cs.auc.dk/index1a.html

FATES (Workshop on Formal Approaches to Testing of Software), Aalborg, Denmark, August 25. Participant: Keijo Heljanko. http://fmt.cs.utwente.nl/conferences/fates/

HeCSE Summer School, Espoo, Finland, August 27–28. A talk given by Keijo Heljanko (*Bounded Reachability Checking with Process Semantics*).

MOCA (Workshop on Modelling of Objects, Components, and Agents), Aarhus, Denmark, August 27–28. Participants: Nisse Husberg and Elina Parviainen. Husberg is a member of the programme committee. http://www.informatik.uni-hamburg.de/TGI/events/moca01/MOCA01-Programme.html

CPN (The 3rd Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools), Aarhus, Denmark, August 29–31. A session chaired by Nisse Husberg. Other participants: Leo Ojala and Elina Parviainen. Husberg is a member of the programme committee. http://www.daimi.au.dk/CPnets/workshop01/programme.html

September

Symposium on Brains, Genes and Chips, Stockholm, Sweden, September 10–12. Participant: Pekka Orponen. (Summary: *Brains, Genes and Chips — Information Processing in Biological and Man-made Systems*, ISBN 91-631-2050-X.)

International Multiconference of Measurement, Modelling, and Evaluation of Computer-Communication Systems, Aachen, Germany, September 11–14. Participant: Leo Ojala. http://www.informatik.unibw-muenchen.de/mmb/mmb40/programmheft.pdf

LPNMR (The 6th International Conference on Logic Programming and Nonmonotonic Reasoning), Vienna, Austria, September 17–19. Talks given by Tomi Janhunen [23], Ilkka Niemelä [22], and Tommi Syrjänen [29, 31]. A session chaired by Niemelä. Niemelä is a member of the programme committee. http://www.kr.tuwien.ac.at/lpnmr01/

ECSQARU (The 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty), Toulouse, France, September 19–21. Ilkka Niemelä is a member of the programme committee. http://www.irit.fr/ECSQARU-2001/Ecsqaru-2001.html

October

CS&P (Workshop on Concurrency, Specification and Programming), Warsaw, Poland, October 3–5. A talk given by Kimmo Varpaaniemi [32]. http://alfa.mimuw.edu.pl/csp2001/

NWPT (The 13th Nordic Workshop on Programming Theory), Lyngby, Denmark, October 10–12. Participants: Nisse Husberg and Toni Jussila. http://www.imm.dtu.dk/nwpt01/

Autumn Seminar of the Institute of Cybernetics (Estonia), Roosta, Estonia, October 11–12. Participant: Helger Lipmaa. http://www.cs.ioc.ee/ioc/teated/roosta01/

November

NordSec (The 6th Nordic Workshop on Secure IT Systems), Lyngby, Denmark, November 1–2. A talk given by Catharina Candolin [18]. http://www.imm.dtu.dk/%7enordsec/

TietoEnator's seminar, Helsinki, Finland, November 21. A talk given by Hannu H. Kari (*Extinction of Dinosaurs — End of the Era of Telecom Operators*).

ICLP (The 17th International Conference on Logic Programming), Paphos, Cyprus, November 26 – December 1. Participant: Misa Keinänen. http://www.cs.ucy.ac.cy/%7eiclpcp01/ICLP01home.html

CP (The 7th International Conference on Principles and Practice of Constraint Programming), Paphos, Cyprus, November 26 – December 1. Participant: Misa Keinänen. http://www.cs.ucy.ac.cy/%7eiclpcp01/CP01home.html

December

CICLOPS (Colloquium on Implementation of Constraint and Logic Programming Systems), Paphos, Cyprus, December 1. Participant: Misa Keinänen. Ilkka Niemelä is a member of the programme committee. http://www.cs.nmsu.edu/%7ecomplog/conferences/iclp01/

Seminar on Evolution in Metapopulations, Turku, Finland, December 13. Participant: Pekka Orponen. http://users.utu.fi/evakis/metasem.htm

4.2 Visits

Catharina Candolin visited the Tartu Information and Communication Technology Development Center, Estonia, on October 22–24. Talk: *Network security.* http://tiktak.tartu.ee/

Nisse Husberg visited the Computer Science Laboratory of Mälardalen University, Västerås, Sweden, on April 6. Talk: *`EMMA` and `MARIA`: analysers for industrial systems specified in SDL.*

http://www.mrtc.mdh.se/csl/cslevents.phtml?start=0&past=1

Husberg gave an invited talk about broadband nets in Kristiinankaupunki, Finland, on April 19. Inviter: the town council of Kristiinankaupunki.

Husberg gave an invited talk about broadband nets in Närpiö, Finland, on May 9. Inviter: the town council of Närpiö.

http://vblnetra1.vasabladet.fi/nyheter/010510/nyhet9.html

On November 24–27, Husberg visited the European Parliament, Brussels, Belgium, as a guest of MEP Astrid Thors.

4.3 Guests

D.Sc. (Tech.) Tuomas Aura from Microsoft Research, Cambridge, UK, stayed for one day, gave a talk (*Mobile IP Security*) on December 5, and was hosted by Ilkka Niemelä.

Stud.Tech. Jan Kallenbach (cf. 1.5) from Ilmenau Technical University, Germany, stayed for two months and was hosted by Tomi Janhunen.

Professor Ekkart Kindler from Technical University of Munich, Germany, stayed for one day, gave a talk (*Petri Nets and Components*) on November 2, and was hosted by Nisse Husberg.

Postgraduate student Louise Lorentsen from University of Aarhus, Denmark, stayed for one day, gave a talk (*Modelling Feature Interaction Patterns in Nokia Mobile Phones using Coloured Petri Nets*) on May 4, and was hosted by Nisse Husberg.

Professor Wolfgang Reisig from Humboldt University Berlin, Germany, stayed for two days, gave a talk (*Recent Results on Distributed Algorithms*) on June 8, and was hosted by Nisse Husberg.

Stud.Tech. André Schulz from Brandenburg University of Technology Cottbus, Germany, stayed for one day and was hosted by Nisse Husberg.

Professor Mirosław Truszczyński from University of Kentucky, Lexington KY, USA, stayed for six days, gave a talk (*Fixed-Parameter Complexity of Semantics for Logic Programs*) on August 22, and was hosted by Ilkka Niemelä.

Professor Moshe Y. Vardi from Rice University, Houston TX, USA, stayed for one day, gave a talk (*The Design of a Formal Property-Specification Language*) on August 17, and was hosted by Ilkka Niemelä.

Dipl.-Ing. Michael Weber from Humboldt University Berlin, Germany, stayed for two days and was hosted by Nisse Husberg.

5 PUBLICATIONS

5.1 Journal articles

[1] Galina T. Bogdanova, Andries E. Brouwer, Stoian N. Kapralov, and Patric R.J. Östergård, "Error-Correcting Codes over an Alphabet of Four Elements," *Designs, Codes and Cryptography*, Vol. 23, No. 3, pp. 333–342.

[2] Galina T. Bogdanova and Patric R.J. Östergård, "Bounds on codes over an alphabet of five elements," *Discrete Mathematics*, Vol. 240, No. 1–3, pp. 13–19. http://www.elsevier.nl/gej-ng/10/16/24/170/27/28/abstract.html

[3] Stefan Brass, Jürgen Dix, Ilkka Niemelä, and Teodor C. Przymusinski, "On The Equivalence of the Static and Disjunctive Well-Founded Semantics and its Computation," *Theoretical Computer Science*, Vol. 258, No. 1–2, pp. 523–553. http://www.elsevier.nl/gej-ng/10/41/16/200/21/34/abstract.html

[4] Alexander A. Davydov and Patric R.J. Östergård, "Linear codes with covering radius R=2,3 and codimension tR," *IEEE Transactions on Information Theory*, Vol. 47, No. 1, pp. 416–421. http://ieeexplore.ieee.org/iel5/18/19571/00904551.pdf?isNumber=19571

[5] Alexander A. Davydov and Patric R.J. Östergård, "Recursive constructions of complete caps," *Journal of Statistical Planning and Inference*, Vol. 95, No. 1–2, pp. 167–173. http://www.elsevier.nl/gej-ng/10/29/19/88/25/37/abstract.html

[6] T. Aaron Gulliver and Patric R.J. Östergård, "Improved bounds for ternary linear codes of dimension 8 using tabu search," *Journal of Heuristics*, Vol. 7, No. 1, pp. 37–46.

[7] Tommi Junttila, "Computational Complexity of the Place/Transition-Net Symmetry Reduction Method," *Journal of Universal Computer Science*, Vol. 7, No. 4, pp. 307–326. http://www.jucs.org/jucs_7_4/computational_complexity_of_the

[8] Petteri Kaski and Patric R.J. Östergård, "There exists no (15,5,4) RBIBD," *Journal of Combinatorial Designs*, Vol. 9, No. 5, pp. 357–362. http://www3.interscience.wiley.com/cgi-bin/issuetoc?ID=85009579

[9] Marko Mäkelä, "Formaalit menetelmät mutkikkaiden järjestelmien suunnittelun apuna — Rinnakkaisuus hallintaan (Formal Methods for Concurrent Systems)," *Prosessori (erikoisnumero: Elektroniikan suunnittelu)*, pp. 68–71. http://www.prosessori.fi/arkisto/hakutulos2.asp?haku=13/2001

[10] Leo Ojala, Nisse Husberg, and Teemu Tynjälä, "Modelling and analysing a distributed dynamic channel allocation algorithm for mobile computing using high-level net methods," *International Journal on Software Tools for Technology Transfer*, Vol. 3, No. 4, pp. 382–393. http://link.springer.de/link/service/journals/10009/bibs/1003004/10030382.htm

[11] Patric R.J. Östergård, "A new algorithm for the maximum-weigth clique problem," *Nordic Journal of Computing*, Vol. 8, No. 4, pp. 424–436.

[12] Patric R.J. Östergård, "There are 270,474,142 nonisomorphic 2-(9,4,6) designs," *Journal of Combinatorial Mathematics and Combinatorial Computing*, Vol. 37, No. 1, pp. 173–176.

[13] Patric R.J. Östergård and Uri Blass, "On the Size of Optimal Binary Codes of Length 9 and Covering Radius 1," *IEEE Transactions on Information Theory*, Vol. 47, No. 6, pp. 2556–2557. http://ieeexplore.ieee.org/iel5/18/20448/00945268.pdf?isNumber=20448

[14] Patric R.J. Östergård and William D. Weakley, "Values of Domination Numbers of the Queen's Graph," *The Electronic Journal of Combinatorics*, Vol. 8, No. 1, 19 p. http://www.combinatorics.org/Volume_8/Abstracts/v8i1r29.html

5.2 Articles in collections

[15] Jürgen Dix, Ulrich Furbach, and Ilkka Niemelä, "Nonmonotonic Reasoning: towards Efficient Calculi and Implementations," in A. Robinson and A. Voronkov (Eds.), *Handbook of Automated Reasoning, Volume II*, Elsevier Science, Amsterdam, pp. 1241–1354.

5.3 Conference papers

[16] Tuomas Aura, Pekka Nikander, and Jussipekka Leiwo, "DOS-Resistant Authentication with Client Puzzles," in B. Christianson, B. Crispo, J.A. Malcolm, and M. Roe (Eds.), *Security Protocols, 8th International Workshop, Cambridge, UK, April 2000, Revised Papers*, Lecture Notes in Computer Science, Vol. 2133, Springer-Verlag, Berlin, Germany, pp. 170–177. http://link.springer.de/link/service/series/0558/bibs/2133/21330170.htm

[17] Tuomas Aura, "DOS-Resistant Authentication with Client Puzzles (Transcript of Discussion)," in B. Christianson, B. Crispo, J.A. Malcolm, and M. Roe (Eds.), *Security Protocols, 8th International Workshop, Cambridge, UK, April 2000, Revised Papers*, Lecture Notes in Computer Science, Vol. 2133, Springer-Verlag, Berlin, Germany, pp. 178–181. http://link.springer.de/link/service/series/0558/bibs/2133/21330178.htm

[18] Catharina Candolin and Pekka Nikander, "IPv6 Source Addresses Considered Harmful," in H.R. Nielson (Ed.), *Proceedings of Nordsec 2001*, Technical University of Denmark, Department of Informatics and Mathematical Modelling, Technical Report IMM-TR-2001-14, Lyngby, Denmark, November, pp, 54–68.
http://www.tml.hut.fi/%7epnr/publications/nordsec2001.pdf

[19] Javier Esparza and Keijo Heljanko, "Implementing LTL Model Checking with Net Unfoldings," in M.B. Dwyer (Ed.), *Model Checking Software, 8th International SPIN Workshop*, Lecture Notes in Computer Science, Vol. 2057, Springer-Verlag, Berlin, Germany, pp. 37–56. http://link.springer.de/link/service/series/0558/bibs/2057/20570037.htm

[20] Keijo Heljanko, "Bounded Reachability Checking with Process Semantics," in K.G. Larsen and M. Nielsen (Eds.), *CONCUR 2001 — Concurrency Theory*, Lecture Notes in Computer Science, Vol. 2154, Springer-Verlag, Berlin, Germany, pp. 218–232. http://link.springer.de/link/service/series/0558/bibs/2154/21540218.htm

[21] Keijo Heljanko and Ilkka Niemelä, "Answer Set Programming and Bounded Model Checking," in *Answer Set Programming: Towards Efficient and Scalable Knowledge Representation and Reasoning*, AAAI Press, pp. 90–96. http://www.tcs.hut.fi/%7eini/papers/HelNie-ASP2001.ps.gz

[22] Keijo Heljanko and Ilkka Niemelä, "Bounded LTL Model Checking with Stable Models," in T. Eiter, W. Faber, and M. Truszczyński (Eds.), *Logic Programming and Nonmonotonic Reasoning*, Lecture Notes in Artificial Intelligence, Vol. 2173, Springer-Verlag, Berlin, Germany, pp. 200–212.
http://link.springer.de/link/service/series/0558/bibs/2173/21730200.htm

[23] Tomi Janhunen, "On the Effect of Default Negation on the Expressiveness of Disjunctive Rules," in T. Eiter, W. Faber, and M. Truszczyński (Eds.), *Logic Programming and Nonmonotonic Reasoning*, Lecture Notes in Artificial Intelligence, Vol. 2173, Springer-Verlag, Berlin, Germany, pp. 93–106.
http://link.springer.de/link/service/series/0558/bibs/2173/21730093.htm

[24] Timo Latvala, "Model Checking LTL Properties of High-Level Petri Nets with Fairness Constraints," in J.-M. Colom and M. Koutny (Eds.), *Application and Theory of Petri Nets 2001*, Lecture Notes in Computer Science, Vol. 2075, Springer-Verlag, Berlin, Germany, pp. 242–262. http://link.springer.de/link/service/series/0558/bibs/2075/20750242.htm

[25] Marko Mäkelä, "Optimising Enabling Tests and Unfoldings on Algebraic System Nets," in J.-M. Colom and M. Koutny (Eds.), *Application and Theory of Petri Nets 2001*, Lecture Notes in Computer Science, Vol. 2075, Springer-Verlag, Berlin, Germany, pp. 283–302.
http://link.springer.de/link/service/series/0558/bibs/2075/20750283.htm

[26] Leo Ojala, Elina Parviainen, Olli-Matti Penttinen, Harriet Beaver, and Teemu Tynjälä, "Modeling Feynman's Quantum Computer using Stochastic High Level Petri Nets," in *SMC 2001 Conference Proceedings*, IEEE, 6 p.

[27] Leo Ojala, Teemu Tynjälä, and Harriet Beaver, "Modeling Serial Quantum Processors Using Petri Nets," in N.C. Callaos, F.G. Tinetti, J.M. Champarnaud, and J.K. Lee (Eds.), *SCI2001/ISAS2001 Proceedings, Volume XIV, Computer Science and Engineering: Part II*, International Institute of Informatics and Systematics, Orlando FL, USA, pp. 463–468.

[28] Timo Soininen, Ilkka Niemelä, Juha Tiihonen, and Reijo Sulonen, "Representing Configuration Knowledge With Weight Constraint Rules," in *Answer Set Programming: Towards Efficient and Scalable Knowledge Representation and Reasoning*, AAAI Press, pp. 195–201.
http://www.tcs.hut.fi/%7eini/papers/snts-ASP2001.ps.gz

[29] Tommi Syrjänen, "Omega-Restricted Logic Programs," in T. Eiter, W. Faber, and M. Truszczyński (Eds.), *Logic Programming and Nonmonotonic Reasoning*, Lecture Notes in Artificial Intelligence, Vol. 2173, Springer-Verlag, Berlin, Germany, pp. 267–279.
http://link.springer.de/link/service/series/0558/bibs/2173/21730267.htm

[30] Tommi Syrjänen, "Version Spaces and Rule-Based Configuration Management," in T. Soininen (Ed.), *Working Notes of the IJCAI 2001 Workshop on Configuration*, pp. 78–84.
http://www.tcs.hut.fi/%7etssyrjan/publications/syrjanen_ijcai01cw.ps.gz

[31] Tommi Syrjänen and Ilkka Niemelä, "The Smodels System," in T. Eiter, W. Faber, and M. Truszczyński (Eds.), *Logic Programming and Nonmonotonic Reasoning*, Lecture Notes in Artificial Intelligence, Vol. 2173, Springer-Verlag, Berlin, Germany, pp. 434–438.
http://link.springer.de/link/service/series/0558/bibs/2173/21730434.htm

[32] Kimmo Varpaaniemi, "Minimizing the Number of Successor States in the Stubborn Set Method," in L. Czaja (Ed.), *Concurrency, Specification and Programming: Proceedings of the CS&P'2001 Workshop, Warsaw, 3–5 October 2001*, Zakład Graficzny UW, zam. 583/2001, Warsaw, pp. 279–290. http://www.tcs.hut.fi//Publications/papers/kvacsp01.ps.gz

5.4 Reports

[33] Javier Esparza and Keijo Heljanko, *Implementing LTL Model Checking with Net Unfoldings*, Helsinki University of Technology, Laboratory for Theoretical Computer Science, Research Reports HUT-TCS-A68, Espoo, March, 29 p.
http://www.tcs.hut.fi/Publications/reports/A68.pdf

[34] Keijo Heljanko, Victor Khomenko, and Maciej Koutny, *Parallelisation of the Petri Net Unfolding Algorithm*, University of Newcastle upon Tyne, Department of Computing Science, Technical Reports CS-TR-733, Newcastle upon Tyne, UK, June, 14 p. http://www.cs.ncl.ac.uk/research/trs/papers/733.pdf

[35] Petteri Kaski, *Isomorph-Free Exhaustive Generation of Combinatorial Designs* (essentially the same publication as [44]), Helsinki University of Technology, Laboratory for Theoretical Computer Science, Research Reports HUT-TCS-A70, Espoo, December, 125 p.

[36] Eero Lassila, *A Tree Expansion Formalism for Generative String Rewriting*, Helsinki University of Technology, Laboratory for Theoretical Computer Science, Technical Reports HUT-TCS-B20, Espoo, June, 70 p. http://www.tcs.hut.fi/Publications/reports/B20.pdf

[37] Timo Latvala, *Model Checking Linear Temporal Logic Properties of Petri Nets with Fairness Constraints*, Helsinki University of Technology, Laboratory for Theoretical Computer Science, Research Reports HUT-TCS-A67, Espoo, March, 40+iii p. http://www.tcs.hut.fi/Publications/reports/A67.pdf

[38] Helger Lipmaa, *Statistical Zero-Knowledge Proofs from Diophantine Equations*, International Association for Cryptologic Research, Cryptology ePrint Archive 2001/086. http://eprint.iacr.org/2001/086/

[39] Helger Lipmaa, N. Asokan, and Valtteri Niemi, *Secure Vickrey Auctions without Threshold Trust*, International Association for Cryptologic Research, Cryptology ePrint Archive 2001/095. http://eprint.iacr.org/2001/095/

[40] Marko Mäkelä, *A Reachability Analyser for Algebraic System Nets*, Helsinki University of Technology, Laboratory for Theoretical Computer Science, Research Reports HUT-TCS-A69, Espoo, June, 85 p. http://www.tcs.hut.fi/Publications/reports/A69.pdf

[41] Kimmo Varpaaniemi (Ed.), *Annual Report for the Year 2000*, Helsinki University of Technology, Laboratory for Theoretical Computer Science, Annual Reports HUT-TCS-Y2000, Espoo, July, 30 p. http://www.tcs.hut.fi/Publications/reports/y00pdfm.pdf

5.5 Licentiate's theses

[42] Toni Jussila, *Bounded Model Checking for Verifying Concurrent Programs*, 72 p. Accepted by Department of Computer Science and Engineering on December 17. Jussila received the degree of Lic.Sc. (Tech.) on December 17.

5.6 Master's theses

[43] Maarit Hietalahti, *Efficient Key Agreement for Ad-hoc Networks*, iv+52 p. Accepted by Department of Engineering Physics and Mathematics on June 12. Hietalahti received the degree of M.Sc. (Tech.) on June 12. http://www.tcs.hut.fi/%7emhietala/mhietala_mt.ps

[44] Petteri Kaski, *Isomorph-Free Exhaustive Generation of Combinatorial Designs* (essentially the same publication as [35]), 125 p. Accepted by Department of Computer Science and Engineering on September 24. Kaski received the degree of M.Sc. (Tech.) on September 24.

[45] Jari Katajavuori, *An Architecture for Dynamic Software Upgrading Over-the-air in Mobile Systems*, 83 p. Accepted by Department of Computer Science and Engineering on March 19. Katajavuori received the degree of M.Sc. (Tech.) on March 19.

[46] Harri Kuusisto, *Käyttäjän sähköinen tunnistaminen ja yksityisyys (User authentication and privacy)*, 76 p. Accepted by Department of Computer Science and Engineering on December 17. Kuusisto received the degree of M.Sc. (Tech.) on December 17.

[47] Veera Lehtonen, *Implementation of a robust electronic voting system*, 63 p. Accepted by Department of Computer Science and Engineering on April 23. Lehtonen received the degree of M.Sc. (Tech.) on April 23.

[48] Rauni Pääkkönen, *Implementing a formal agent description language*, 62 p. Accepted by Department of Computer Science and Engineering on December 17. Pääkkönen received the degree of M.Sc. (Tech.) on December 17.

[49] Viktor Rosendahl, *A customized very long instruction word processor for Viterbi decoding*, 58 p. Accepted by Department of Electrical and Communications Engineering on June 4. Rosendahl received the degree of M.Sc. (Tech.) on June 4.

[50] Janne O. Salmi, *An adaptive mobility management mechanism for IP networks using ad hoc routing*, 66 p. Accepted by Department of Computer Science and Engineering on May 21. Salmi received the degree of M.Sc. (Tech.) on November 7.

[51] Kjell Sand, *Development of Unit and Module Testing in Embedded Systems*, 65 p. Accepted by Department of Electrical and Communications Engineering on December 17. Sand received the degree of M.Sc. (Tech.) on December 17.

[52] Sami O. Virtanen, *Effective Long Code Generation in WCDMA*, 56 p. Accepted by Department of Electrical and Communications Engineering on October 29. Virtanen received the degree of M.Sc. (Tech.) on October 29.

5.7 Software

[53] Lasse Anderson, Johannes Helander, Keijo Heljanko, Tomi Janhunen, Robert Jürgens, Ismo Kangas, Kari J. Nurmela, Kenneth Oksanen, Olavi Pesonen, Marko Rauhamaa, James Reilly, Heikki Suonsivu, Kimmo Valkealahti, Kimmo Varpaaniemi, and Pauli Väisänen, *`PROD 3.3.09` — an advanced tool for efficient reachability analysis.* http://www.tcs.hut.fi/Software/prod/index.html

[54] Keijo Heljanko, *`punroll 0.3`: a bounded reachability checker.* http://www.tcs.hut.fi/%7ekepa/tools/punroll/

[55] Keijo Heljanko and Patrik Simons, *`boundsmodels 0.9`: a bounded LTL model checker.* http://www.tcs.hut.fi/%7ekepa/tools/boundsmodels/

[56] Keijo Heljanko and Patrik Simons, *`unfsmodels 0.9`: an LTL model checker using net unfoldings.* http://www.tcs.hut.fi/%7ekepa/tools/unfsmodels/

[57] Tommi Junttila, *`BCSat 0.3` — a satisfiability checker for Boolean circuits.* http://www.tcs.hut.fi/%7etjunttil/bcsat/index.html

[58] Marko Mäkelä, *`lsts2dot` — a translator from `TVT` LSTS files to the input file language of `GraphViz`.* http://www.tcs.hut.fi/Software/maria/tools/tvt/lsts2dot.C

[59] Marko Mäkelä, Timo Latvala, and Kimmo Varpaaniemi, *`MARIA 1.0` — a modular reachability analyzer.* http://www.tcs.hut.fi/Software/maria/index.html

[60] Marko Mäkelä and Kimmo Varpaaniemi, *`lsts2pn` — a translator from `TVT` LSTS files to the input file language of `MARIA`.* http://www.tcs.hut.fi/Software/maria/tools/tvt/tvt.html

[61] Mauno Rönkkö, Heikki Tauriainen, and Marko Mäkelä, *`LBT`: LTL to Büchi conversion.* http://www.tcs.hut.fi/Software/maria/tools/lbt/index.html

[62] Tommi Syrjänen, *`lparse-1.0.9`: a local grounder for `Smodels`.* http://www.tcs.hut.fi/Software/smodels/lparse/index.html

[63] Heikki Tauriainen, *`lbtt 1.0.0` — an LTL-to-Büchi translator testbench.* http://www.tcs.hut.fi/%7ehtauriai/lbtt/index.html

HELSINKI UNIVERSITY OF TECHNOLOGY LABORATORY FOR THEORETICAL COMPUTER SCIENCE
ANNUAL REPORT 2001