HELSINKI UNIVERSITY OF TECHNOLOGY
Faculty of Information Technology
Department of Computer Science
DIGITAL SYSTEMS LABORATORY
Annual Report
May 1994

ISBN 951–22–2145–4

ANNUAL REPORT FOR THE YEAR 1993

edited by ILKKA NIEMELÄ

Digital Systems Laboratory
Department of Computer Science
Helsinki University of Technology
Otaniemi, FINLAND

Helsinki University of Technology
Department of Computer Science
Digital Systems Laboratory
Otaniemi, Otakaari 1
FIN–02150 ESPOO, FINLAND

HELSINKI UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE
DIGITAL SYSTEMS LABORATORY
Annual Report
May 1994 ISBN 951–22–2145–4

Annual Report for the Year 1993

edited by ILKKA NIEMELÄ
Senior Assistant

Introduction:

This report describes the educational and research activities of the Digital Systems Laboratory at Helsinki University of Technology during the year 1993.

The educational and research activities of the Chair and of the Laboratory have recently been concentrated upon the areas: specification, modelling, and analysis of parallel and distributed digital systems and theoretical computer science.

Printing: TKK Monistamo; Otaniemi, 1994

Helsinki University of Technology
Department of Computer Science
Digital Systems Laboratory
Otaniemi, Otakaari 1
FIN–02150 ESPOO, FINLAND

Phone: $\frac{90}{+358\text{–}0}$ 4511
Telex: 125 161 htkk sf
Telefax: +358–0–465 077
E-mail: lab@saturn.hut.fi

Contents

1 Personnel

The university resources of the Chair consist of one *professor*, one *senior assistant*, one *teaching assistant*, a *laboratory manager*, and a *secretary*. A heavy load of the teaching tasks is also set to a large number of the students who, at the same time, are participating in the post-graduate education. The need of more permanent personnel resources for management, teaching, and research has been strongly felt during the whole history of the Chair.

1.1 University staff

Ojala, Leo	Prof.	Head of the Laboratory
Ukkonen, Esko	Prof.	Docent
Kangasniemi, Ulla	Mrs.	Laboratory Secretary
Lilius, Johan	Lic.Tech.	Laboratory Manager
Niemelä, Ilkka	Dr.Tech.	Senior Assistant on leave of absence until 31.8.
Tiusanen, Mikko	Ph.D.	Assistant on leave of absence until 30.6. and 11.8.–31.12. Senior Assistant as substitute until 31.5.
Janhunen, Tomi	M.Sc.Eng.	Senior Assistant as substitute 1.6.–31.8.
Husberg, Nisse	Dr.Tech.	Assistant as substitute 11.8.–31.8.
Nurmela, Kari	M.Sc.Eng.	Assistant as substitute 1.9.–31.12.
Varpaaniemi, Kimmo	Lic.Tech.	Assistant as substitute 1.1.–30.6.

1.2 Special lecturers

Heinonen, Rauno	Lic.Tech.	spring and autumn term
Kettunen, Esa	M.Sc.Eng.	spring term
Moslemie, Abbas	Lic.Tech.	spring term
Saksa, Antti	Lic.Tech.	spring term

also:

Janhunen, cf. 1.1, 1.4		spring term
Lilius, cf. 1.1		spring and autumn term
Nurmela, cf. 1.1, 1.4		spring and autumn term
Rintanen, cf. 1.4		spring and autumn term
Tiusanen, cf. 1.1		autumn term
Östergård, cf. 1.4		spring and autumn term

1.3 Part-time teaching assistants

Syvänne, Tuomo	Stud.Tech.	spring term

also:

Pyssysalo, cf. 1.4		autumn term
Janhunen, cf. 1.1, 1.4		autumn term
Nurmela, cf. 1.1, 1.4		spring term
Rautila, cf. 1.4		spring and autumn term
Väisänen, cf. 1.4		spring and autumn term

1.4 Researchers

Grönberg, Peter	Stud.Tech.	Research assistant 1.1.–30.4.
Janhunen, Tomi	M.Sc.Eng.	Researcher 1.1.–30.5., 1.9.–31.12. cf. also 1.1
Kaisla, Tuija	M.Sc.Eng.	Researcher 1.8.–31.10.
Nurmela, Kari	M.Sc.Eng.	Researcher 1.1.–31.8. cf. also 1.1
Pyssysalo, Tino	Stud.Tech.	Research trainee 1.6.–31.8. Research assistant 1.10.–31.12.
Rautila, Mika	Stud.Tech.	Research assistant 1.4.–31.8.
Rissanen, Ahti	Stud.Tech.	Research assistant 1.4.–31.8.
Rintanen, Jussi	Lic.Tech.	Researcher 1.1.–31.12.
Saarhelo, Jari	Stud.Tech.	Research assistant 1.5.–31.12.
Varpaaniemi, Kimmo	Lic.Tech.	Researcher 1.7.–31.12. cf. also 1.1
Väisänen, Pauli	Stud.Tech.	Research assistant 1.1.–31.12.
Östergård, Patric	Dr.Tech.	Researcher 1.1.–31.12.

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.

In the following list, for each course it is given in parentheses: the code according to the Study Programme of the University; the number of study credits (abbr. by *crt*, one crt signifies 40 hours of effective work), abbreviations *spt* and *aut* express the spring term and autumn term during the calendar year, respectively; and the name(s) of the lecturer(s).

2.1 Theoretical computer science

Logic in Computer Science: Basics

(Tik-79.144; 2 crts, aut.; *Rintanen*)
Propositional and predicate calculus; syntax, semantics, and proof theory of these. Applications of logic in computer science, e.g. mechanical theorem proving and logic programming.

Logic in Computer Science: Special Topics

(Tik-79.145; 3 crts, spt; *Janhunen, Lilius, Rintanen*)
The basics of modal and epistemic logics: proof, model theory and decidability results. Temporal logics based on partial order semantics and their application to the verification of distributed systems. Nonmonotonic logics and their applications in computer science.

Introduction to Theoretical Computer Science

(Tik-79.147; 3 crts, spt.; *Heinonen, Moslemie*)
Basics of the theories of automata and formal languages. Computability theory. Principal limitations of computers. Basics of discrete mathematics: abstract algebra, algorithms, graphs, combinatorics. Material designed to cover also applications in computer science.

Introduction to Theoretical Computer Science

(Tik-79.148; 2 crts, aut.; *Heinonen*)
Basics of the theories of automata and formal languages. Computability theory. Principal limitations of computers.

Automata and Formal Languages

(Tik-79.151; 2 crts, spt.; *Heinonen*)
The topic varies each year. The topic in 1993: regular, context free, and context sensitive languages. Fuzzy languages. Corresponding automata. Parsing. Lindenmayer languages and systems.

Algorithms and Computability

(Tik-79.153; 2 crts, aut.; *Heinonen*)
Models of computation. Complexity theory. The principles of construction of algorithms. Classification and reducibility of problems. Decidability.

Formal Description of Computing Systems

(Tik-79.157; 2 crts, aut.; *Lilius*)
Introduction to modern specification and verification methods and tools. Fundamentals of algebraic specification, specification using process algebra, and verification using temporal logics. Use of computer tools (Larch Prover, Concurrency Workbench, PAM, OBJ3).

Cryptography and Data Security

(Tik-79.159; 3 crts, spt.; *Saksa, Östergård*)
Data security. Data encryption and decryption. Classical, contemporary and future cryptosystems. Digital signatures and message authentication.

Student Project in Theoretical Computer Science

(Tik-79.189; 3 crts, spt. & aut.)
An independent study in the field of theoretical computer science.

Special Course on Theoretical Computer Science

(Tik-79.192; 2 crts, aut.; *Ojala*)
The course focuses on some current application of theoretical computer science. The topic varies from year to year. The topic in autumn 1993: the use of temporal logic to the specification and verification of reactive systems.

Seminar on Theoretical Computer Science

(Tik-79.194; 2 crts, spt.; *Ojala*)
Selected topics in computer science: knowledge in distributed digital systems; epistemic logic; applications to communication protocols.

Postgraduate Course on Digital Systems Science

(Tik-79.198, 10 crts, spt. & aut.; *Ojala*)
Topics of the spring term: formal methods in distributed computing: Petri nets, process algebra.

Topics of the autumn term: logic in theoretical computer science: model theory.

2.2 Parallel and distributed systems

Parallel and Distributed Digital Systems

(Tik-79.179; 3 crts, aut.; *Lilius, Tiusanen, Nurmela*)
Introduction to net theory. Net classes, their properties and analysis techniques. Applications in modelling digital systems with nets. Process algebras, specifically CCS, and its application to digital systems. Program specification and verification by using temporal logic.

Parallel and Distributed Computation

(Tik-79.184; 3 crts, aut.; *Östergård*)
Formal specification and verification of parallel systems. Modelling and analysis of communication protocols. Partial analysis of reachability graphs.

Formal Description Techniques for Concurrent Systems

(Tik-79.193; 2 crts, spt.; *Kettunen*)
Specification, validation, testing and analysis methods for large parallel, embedded, and real time systems. Specification languages LOTOS and SDL.

3 Research activities

The particular research topics and projects are described in the following subsections. In the descriptions the members of the research groups are given in parentheses, beginning with the leading or senior researcher.

3.1 Theoretical studies

Petri net research *(Ojala, Husberg, Kaisla, Lilius, Tiusanen, Varpaaniemi)*

- *High level distributed transition systems* (Husberg)

  The main research activity has been about practical application of high level distributed transition systems. They have been compared to predicate/transition nets and coloured nets but also to (sequential) transition systems. The design of an implementation of high level distributed transition systems has started.

- *Fault-tolerant real-time systems* (Kaisla)

  The aim of the research has been to apply formal methods for analysing distributed, fault-tolerant real-time systems. We have concentrated on a clock synchronization protocol which is an example of a simple fault-tolerant agreement protocol in a real-time system. The main emphasis has been on proving the correctness of the protocol. We want to show that the clock system achieves agreement (the largest distance between any two correct clocks at any time is bounded) and maintains the required accuracy (the deviation of a correct clock from real-time is bounded). We have considered two real-time formalisms: timed automata, and a method based on extended Hoare triples. With timed automata of finite size it turned out to be difficult to model a protocol where the timing of actions is based not only on time but also, e.g., on the number of messages received. Extended Hoare triples were found to be suitable for modelling and proving the problem, although in both methods the modelling of the behaviour of a faulty process posed an interesting problem.

- *The analysis of algebraic high level nets* (Lilius).

  The notion of folding from a Petri net to an algebraic high level net has been formalised. It has been shown how for each Petri net N an algebraic high level net $Fold(N)$ can be constructed such that the unfolding of $Fold(N)$ is isomorphic to N. This construction was then shown to be functorial, establishing a functor from the category of Petri nets to the category of algebraic high level nets.

  The semantics of Petri nets has been studied within the "Objects are sheaves" paradigm. Transitions and places are represented as sheaves and nets are represented as diagrams of sheaves. Both an interleaving semantics, and a non-interleaving semantics have been shown to arise as the limit of the sheaf diagram representing the net.

- *Static analysis of Ada tasking programs* (Tiusanen)

  The research on static analysis of Ada tasking programs is interested in methods to discover errors in Ada programs caused by the use of parallel processing—tasking—constructs. Here the Ada tasking program is first abstracted to a finite state model of, basically, the control flow of its tasks. The abstraction consists of ignoring the variables of the program to a varying degree. Most of the approaches use a form of *state space generation*, i.e. reachability analysis, as a method of analysis.

  The research done has combined Petri net based analysis techniques for reduced effort reachability graph generation, namely the *symbolic* or set-at-a-time generation using e.g. Binary-Decision Diagrams (BDDs) by Clarke et al., the *symmetry* method by Jensen et al. that only generates a part of the reachability graph when symmetric parts are guaranteed to exist, and the *stubborn set* method by Valmari that tries to avoid generating states irrelevant for the decision of properties of interest, e.g. deadlock-freedom. This work is related to the work done on the analysis software PROD by Varpaaniemi. Limiting the extent to which the variables are ignored is another interesting future direction for which the use of high level Petri nets can serve as a good vehicle, since the variables are easily represented in such a model.

- *Efficient detection of deadlocks in Petri nets* (Varpaaniemi)

  Reachability analysis is a powerful formal method for analysis of concurrent and distributed finite state systems. It suffers from the *state space explosion problem*, however: the state space of a system can be far too large to be completely generated. The study considers two promising methods, Valmari's *stubborn set method* and Godefroid's *sleep set method* to avoid generating all of the state space when searching for *undesirable reachable terminal states*, also called *deadlocks*. What makes deadlocks especially interesting is the fact that the verification of a *safety property* can often be reduced to deadlock detection. The considered methods utilize the *independence of transitions* to cut down on the number of states inspected during the search. These methods have been *combined* by Godefroid, Pirottin, and Wolper to further reduce the number of inspected states.

  It has been shown that the stubborn set method and the sleep set method can be combined without any of the assumptions previously placed on the stubborn sets as far as the detection of reachable terminal states in *place/transition nets*, a class of Petri nets, is concerned. This result is established by showing a more general result which gives a sufficient condition for a method to be compatible with the sleep set method in the detection of reachable terminal states in place/transition nets.

  The number of enabled transitions in a stubborn set can drastically affect the number of states inspected by the stubborn set method during the search for reachable terminal states. Some heuristics for relieving the problem have been proposed. The value of *dynamically stubborn sets* is emphasized as a useful generalization of stubborn sets and some results are presented that improve the understanding of the stubborn set method.

Applications of logic in computer science and engineering *(Niemelä, Janhunen, Rintanen, Väisänen)*

The research has concentrated on the theoretical foundations of knowledge representation and it has been focused on *nonmonotonic reasoning*, one of the key areas in knowledge representation. *Autoepistemic logic*, a leading formalization of nonmonotonic reasoning, is used as the basis of the research. Autoepistemic logic has been studied as a unified framework for nonmonotonic reasoning and, in particular, as a basis for automating nonmonotonic reasoning. The notions of priorities and weakened negative introspection in autoepistemic reasoning have been investigated intensively. A study of the use of the logic of knowledge in distributed systems has been initiated.

- *A unified framework for nonmonotonic reasoning* (Niemelä)

  The problem of automating nonmonotonic reasoning has been investigated. Several forms of nonmonotonic reasoning have been proposed in the literature. However, despite of different starting points and underlying intuitions, leading approaches to nonmonotonic reasoning have turned out to be closely related. Our aim is to exploit the close connections so that a single theorem prover could be used as a unified reasoning tool for a wide range of forms of nonmonotonic reasoning. An interesting approach is to choose some formalization of nonmonotonic reasoning as a basic system and develop a theorem prover for it. Decision procedures for other forms of nonmonotonic reasoning can then be realized by reducing decision problems in a given formalization to decision problems in the basic system.

  We have considered the possibility of using a recently developed variant of autoepistemic logic based on *L-hierarchic expansions* as the basic system. It has already been shown how other leading formalizations of nonmonotonic reasoning such as default logic, circumscription, logic programs, truth maintenance systems, and forms of abduction can be understood as special cases of the new variant of autoepistemic logic. Hence the new variant of autoepistemic logic can be seen as a unified basis for a large range of forms of nonmonotonic reasoning which are important in knowledge representation.

  We have developed a novel decision method for autoepistemic reasoning which is applicable in a general setting, i.e. for an autoepistemic logic defined on top of a given classical logic. The aim has been to devise a conceptually clear method where sources of complexity and targets for further optimization are clearly identifiable. Two orthogonal sources can be identified: classical reasoning and conflict resolution. The classical reasoning task can be handled using a separate theorem prover. The decision method imposes no additional requirements leaving abundant room for optimizations of the theorem prover. The conflict resolution task is solved by employing a simple backtracking scheme. The method exploits efficiently the monotonicity of the underlying consequence relation to reduce choice points in backtracking. Jussi Rintanen has developed a prototype implementation of the novel decision method in

Standard ML.

As an example of the optimization possibilities we have developed a new complete decision method for default logic where the search space of applying default rules is pruned efficiently by using ideas from autoepistemic reasoning. The method provides an interesting approach to implementing truth maintenance systems as well as stable model semantics and answer set semantics of logic programs.

- *Weakened negative introspection* (Niemelä, Janhunen)

  In autoepistemic reasoning, making a default assumption amounts to performing a negative introspection step, i.e. choosing to disbelieve a formula. The research has been concentrated on a variant of autoepistemic logic called cautious autoepistemic logic which is based on a novel principle for negative introspection. The properties of cautious autoepistemic logic have been investigated, e.g. the major reasoning tasks of cautious autoepistemic logic have been classified with respect to the polynomial time hierarchy (PH). The negative introspection of cautious autoepistemic logic corresponds to a complete problem on the second level of PH. General logic programs, i.e. logic programs with negation as failure, have also been analysed using the framework. The semantics of a general logic program can be defined in terms of cautious autoepistemic logic. Most importantly, the semantics covers the class of general logic programs as a whole and it is more accurate than the widely known well-founded semantics.

- *Priorities in Nonmonotonic Reasoning* (Rintanen)

  A major problem in representing knowledge is posed by the possibility of default rules that conflict. If the application of both of two default rules leads to a contradiction, they cannot both be applied. In such cases a reasoner may either remain indifferent or prioritize one rule over the other.

  To investigate reasoning with conflicting defaults, a version of autoepistemic logic with a priority mechanism has been defined. Priorities are formalized as partial orders on an arbitrary set of formulae. A subset of all possible sets of conclusions (stable expansions), the *preferred* stable expansions, is determined by the priorities. The priority mechanism presented is suitable for resolving conflicts, for example, in inheritance hierarchies and diagnostic reasoning.

  Although the priority notion is declarative, it has nevertheless a computational interpretation that naturally leads to useful procedures for automated theorem-proving. In the research this has been demonstrated by giving a family of decision procedures for prioritized autoepistemic logic and proving them correct.

The properties of prioritized autoepistemic logic and its relation to other formalizations of nonmonotonic reasoning with priorities have been investigated. Prioritized autoepistemic logic generalizes preferred subtheories of Brewka and ordered theory presentations of Ryan. The decision problems of propositional prioritized autoepistemic logic seem to be one level higher in the polynomial hierarchy than the respective decision problems of propositional autoepistemic logic without priorities. Contrary to initial research hypotheses, priorities lead to tractable decision problems only in few cases.

Automated theorem-proving with nonmonotonic logics has been investigated both with priorities and in the general context. An automatic theorem-proving system has been implemented in Standard ML. The system contains theorem-provers for Reiter's default logic, Moore's autoepistemic logic, Niemelä's L-hierarchic enumeration-based autoepistemic logic, and for the prioritized autoepistemic logic defined as part of the investigation concerning priorities.

- *Modal logic S5 specifications* (Väisänen)

  The logic of knowledge has been argued to be a useful tool in the field of modelling, analysing and verifying distributed systems. A distributed system can be modelled by giving a set of logical sentences which characterize the system (specification) and properties of the system can be verified by determining whether sentences characterizing the properties are logical consequences of the specification.

  The logical consequence relation of the modal logic S5 is an object of a special interest because the role of S5 has been emphasized when modelling distributed systems. There are two main ways to determine if a given S5 sentence is a logical consequence of a set of sentences. Either a direct theorem proving method for S5 is used or S5 sentences are first translated into classical predicate logic and theorem proving methods for predicate logic are employed. We have investigated a very simple translation scheme which exploits the properties of the standard equality relation of the predicate logic. The translation scheme has been applied together with the tableau and the resolution method to different formalizations of some well known problems and it has been found to be useful.

  Frequently in the literature the semantics of S5 sentences (especially in the context of different variants of the three wise men puzzle) has been visualized by a special kind of a graph. The nodes of the graph are all the possible truth assignments to the primitive propositions. The edges of the graph describe the knowledge of an agent. These graphs have inspired us to study methods visualizing multimodal S5 sentences and the formal correspondence between the visualization and the Kripke models.

Construction methods in coding and design theory *(Östergård, Nurmela)*

The aim of the research is to develop methods for constructing covering codes and covering designs. Particular stress has been laid on computational methods. Among these, simulated annealing and tabu search, two methods applicable to combinatorial optimization problems, have played a central role. In the research the performance of these and other combinatorial optimization methods has also been evaluated.

These optimization methods can as such successfully be used to construct good small covering codes and covering designs. As for covering codes, research has also been done on constructing large codes; the two most important methods work as follows: old codes are partitioned and combined to obtain a new code, and a new code is constructed by taking a union of cosets of a linear code. Various other combinatorial and algebraic methods have also been developed. These methods have been used to construct many record-breaking covering codes and covering designs. These can be used as football pools systems and lottery systems, respectively. Preliminary studies have been carried out on constructions of other combinatorial structures than those of the aforementioned types.

3.2 Technical projects

PROD 2.0 *(Varpaaniemi)*

The PROD 2.0 project is a part of the ESPRIT project EP6067 CALIBAN (Causal Calculi Based on Nets) and its aim is to extend and enhance the predicate/transition net reachability analysis tool PROD developed at the Digital System Laboratory.

Two reductive reachability analysis methods, the *symmetry method* and the so called *CFFD-preserving stubborn set method*, have been implemented in PROD. In addition, an interconnection between PROD and the ARA tool of the Technical Research Centre of Finland (VTT) has been built. ARA is a reachability analysis tool for LOTOS. The interconnection is such that ARA gives PROD a set of LTSs (labelled transition systems) together with a set of actions to be hidden, and PROD computes a parallel composition of the LTSs and hides the requested actions.

A new algorithm for computing the place permutations corresponding to symmetries has been derived from the algorithm of Schmidt and a new algorithm for computing stubborn sets for the CFFD-preserving stubborn set method has been derived from the algorithms of Valmari. The new algorithms save time and space in the symmetry method and avoid inspecting redundant states in the CFFD-preserving stubborn set method.

PRED — a Petri net editor and graphical user interface to PROD *(Nurmela, Väisänen)*

A graphical Petri net editor and user interface to PROD has been implemented. With this software package, called PRED, you can draw, edit and print high level predicate/transition nets in a graphical form. You can automatically compile the net drawing and generate the reachable states from the initial state using PROD. You can also visualize the results of the reachability analysis, i.e. show markings of places in various states. PRED has been implemented by using the InterViews C++ class library and more specifically, the Unidraw part of it. The PRED User's Manual and the System Manual are available.

Analysis of a high speed data bus with predicate/transitions nets *(Pyssysalo)*

The Telecommunications Laboratory of Technical Research Centre of Finland started a research program in 1990 to develop an efficient and cost-effective interconnection network for high speed multiprocessor applications. During the research program a high throughput bus concept, named Frame Synchronized Ring (FSR-bus), was introduced.

In this research the medium access control algorithm (MAC protocol) of the FSR-bus has been analyzed with PROD which is a predicate/transition net reachability analysis tool developed at the Digital Systems Laboratory. With PROD it has been proven that the MAC protocol cannot deadlock and it is fair, i.e., if a node of the bus wants to send data, it gets access to the bus within a finite queuing time. The effectiveness of the methods for reducing the size of the reachability graph implemented in PROD (the stubborn set method and the symmetry method) has been studied. The results show that with these methods the state space of a reachability graph is greatly reduced but their application can be slow. The performance of the FSR-bus in terms of maximal queuing times of the nodes has been analyzed.

Process algebraic methods in specifying and analysing intelligent network services *(Rautila)*

In the study the functioning of a telecommunication network relating to an intelligence network service is examined as observable behaviour of the network. The behaviour of a service is described as a process algebraic agent. This agent can be analysed through its state space in several ways. For example, it can be determined if the agent's behaviour includes deadlock situations. Also the temporal properties of the agent can be analysed through its state space. By analysing the agent carefully enough one can ensure that it describes precisely what is required of the service. Such an unambiguous description of the behaviour of the service can then be used in the verification of a service implementation. This is done by first making the process algebraic models of the service logic program and the intelligent network platform in hand. If the composition of these two agents is equivalent to the service description, then implementation satisfies the service description. Many typical intelligent network services are described as process algebraic agents and their ETSI platform implementations are verified.

A new process algebraic language is developed which is suitable for describing intelligent network services. The semantics of the new algebraic language is defined using CCS. This makes it possible to analyse services described in the new language using already existing CCS-analysers. This possibility is utilized by analysing the services described employing "The Edinburgh Concurrency Workbench" (cwb).

Design and implementation of a simulation tool for intelligent networks *(Rissanen)*

The aim of the work has been to implement a part of the international recommendation for intelligent networks (ITU-T Recommendations Q.1201, Q.1202, Q.1203, Q.1204, Q.1205, Q.1211, Q.1213, Q.1214, Q.1215, Q.1290). Additional material has also been used from a recommendation for the Finnish national intelligent network architecture. C-based Virtual Operating System (CVOPS) has been employed as a major development environment. CVOPS was chosen because it provides a flexible and efficient environment for implementing and testing communication protocols. This work has been focused on implementing the service switching point that controls the detection of phone calls which use intelligent networks. During the project some new protocols have been developed because the corresponding parts of the recommendations are left open at this time. Further developments of the simulation tool have been outlined.

4 Examinations and theses

4.1 Doctor of Technology

Niemelä, Ilkka:	Autoepistemic Logic as a Unified Basis for Nonmonotonic Reasoning
Tiusanen, Mikko:	Static Analysis of Ada Tasking Programs: Models and Algorithms

	(Doctor of Philosophy in Electrical Engineering and Computer Science, University of Illinois at Chicago, USA. Tiusanen studied at the University of Illinois at Chicago as a graduate student from September 1990 until June 1992. Then he returned to Finland and carried on his research at the Digital Systems Laboratory. He defended his thesis in Chicago in June 1993. The supervisor of the thesis was Prof. Tadao Murata.)
Östergård, Patric:	Construction Methods for Covering Codes

4.2 Licentiate in Technology

Rintanen, Jussi:	Priorities and Nonmonotonic Reasoning
Sandqvist, Sam:	On Finding Optimal Potential Customers from a Large Marketing Database — a Genetic Algorithm Approach
Varpaaniemi, Kimmo:	Efficient Detection of Deadlocks in Petri Nets

4.3 Master of Science in Engineering

Janhunen, Tomi:	Weakened Negative Introspection in Autoepistemic Reasoning
Nurmela, Kari:	Constructing Combinatorial Designs by Local Search
Paananen, Jorma:	Implementing a Database Interface in Two Graphical User Interfaces (in Finnish)
Patinen, Juha-Pekka:	An Interactive Theorem Prover for Linear Logic
Pesonen, Olavi:	Factory Model in Computer Aided Process Planning (in Finnish)

5 Awards

Jussi Rintanen was granted the annual Pro Gradu award of the Finnish Society for Computer Science for a distinguished master's thesis in Computer Science.

6 Conferences, tours, and visits

6.1 Conferences and tours

Ojala, Leo	14th International Conference on Application and Theory of Petri Nets, 21.–25.6., Chicago, Illinois, USA.
	4th International Conference on Concurrency Theory, 30.8.–2.9., Hildesheim, Germany.
	International Workshop on Concurrency, Specification, and Programming, 14.–16.10., Nieborów, Poland.
	5th International Workshop on Petri Nets and Performance Models, 19.–22.10., Toulouse, France.
Husberg, Nisse	4th International Conference on Concurrency Theory, 30.8.–2.9., Hildesheim, Germany.
Janhunen, Tomi	3rd Kurt Gödel Colloquium, Computational Logic and Proof Theory, 23.–27.8, Brno, Czech Republic (presentation: A Scheme for Weakened Negative Introspection in Autoepistemic Reasoning, with Niemelä).
Lilius, Johan	3rd International Conference on Algebraic Methodology and Software Technology, 22.–25.6., Enschede, The Netherlands.
	Annual General Meeting of the ESPRIT Basic Research Working Group CALIBAN, 27.–28.9., Fischbachau, Bavaria, Germany.
Kaisla, Tuija	5th International Conference on Computer-Aided Verification, 28.6.–1.7., Elounda, Crete, Greece.
Niemelä, Ilkka	3rd Kurt Gödel Colloquium, Computational Logic and Proof Theory, 23.–27.8, Brno, Czech Republic (presentation: A Scheme for Weakened Negative Introspection in Autoepistemic Reasoning, with Janhunen).
	AAAI Fall Symposium Series — Automated Deduction in Nonstandard Logics, 22.–24.10.1994 Raleigh, NC, USA (presentation: A Decision Method for Nonmonotonic Reasoning Based on Autoepistemic Logic).

University of Kentucky, Department of Computer Science, 25.–27.10., Lexington, KY, USA (presentation: A Decision Method for Nonmonotonic Reasoning Based on Autoepistemic Logic).

Nurmela, Kari
Southeastern International Conference on Combinatorics, Graph Theory and Computing, 22.-26.2., Boca Raton, Florida (presentation: Upper Bounds for Covering Designs by Simulated Annealing, with Östergård).

Rintanen, Jussi
The autumn meeting of the Finnish Society for Computer Science, 17.11., Espoo, Finland (presentation: Stratification and Tractability in Nonmonotonic Reasoning).

Tiusanen, Mikko
14th International Conference on Application and Theory of Petri Nets, 21.–25.6., Chicago, Illinois, USA.

International Workshop on Concurrency, Specification, and Programming, 14.–16.10., Nieborów, Poland (presentation: Symbolic Stubborn Set Search).

Varpaaniemi, Kimmo
13th International IFIP WG 6.1 Symposium on Protocol Specification, Testing, and Verification, 25.–28.5., Liège, Belgium.

14th International Conference on Application and Theory of Petri Nets, 21.–25.6., Chicago, Illinois, USA (presentation: PROD — a Reachability Analysis Tool).

5th International Conference on Computer-Aided Verification, 28.6.–1.7., Elounda, Crete, Greece.

Annual General Meeting of the ESPRIT Basic Research Working Group CALIBAN, 27.–28.9., Fischbachau, Bavaria, Germany. (presentation: On Implementing the Symmetry Method).

International Workshop on Concurrency, Specification, and Programming, 14.–16.10., Nieborów, Poland (presentation: Dynamically Stubborn Sets and the Sleep Set Method).

5th International Workshop on Petri Nets and Performance Models, 19.–22.10., Toulouse, France.

Östergård, Patric	24th Southeastern International Conference on Combinatorics, Graph Theory and Computing, 22.–26.2., Boca Raton, Florida, USA (presentation: Upper Bounds for Covering Designs by Simulated Annealing, with Nurmela).
	14th British Combinatorial Conference, 5.–9.7., Keele, United Kingdom, (presentation: New Constructions for q-ary Covering Codes).
	1st French-Israeli Workshop in Algebraic Coding, 19.–21.7., Paris, France (presentation: A Coloring Problem in Hamming Spaces).
	5th Nordic Conference in Combinatorics, 29.8–1.9., Halvorsbøle, Norway (presentation: On Construction Methods for Covering Codes).

6.2 Research visits

Niemelä, Ilkka	SRI International, Artificial Intelligence Center, Menlo Park, CA, USA, 1.2.–31.7.1993.
Tiusanen, Mikko	University of Illinois at Chicago, Chicago, Illinois, USA, 15.5.–6.7.1993.

6.3 Visiting scholars

Gerhard Brewka, Dr., Geschellschaft für Mathematik und Datenverarbeitung, Sankt Augustin, Germany, 16.–18.9. (gave a lecture titled "Adding Priorities and Specificity to Default Logic".)

Gérard Cohen, Prof., Département Informatique, Ecole Nationale Supérieure des Télécommunications, Paris, France, 8.–11.9., (gave a lecture titled "A Few Applications of Coding Theory").

7 Documents

7.1 Journal articles

Kari Nurmela and *Patric Östergård*: Upper Bounds for Covering Designs by Simulated Annealing, Congressus Numerantium, 96 (1993), 93–111.

7.2 Conference papers

Nisse Husberg: Composition and Synchronization in High Level Distributed Transition Systems; Hans-Dieter Burkhardt, Peter Starke, Ludwig Czaja

(eds.): Workshop Concurrency, Specification & Programming, November, 1992, Informatik-Preprint 22, Humboldt Universität zu Berlin, 1993; pp 55–67.

Nisse Husberg: On the Relations Between Petri Nets and Transition Systems: Composition and Synchronization; Proceedings of the Fourth Nordic Workshop on Program Correctness, Bergen, Norway, November 1992, University of Bergen, Reports in Informatics No. 78, 1993, pp. 189–198.

Tomi Janhunen and *Ilkka Niemelä*: A Scheme for Weakened Negative Introspection in Autoepistemic Reasoning; Proceedings of the Third Kurt Gödel Colloquium, Brno, Czech Republic, August 1993, Lecture Notes in Computer Science 713, Springer-Verlag, 1993; pp. 211-222.

Johan Lilius: On Invariants and Substitution Systems; Hans-Dieter Burkhardt, Peter Starke, Ludwig Czaja (eds.): Workshop Concurrency, Specification & Programming, November, 1992, Informatik-Preprint 22, Humboldt Universität zu Berlin, 1993; pp. 78–85.

Ilkka Niemelä: A Decision Method for Nonmonotonic Reasoning Based on Autoepistemic Logic; Automated Deduction in Nonstandard Logics — Papers from the 1993 Fall Symposium, Raleigh, NC, USA, October, 1993; pp. 97–104.

Mikko Tiusanen: On Models of Static Analysis of Ada Tasking Programs; Hans-Dieter Burkhardt, Peter Starke, Ludwig Czaja (eds.): Workshop Concurrency, Specification & Programming, November, 1992, Informatik-Preprint 22, Humboldt Universität zu Berlin, 1993; pp. 147–162.

Kimmo Varpaaniemi: On Choosing a Scapegoat in the Stubborn Set Method; Hans-Dieter Burkhardt, Peter Starke, Ludwig Czaja (eds.): Workshop Concurrency, Specification & Programming, November, 1992, Informatik-Preprint 22, Humboldt Universität zu Berlin, 1993; pp. 163–171.

7.3 Laboratory reports

Mikko Tiusanen (Ed.): Annual Report for the Year 1992, May 1993, 17 p.

Series A: Research Reports

No. 21 *Tomi Janhunen*: Weakened Negative Introspection in Autoepistemic Reasoning, February 1993, 82 p.

No. 22 *Patric R. J. Östergård* and *Heikki O. Hämäläinen*: New Upper Bounds for Binary/Ternary Mixed Covering Codes, March 1993, 33 p.

No. 23 *Johan Lilius*: A Sheaf Semantics for Petri Nets, June 1993, 13 p.

No. 24 *Ilkka Niemelä*: Autoepistemic Logic as a Unified Basis for Nonmonotonic Reasoning, August 1993, 162 p.

No. 25 *Patric R. J. Östergård*: Construction Methods for Covering Codes, September 1993, 25 p.

No. 26 *Kimmo Varpaaniemi*: Efficient Detection of Deadlocks in Petri Nets, October 1993, 56 p.

No. 27 *Kari J. Nurmela*: Constructing Combinatorial Designs by Local Search, November 1993, 76 p.

No. 28 *Jussi Rintanen*: Priorities and Nonmonotonic Reasoning, December 1993, 90 p.

Series B: Technical Reports

No. 10 *Kari J. Nurmela* and *Patric R. J. Östergård*: Constructing Covering Designs by Simulated Annealing, January 1993, 25 p.

No. 11 *Peter Grönberg*, *Mikko Tiusanen*, and *Kimmo Varpaaniemi*: PROD—A Pr/T-Net Reachability Analysis Tool, June 1993, 44 p.

7.4 Other publications

Ilkka Niemelä: Logiikka tietämyskielenä (Logic as a Knowledge Representation Language; in Finnish), E. Hyvönen and I. Karanta and M. Syrjänen (toim.): *Tekoälyn ensyklopedia*, Gaudeamus, 1993; pp. 116–124.