BRICS Research Series, Abstracts, 1997

April 29, 2003

This document is also available as *PostScript* and *DVI*.

References

[RS-97-53] *PostScript*, *PDF*, *DVI*.
Olivier Danvy.
Online Type-Directed Partial Evaluation.
December 1997.
31 pp. Extended version of an article appearing in *Third Fuji International Symposium on Functional and Logic Programming*, FLOPS '98 Proceedings (Kyoto, Japan, April 2–4, 1998), pages 271–295, World Scientific, 1998.

Abstract: In this experimental work, we extend type-directed partial evaluation (a.k.a. "reduction-free normalization" and "normalization by evaluation") to make it online, by enriching it with primitive operations (delta-rules). Each call to a primitive operator is either unfolded or residualized, depending on the operands and either with a default policy or with a user-supplied filter. The user can also specify how to residualize an operation, by pattern-matching over the operands. Operators may be pure or have a computational effect.

We report a complete implementation of online type-directed partial evaluation in Scheme, extending our earlier offline implementation. Our partial evaluator is native in that it runs compiled code instead of using the usual meta-level technique of symbolic evaluation.

[RS-97-52] *PostScript*, *PDF*, *DVI*.
Paola Quaglia.
On the Finitary Characterization of π-Congruences.
December 1997.
59 pp.

Abstract: Some alternative characterizations of late full congruences, either strong or weak, are presented. Those congruences are classically defined by requiring the corresponding ground bisimilarity under all name substitutions.

We first improve on those infinitary definitions by showing that congruences can be alternatively characterized in the π-calculus by sticking to a finite number of carefully identified substitutions,

and hence, by resorting to only a finite number of ground bisimilarity checks.

Then we investigate the same issue in both the ground and the non-ground $\pi\xi$-calculus, a CCS-like process algebra whose ground version has already been proved to coincide with ground π-calculus. The $\pi\xi$-calculus perspective allows processes to be explicitly interpreted as functions of their free names. As a result, a couple of alternative characterizations of π-congruences are given, each of them in terms of the bisimilarity of one single pair of $\pi\xi$-processes. In one case, we exploit λ-closures of processes, so inducing the effective generation of the substitutions necessary to infer non-ground equivalence. In the other case, a more promising call-by-need discipline for the generation of the wanted substitutions is used. This last strategy is also adopted to show a coincidence result with open semantics.

By minor changes, all of the above characterizations for late semantics can be suited for congruences of the early family.

[RS-97-51] *PostScript*, *PDF*, *DVI*.
James McKinna and Robert Pollack.
Some Lambda Calculus and Type Theory Formalized.
December 1997.
43 pp Appears in *Journal of Automated Reasoning* 23(3–4):373–409 (1999).

Abstract: In previous papers we have used the LEGO proof development system to formalize and check a substantial body of knowledge about lambda calculus and type theory. In the present paper we survey this work, including some new proofs, and point out what we feel has been learned about the general issues of formalizing mathematics. On the technical side, we describe an abstract, and simplified, proof of standardization for beta reduction, not previously published, that does not mention redex positions or residuals. On the general issues, we emphasize the search for formal definitions that are convenient for formal proof and convincingly represent the intended informal concepts.

[RS-97-50] *PostScript*, *PDF*, *DVI*.
Ivan B. Damgrd and Birgit Pfitzmann.
Sequential Iteration of Interactive Arguments and an Efficient Zero-Knowledge Argument for NP.
December 1997.
19 pp. Appears in Larsen, Skyum and Winskel, editors, *25th International Colloquium on Automata, Languages, and Programming*, ICALP '98 Proceedings, LNCS 1443, 1998, pages 772–783.

Abstract: We study the behavior of interactive arguments under sequential iteration, in particular how this affects the error probability. This problem turns out to be more complex than one might expect from the fact that for interactive proofs, the error trivially decreases exponentially in the number of iterations.

In particular, we study the typical efficient case where the iter-

ated protocol is based on a single instance of a computational problem. This is not a special case of independent iterations of an entire protocol, and real exponential decrease of the error cannot be expected, but nevertheless, for practical applications, one needs concrete relations between the complexity and error probability of the underlying problem and that of the iterated protocol. We show how this problem can be formalized and solved using the theory of proofs of knowledge.

We also prove that in the non-uniform model of complexity the error probability of independent iterations of an argument does indeed decrease exponentially – to our knowledge this is the first result about a strictly exponentially small error probability in a computational cryptographic security property. As an illustration of our first result, we present a very efficient zero-knowledge argument for circuit satisfiability, and thus for any NP problem, based on any collision-intractable hash function. Our theory applies to show the soundness of this protocol. Using an efficient hash function such as SHA-1, the protocol can handle about 20000 binary gates per second at an error level of 2^{-50}.

[RS-97-49] *PostScript*, *PDF*, *DVI*.
Peter D. Mosses.
CASL for ASF+SDF Users.
December 1997.
22 pp. Appears in Sellink, Editor, *2nd International Workshop on the Theory and Practice of Algebraic Specifications, Electronic Workshops in Computing*, ASF+SDF '97 Proceedings, Springer-Verlag, 1997.

Abstract: CASL is an expressive language for the algebraic specification of software requirements, design, and architecture. It has been developed by an open collaborative effort called CoFI (Common Framework Initiative for algebraic specification and development). CASL combines the best features of many previous algebraic specification languages, and it is hoped that it may provide a focus for future research and development in the use of algebraic techniques, as well being attractive for industrial use.

This paper presents CASL for users of the ASF+SDF framework. It shows how familiar constructs of ASF+SDF may be written in CASL, and considers some problems that may arise when translating specifications from ASF+SDF to CASL. It then explains and motivates various CASL constructs that cannot be expressed directly in ASF+SDF. Finally, it discusses the rôle that the ASF+SDF system might play in connection with tool support for CASL.

Comments: http://www.springer.co.uk/ewic/workshops/ASFSDF97.

[RS-97-48] *PostScript*, *PDF*, *DVI*.
Peter D. Mosses.
CoFI: The Common Framework Initiative for Algebraic Specification and Development.
December 1997.

24 pp. Appears in Bidoit and Dauchet, editors, *Theory and Practice of Software Development: 7th International Joint Conference CAAP/FASE*, TAPSOFT '97 Proceedings, LNCS 1214, 1997, pages 115–137.

Abstract: An open collaborative effort has been initiated: to design a common framework for algebraic specification and development of software. The rationale behind this initiative is that the lack of such a common framework greatly hinders the dissemination and application of research results in algebraic specification. In particular, the proliferation of specification languages, some differing in only quite minor ways from each other, is a considerable obstacle for the use of algebraic methods in industrial contexts, making it difficult to exploit standard examples, case studies and training material. A common framework with widespread acceptance throughout the research community is urgently needed.

The aim is to base the common framework as much as possible on a critical selection of features that have already been explored in various contexts. The common framework will provide a family of specification languages at different levels: a central, reasonably expressive language, called CASL, for specifying (requirements, design, and architecture of) conventional software; restrictions of CASL to simpler languages, for use primarily in connection with prototyping and verification tools; and extensions of CASL, oriented towards particular programming paradigms, such as reactive systems and object-based systems. It should also be possible to embed many existing algebraic specification languages in members of the CASL family.

A tentative design for CASL has already been proposed. Task groups are studying its formal semantics, tool support, methodology, and other aspects, in preparation for the finalization of the design.

[RS-97-47] *PostScript*, *PDF*.
Anders B. Sandholm and Michael I. Schwartzbach.
Distributed Safety Controllers for Web Services.
December 1997.
20 pp. Appears in Astesiano, editor, *Fundamental Approaches to Software Engineering: First International Conference*, FASE '98 Proceedings, LNCS 1382, 1998, pages 270–284.

Abstract: We show how to use high-level synchronization constraints, written in a version of monadic second-order logic on finite strings, to synthesize safety controllers for interactive web services. We improve on the nave runtime model to avoid state-space explosions and to increase the flow capacities of services.

[RS-97-46] *PostScript*, *PDF*, *DVI*.
Olivier Danvy and Kristoffer H. Rose.
Higher-Order Rewriting and Partial Evaluation.
December 1997.
20 pp. Extended version of paper appearing in Nipkow, editor, *Rewriting Techniques and Applications: 9th International Con-*

ference, RTA '98 Proceedings, LNCS 1379, 1998, pages 286–301.

Abstract: We demonstrate the usefulness of higher-order rewriting techniques for specializing programs, i.e., for partial evaluation. More precisely, we demonstrate how casting program specializers as combinatory reduction systems (CRSs) makes it possible to formalize the corresponding program transformations as metareductions, i.e., reductions in the internal "substitution calculus." For partial-evaluation problems, this means that instead of having to prove on a case-by-case basis that one's "two-level functions" operate properly, one can concisely formalize them as a combinatory reduction system and obtain as a corollary that static reduction does not go wrong and yields a well-formed residual program.

We have found that the CRS substitution calculus provides an adequate expressive power to formalize partial evaluation: it provides sufficient termination strength while avoiding the need for additional restrictions such as types that would complicate the description unnecessarily (for our purpose). We also review the benefits and penalties entailed by more expressive higher-order formalisms.

In addition, partial evaluation provides a number of examples of higher-order rewriting where being higher order is a central (rather than an occasional or merely exotic) property. We illustrate this by demonstrating how standard but non-trivial partial-evaluation examples are handled with higher-order rewriting.

[RS-97-45] *PostScript*, *PDF*, *DVI.*
Uwe Nestmann.
What Is a 'Good' Encoding of Guarded Choice?
December 1997.
28 pp. Revised and slightly extended version of a paper published in Parrow and Palamidessi, editors, *4th Workshop on Expressiveness in Concurrency*, EXPRESS '97 Proceedings, ENTCS 7, 1997. Revised version appears in *Information and Computation* 156:287–319 (2000).

Abstract: The π-calculus with synchronous output and mixed-guarded choices is strictly more expressive than the π-calculus with asynchronous output and no choice. As a corollary, Palamidessi recently proved that there is no fully compositional encoding from the former into the latter that preserves divergence-freedom and symmetries. This paper shows that there are nevertheless 'good' encodings between these calculi.

In detail, we present a series of encodings for languages with (1) input-guarded choice, (2) both input- and output-guarded choice, and (3) mixed-guarded choice, and investigate them with respect to compositionality and divergence-freedom. The first and second encoding satisfy all of the above criteria, but various 'good' candidates for the third encoding—inspired by an existing distributed implementation—invalidate one or the other criterion. While essentially confirming Palamidessi's result, our

study suggests that the combination of strong compositionality and divergence-freedom is too strong for more practical purposes.

[RS-97-44] *PostScript*, *PDF*, *DVI.*
Gudmund Skovbjerg Frandsen.
On the Density of Normal Bases in Finite Fields.
December 1997.
14 pp. Appears in *Finite Fields and Their Applications*, 6(1):23–38, 2000.

Abstract: Let $\mathbf{F}_{q^n}$ denote the finite field with q^n elements, for q being a prime power. $\mathbf{F}_{q^n}$ may be regarded as an n-dimensional vector space over $\mathbf{F}_q$. $\alpha \in \mathbf{F}_{q^n}$ generates a *normal* basis for this vector space ($\mathbf{F}_{q^n} : \mathbf{F}_q$), if $\{\alpha, \alpha^q, \alpha^{q^2}, \ldots, \alpha^{q^{n-1}}\}$ are linearly independent over $\mathbf{F}_q$. Let $N(q,n)$ denote the number of elements in $\mathbf{F}_{q^n}$ that generate a normal basis for $\mathbf{F}_{q^n} : \mathbf{F}_q$, and let $\nu(q,n) = \frac{N(q,n)}{q^n}$ denote the frequency of such elements.

We show that there exists a constant $c > 0$ such that

$$\nu(q,n) \;\geq\; c\frac{1}{\sqrt{\lceil \log_q n \rceil}}, \quad \text{for all } n, q \geq 2$$

and this is optimal up to a constant factor in that we show

$$0.28477 \;\leq\; \lim_{n\to\infty} \inf\, \nu(q,n)\sqrt{\log_q n} \;\leq\; 0.62521, \qquad \text{for all } q \geq 2$$

We also obtain an explicit lower bound:

$$\nu(q,n) \;\geq\; \frac{1}{e\lceil \log_q n \rceil}, \quad \text{for all } n, q \geq 2$$

.

[RS-97-43] *PostScript*, *PDF*, *DVI.*
Vincent Balat and Olivier Danvy.
Strong Normalization by Type-Directed Partial Evaluation and Run-Time Code Generation (Preliminary Version).
December 1997.
16 pp. Appears in Leroy and Ohori, editors, *Types in Compilation, 2nd International Workshop*, TIC '98 Proceedings, LNCS 1473, 1998, pages 240–252.

Abstract: We investigate the synergy between type-directed partial evaluation and run-time code generation for the Caml dialect of ML. Type-directed partial evaluation maps simply typed, closed Caml values to a representation of their long beta-eta-normal form. Caml uses a virtual machine and has the capability to load byte code at run time. Representing the long beta-eta-normal forms as byte code gives us the ability to strongly normalize higher-order values (i.e., weak head normal forms in ML), to compile the resulting strong normal forms into byte code, and to load this byte code all in one go, at run time.

We conclude this note with a preview of our current work on scaling up strong normalization by run-time code generation to the Caml module language.

[RS-97-42] *PostScript*, *PDF*, *DVI*.
Ulrich Kohlenbach.
On the No-Counterexample Interpretation.
December 1997.
26 pp. Appears in *Journal of Symbolic Logic* 64(4):1491–1511, 1999.

Abstract: In [Kreisel 51],[Kreisel 52] G. Kreisel introduced the no-counterexample interpretation (n.c.i.) of Peano arithmetic. In particular he proved, using a complicated ε-substitution method (due to W. Ackermann), that for every theorem A (A prenex) of first-order Peano arithmetic **PA** one can find ordinal recursive functionals $\underline{\Phi}_A$ of order type $< \varepsilon_0$ which realize the Herbrand normal form A^H of A.

Subsequently more perspicuous proofs of this fact via functional interpretation (combined with normalization) and cut-elimination where found. These proofs however do not carry out the n.c.i. as a **local** proof interpretation and don't respect the modus ponens on the level of the n.c.i. of formulas A and $A \to B$. Closely related to this phenomenon is the fact that both proofs do not establish the condition (δ) and – at least not constructively – (γ) which are part of the definition of an 'interpretation of a formal system' as formulated in [Kreisel 51].

In this paper we determine the complexity of the n.c.i. of the modus ponens rule for

(i) **PA**-provable sentences,

(ii) for arbitrary sentences $A, B \in \mathcal{L}(\mathbf{PA})$ **uniformly** in functionals satisfying the n.c.i. of (prenex normal forms of) A and $A \to B$, and

(iii) for arbitrary $A, B \in \mathcal{L}(\mathbf{PA})$ **pointwise** in given $\alpha(< \varepsilon_0)$-recursive functionals satisfying the n.c.i. of A and $A \to B$.

This yields in particular perspicuous proofs of new uniform versions of the conditions $(\gamma), (\delta)$.

Finally we discuss a variant of the concept of an interpretation presented in [Kreisel 58] and show that it is incomparable with the concept studied in [Kreisel 51],[Kreisel 52]. In particular we show that the n.c.i. of $\mathbf{PA}_n$ by $\alpha(< \omega_n(\omega))$-recursive functionals ($n \geq 1$) is an interpretation in the sense of [Kreisel 58] but not in the sense of [Kreisel 51] since it violates the condition (δ).

[RS-97-41] *PostScript*, *PDF*, *DVI*.
Jon G. Riecke and Anders B. Sandholm.
A Relational Account of Call-by-Value Sequentiality.
December 1997.
24 pp. Appears in *Twelfth Annual IEEE Symposium on Logic in Computer Science*, LICS '97 Proceedings, 1997, pages 258–267. This report is superseded by the later report BRICS RS-99-10.

Abstract: We construct a model for FPC, a purely functional, sequential, call-by-value language. The model is built from partial continuous functions, in the style of Plotkin, further constrained

to be uniform with respect to a class of logical relations. We prove that the model is fully abstract.

[RS-97-40] *PostScript*, *PDF*, *DVI.*
Harry Buhrman, Richard Cleve, and Wim van Dam.
Quantum Entanglement and Communication Complexity.
December 1997.
14 pp.

Abstract: We consider a variation of the multi-party communication complexity scenario where the parties are supplied with an extra resource: particles in an entangled quantum state. We show that, although a prior quantum entanglement cannot be used to simulate a communication channel, it can reduce the communication complexity of functions in some cases. Specifically, we show that, for a particular function among three parties (each of which possesses part of the function's input), a prior quantum entanglement enables them to learn the value of the function with only three bits of communication occurring among the parties, whereas, without quantum entanglement, four bits of communication are necessary. We also show that, for a particular two-party probabilistic communication complexity problem, quantum entanglement results in less communication than is required with only classical random correlations (instead of quantum entanglement). These results are a noteworthy contrast to the well-known fact that quantum entanglement cannot be used to actually simulate communication among remote parties.

[RS-97-39] *PostScript*, *PDF*, *DVI.*
Ian Stark.
Names, Equations, Relations: Practical Ways to Reason about 'new'.
December 1997.
ii+33 pp. Appears in *Fundamenta Informaticae* 33(4):369–396. This supersedes the earlier BRICS Report RS-96-31, and also expands on the paper presented in Groote and Hindley, editors, *Typed Lambda Calculi and Applications: 3rd International Conference*, TLCA '97 Proceedings, LNCS 1210, 1997, pages 336–353.

Abstract: The nu-calculus of Pitts and Stark is a typed lambda-calculus, extended with state in the form of dynamically-generated names. These names can be created locally, passed around, and compared with one another. Through the interaction between names and functions, the language can capture notions of scope, visibility and sharing. Originally motivated by the study of references in Standard ML, the nu-calculus has connections to local declarations in general; to the mobile processes of the pi-calculus; and to security protocols in the spi-calculus.

This paper introduces a logic of equations and relations which allows one to reason about expressions of the nu-calculus: this uses a simple representation of the private and public scope of names, and allows straightforward proofs of contextual equivalence (also known as observational, or observable, equivalence). The logic is based on earlier operational techniques, providing the same power but in a much more accessible form. In particular it allows intuitive and direct proofs of all contextual equivalences between first-order functions with local names.

[RS-97-38] *PostScript*, *PDF*, *DVI*.
Michał Hańćkowiak, Michał Karoński, and Alessandro Panconesi.
On the Distributed Complexity of Computing Maximal Matchings.
December 1997.
16 pp. Appears in *The Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '98 Proceedings, 1998, pages 219–225.
Abstract: We show that maximal matchings can be computed deterministically in polylogarithmically many rounds in the synchronous, message-passing model of computation. This is one of the very few cases known of a non-trivial graph structure which can be computed distributively in polylogarithmic time without recourse to randomization.

[RS-97-37] *PostScript*, *PDF*, *DVI*.
David A. Grable and Alessandro Panconesi.
Fast Distributed Algorithms for Brooks-Vizing Colourings (Extended Abstract).
December 1997.
20 pp. Appears in *The Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '98 Proceedings, 1998, pages 473–480. Journal version in *Journal of Algorithms*, 37(1):85–120, 2000.
Abstract: We give extremely simple and fast randomized ditributed algorithms for computing vertex colourings in triangle-free graphs which use many fewer than Δ colours, where Δ denotes the maximum degree of the input network.

[RS-97-36] *PostScript*, *PDF*, *DVI*.
Thomas Troels Hildebrandt, Prakash Panangaden, and Glynn Winskel.
Relational Semantics of Non-Deterministic Dataflow.
December 1997.
21 pp.
Abstract: We recast dataflow in a modern categorical light using profunctors as a generalization of relations. The well known causal anomalies associated with relational semantics of indeterminate dataflow are avoided, but still we preserve much of the intuitions of a relational model. The development fits with the view of categories of models for concurrency and the general treatment of bisimulation they provide. In particular it fits with the recent categorical formulation of feedback using traced monoidal categories. The payoffs are: (1) explicit relations to existing models and semantics, especially the usual axioms of monotone IO automata are read off from the definition of profunctors, (2) a new definition of bisimulation for dataflow, the proof of the congruence of which benefits from the preservation properties associated with open maps and (3) a treatment of

higher-order dataflow as a biproduct, essentially by following the geometry of interaction programme.

[RS-97-35] *PostScript*, *PDF*, *DVI*.
Gian Luca Cattani, Marcelo P. Fiore, and Glynn Winskel.
A Theory of Recursive Domains with Applications to Concurrency.
December 1997.
ii+23 pp. A revised version appears in *Thirteenth Annual IEEE Symposium on Logic in Computer Science*, LICS '98 Proceedings, 1998, pages 214–225.
Abstract: We develop a 2-categorical theory for recursively defined domains. In particular, we generalise the traditional approach based on order-theoretic structures to category-theoretic ones. A motivation for this development is the need of a domain theory for concurrency, with an account of bisimulation. Indeed, the leading examples throughout the paper are provided by recursively defined presheaf models for concurrent process calculi. Further, we use the framework to study (open-map) bisimulation.

[RS-97-34] *PostScript*, *PDF*, *DVI*.
Gian Luca Cattani, Ian Stark, and Glynn Winskel.
Presheaf Models for the π-Calculus.
December 1997.
ii+27 pp. Appears in Moggi and Rosolini, editors, *Category Theory and Computer Science: 7th International Conference*, CTCS '97 Proceedings, LNCS 1290, 1997, pages 106–126.
Abstract: Recent work has shown that presheaf categories provide a general model of concurrency, with an inbuilt notion of bisimulation based on open maps. Here it is shown how this approach can also handle systems where the language of actions may change dynamically as a process evolves. The example is the π-calculus, a calculus for 'mobile processes' whose communication topology varies as channels are created and discarded. A denotational semantics is described for the π-calculus within an indexed category of profunctors; the model is fully abstract for bisimilarity, in the sense that bisimulation in the model, obtained from open maps, coincides with the usual bisimulation obtained from the operational semantics of the π-calculus. While attention is concentrated on the 'late' semantics of the π-calculus, it is indicated how the 'early' and other variants can also be captured.

[RS-97-33] *PostScript*, *PDF*, *DVI*.
Anders Kock and Gonzalo E. Reyes.
A Note on Frame Distributions.
December 1997.
15 pp. Appears in *Cahiers de Topologie et Gometrie Diffrentielle Catgoriques*, 40(2):127–140, 1999.
Abstract: In the context of constructive locale or frame theory (locale theory over a fixed base locale), we study some aspects of 'frame distributions', meaning sup preserving maps from a frame to the base frame. We derive a relationship between results of Jibladze-Johnstone and Bunge-Funk, and also descriptions in

distribution terms, as well as in double negation terms, of the 'interior of closure' operator on open parts of a locale.

[RS-97-32] *PostScript*, *PDF*, *DVI*.
Thore Husfeldt and Theis Rauhe.
Hardness Results for Dynamic Problems by Extensions of Fredman and Saks' Chronogram Method.
November 1997.
i+13 pp. Appears in Larsen, Skyum and Winskel, editors, *25th International Colloquium on Automata, Languages, and Programming*, ICALP '98 Proceedings, LNCS 1443, 1998, pages 67–78.
Abstract: We introduce new models for dynamic computation based on the cell probe model of Fredman and Yao. We give these models access to nondeterministic queries or the right answer 1 as an oracle. We prove that for the dynamic partial sum problem, these new powers do not help, the problem retains its lower bound of $\Omega(\log n/\log\log n)$.

From these results we easily derive a large number of lower bounds of order $\Omega(\log n/\log\log n)$ for conventional dynamic models like the random access machine. We prove lower bounds for dynamic algorithms for reachability in directed graphs, planarity testing, planar point location, incremental parsing, fundamental data structure problems like maintaining the majority of the prefixes of a string of bits and range queries. We characterise the complexity of maintaining the value of any symmetric function on the prefixes of a bit string.

[RS-97-31] *PostScript*, *PDF*.
Klaus Havelund, Arne Skou, Kim G. Larsen, and Kristian Lund.
Formal Modeling and Analysis of an Audio/Video Protocol: An Industrial Case Study Using UPPAAL.
November 1997.
23 pp. Appears in *The 18th IEEE Real-Time Systems Symposium*, RTSS '97 Proceedings, 1997, pages 2–13.
Abstract: A formal and automatic verification of a real-life protocol is presented. The protocol, about 2800 lines of assembler code, has been used in products from the audio/video company Bang & Olufsen throughout more than a decade, and its purpose is to control the transmission of messages between audio/video components over a single bus. Such communications may collide, and one essential purpose of the protocol is to detect such collisions. The functioning is highly dependent on real-time considerations. Though the protocol was known to be faulty in that messages were lost occasionally, the protocol was too complicated in order for Bang & Olufsen to locate the bug using normal testing. However, using the real-time verification tool UPPAAL, an error trace was automatically generated, which caused the detection of "the error" in the implementation. The error was corrected and the correction was automatically proven correct, again using UPPAAL. A future, and more automated, version of the protocol, where this error is fatal, will incorporate the correction. Hence,

this work is an elegant demonstration of how model checking has had an impact on practical software development. The effort of modeling this protocol has in addition generated a number of suggestions for enriching the UPPAAL language. Hence, it's also an excellent example of the reverse impact.

[RS-97-30] *PostScript*, *PDF*, *DVI*.
Ulrich Kohlenbach.
Proof Theory and Computational Analysis.
November 1997.
38 pp. Slightly revised version appeared in Edalat, Jung, Keimel and Kwiatkowska, editors, *Third Workshop on Computation and Approximation*, COMPROX III Selected Papers, ENTCS 13, 1998, 34 pp.

Abstract: In this survey paper we start with a discussion how functionals of finite type can be used for the proof-theoretic extraction of numerical data (e.g. effective uniform bounds and rates of convergence) from non-constructive proofs in numerical analysis.

We focus on the case where the extractability of polynomial bounds is guaranteed. This leads to the concept of hereditarily polynomial bounded analysis **PBA**. We indicate the mathematical range of **PBA** which turns out to be surprisingly large.

Finally we discuss the relationship between **PBA** and so-called feasible analysis **FA**. It turns out that both frameworks are incomparable. We argue in favor of the thesis that **PBA** offers the more useful approach for the purpose of extracting mathematically interesting bounds from proofs.

In a sequel of appendices to this paper we indicate the expressive power of **PBA**.

[RS-97-29] *PostScript*, *PDF*.
Luca Aceto, Augusto Burgueo, and Kim G. Larsen.
Model Checking via Reachability Testing for Timed Automata.
November 1997.
29 pp. Appears in Steffen, editor, *Tools and Algorithms for the Construction and Analysis of Systems: 4th International Conference*, TACAS '98 Proceedings, LNCS 1384, 1998, pages 263–280.

Abstract: In this paper we develop an approach to model-checking for timed automata via reachability testing. As our specification formalism, we consider a dense-time logic with clocks. This logic may be used to express safety and bounded liveness properties of real-time systems. We show how to automatically synthesize, for every logical formula φ, a so-called *test automaton* T_φ in such a way that checking whether a system S satisfies the property φ can be reduced to a reachability question over the system obtained by making T_φ interact with S.

The testable logic we consider is both of practical and theoretical interest. On the practical side, we have used the logic, and the associated approach to model-checking via reachability testing it supports, in the specification and verification in UP-

PAAL of a collision avoidance protocol. On the theoretical side, we show that the logic is powerful enough to permit the definition of *characteristic properties*, with respect to a timed version of the ready simulation preorder, for nodes of deterministic, τ-free timed automata. This allows one to compute behavioural relations via our model-checking technique, therefore effectively reducing the problem of checking the existence of a behavioural relation among states of a timed automaton to a reachability problem.

[RS-97-28] *PostScript*, *PDF*, *DVI.*
Ronald Cramer, Ivan B. Damgrd, and Ueli Maurer.
Span Programs and General Secure Multi-Party Computation.
November 1997.
27 pp.

Abstract: The contributions of this paper are three-fold. First, as an abstraction of previously proposed cryptographic protocols we propose two cryptographic primitives: homomorphic shared commitments and linear secret sharing schemes with an additional multiplication property. We describe new constructions for general secure multi-party computation protocols, both in the cryptographic and the information-theoretic (or secure channels) setting, based on any realizations of these primitives.

Second, span programs, a model of computation introduced by Karchmer and Wigderson, are used as the basis for constructing new linear secret sharing schemes, from which the two above-mentioned primitives as well as a novel verifiable secret sharing scheme can efficiently be realized.

Third, note that linear secret sharing schemes can have arbitrary (as opposed to threshold) access structures. If used in our construction, this yields multi-party protocols secure against general sets of active adversaries, as long as in the cryptographic (information-theoretic) model no two (no three) of these potentially misbehaving player sets cover the full player set. This is a strict generalization of the threshold-type adversaries and results previously considered in the literature. While this result is new for the cryptographic model, the result for the information-theoretic model was previously proved by Hirt and Maurer. However, in addition to providing an independent proof, our protocols are not recursive and have the potential of being more efficient.

[RS-97-27] *PostScript*, *PDF*, *DVI.*
Ronald Cramer and Ivan B. Damgrd.
Zero-Knowledge Proofs for Finite Field Arithmetic or: Can Zero-Knowledge be for Free?
November 1997.
33 pp Appears inKrawczyk, editor, *Advances in Cryptology: 18th Annual International Cryptology Conference*, CRYPTO '98 Proceedings, LNCS 1462, 1998pp. 424–441.

Abstract: We present zero-knowledge proofs and arguments for arithmetic circuits over finite prime fields, namely given a cir-

cuit, show in zero-knowledge that inputs can be selected leading to a given output. For a field $GF(q)$, where q is an n-bit prime, a circuit of size $O(n)$, and error probability 2^{-n}, our protocols require communication of $O(n^2)$ bits. This is the same worst-cast complexity as the trivial (non zero-knowledge) interactive proof where the prover just reveals the input values. If the circuit involves n multiplications, the best previously known methods would in general require communication of $\Omega(n^3 \log n)$ bits.

Variations of the technique behind these protocols lead to other interesting applications. We first look at the Boolean Circuit Satisfiability problem and give zero-knowledge proofs and arguments for a circuit of size n and error probability 2^{-n} in which there is an interactive preprocessing phase requiring communication of $O(n^2)$ bits. In this phase, the statement to be proved later need not be known. Later the prover can *non-interactively* prove any circuit he wants, i.e. by sending only one message, of size $O(n)$ bits.

As a second application, we show that Shamirs (Shens) interactive proof system for the (IP-complete) QBF problem can be transformed to a zero-knowledge proof system with the same asymptotic communication complexity and number of rounds.

The security of our protocols can be based on any one-way group homomorphism with a particular set of properties. We give examples of special assumptions sufficient for this, including: the RSA assumption, hardness of discrete log in a prime order group, and polynomial security of Diffie-Hellman encryption.

We note that the constants involved in our asymptotic complexities are small enough for our protocols to be practical with realistic choices of parameters.

[RS-97-26] *PostScript*, *PDF*, *DVI*.
Luca Aceto and Anna Inglfsdttir.
A Characterization of Finitary Bisimulation.
October 1997.
9 pp. Appears in *Information Processing Letters* 64(3):127–134, November 1997.

Abstract: In this study, we offer a behavioural characterization of the finitary bisimulation for arbitrary transition systems. This result may be seen as the behavioural counterpart of Abramsky's logical characterization theorem. Moreover, for the important class of sort-finite transition systems we present a sharpened version of a behavioural characterization result first proven by Abramsky.

[RS-97-25] *PostScript*, *PDF*.
David A. Mix Barrington, Chi-Jen Lu, Peter Bro Miltersen, and Sven Skyum.
Searching Constant Width Mazes Captures the AC^0 Hierarchy.
September 1997.
20 pp. Appears in Morvan, Meinel and Krob, editors, *15th An-*

nual Symposium on Theoretical Aspects of Computer Science, STACS '98 Proceedings, LNCS 1373, 1998, pages 73–83.

Abstract: We show that searching a width k maze is complete for Π_k, i.e., for the k'th level of the AC^0 hierarchy. Equivalently, st-connectivity for width k grid graphs is complete for Π_k. As an application, we show that there is a data structure solving dynamic st-connectivity for constant width grid graphs with time bound $O(\log\log n)$ per operation on a random access machine. The dynamic algorithm is derived from the parallel one in an indirect way using algebraic tools.

[RS-97-24] *PostScript*, *PDF*, *DVI*.
Sren B. Lassen.
Relational Reasoning about Contexts.
September 1997.
45 pp. Appears as a chapter in the book *Higher Order Operational Techniques in Semantics*, pages 91–135, Gordon and Pitts, editors, Cambridge University Press, 1998.

Abstract: The syntactic nature of operational reasoning requires techniques to deal with term contexts, especially for reasoning about recursion. In this paper we study applicative bisimulation and a variant of Sands' improvement theory for a small call-by-value functional language. We explore an indirect, relational approach for reasoning about contexts. It is inspired by Howe's precise method for proving congruence of simulation orderings and by Pitts' extension thereof for proving applicative bisimulation up to context. We illustrate this approach with proofs of the unwinding theorem and syntactic continuity and, more importantly, we establish analogues of Sangiorgi's bisimulation up to context for applicative bisimulation and for improvement. Using these powerful bisimulation up to context techniques, we give concise operational proofs of recursion induction, the improvement theorem, and syntactic minimal invariance. Previous operational proofs of these results involve complex, explicit reasoning about contexts.

[RS-97-23] *PostScript*, *PDF*, *DVI*.
Ulrich Kohlenbach.
On the Arithmetical Content of Restricted Forms of Comprehension, Choice and General Uniform Boundedness.
August 1997.
35 pp. Slightly revised version appeared in *Annals of Pure and Applied Logic*, vol.95, pp. 257-285, 1998.

Abstract: In this paper the numerical strength of fragments of arithmetical comprehension, choice and general uniform boundedness is studied systematically. These principles are investigated relative to base systems $\mathcal{T}^\omega_n$ in all finite types which are suited to formalize substantial parts of analysis but nevertheless have provably recursive function(al)s of low growth. We reduce the use of instances of these principles in $\mathcal{T}^\omega_n$-proofs of a large class of formulas to the use of instances of certain arithmetical principles thereby determining faithfully the arithmetical content of the former. This is achieved using the method of elimination of Skolem functions for monotone formulas which was introduced by the author in a previous paper.

As corollaries we obtain new conservation results for fragments of analysis over fragments of arithmetic which strengthen known purely first-order conservation results.

[RS-97-22] *PostScript*, *PDF*, *DVI*.
Carsten Butz.
Syntax and Semantics of the Logic $\mathcal{L}^{\lambda}_{\omega\omega}$.
July 1997.
14 pp. Appears in *Notre Dame of Journal Formal Logic*, 38(3):374–384, 1997.

Abstract: In this paper we study the logic $\mathcal{L}^{\lambda}_{\omega\omega}$, which is first order logic extended by quantification over functions (but not over relations). We give the syntax of the logic, as well as the semantics in Heyting categories with exponentials. Embedding the generic model of a theory into a Grothendieck topos yields completeness of $\mathcal{L}^{\lambda}_{\omega\omega}$ with respect to models in Grothendieck toposes, which can be sharpened to completeness with respect to Heyting valued models. The logic $\mathcal{L}^{\lambda}_{\omega\omega}$ is the strongest for which Heyting valued completeness is known. Finally, we relate the logic to locally connected geometric morphisms between toposes.

[RS-97-21] *PostScript*, *PDF*, *DVI*.
Steve Awodey and Carsten Butz.
Topological Completeness for Higher-Order Logic.
July 1997.
19 pp. Appears in *Journal of Symbolic Logic* 65(3):1168–1182 (2000).

Abstract: Using recent results in topos theory, two systems of higher-order logic are shown to be complete with respect to sheaf models over topological spaces, so-called 'topological semantics'. The first is classical higher-order logic, with relational quantification of finitely high type; the second system is a predicative fragment thereof with quantification over functions between types, but not over arbitrary relations. The second theorem applies to intuitionistic as well as classical logic.

[RS-97-20] *PostScript*, *PDF*, *DVI*.
Carsten Butz and Peter T. Johnstone.
Classifying Toposes for First Order Theories.
July 1997.
34 pp. Appears in *Annals of Pure and Applied Logic*, 91(1):33-58, January 1998.

Abstract: By a classifying topos for a first-order theory $\mathbf{T}$, we mean a topos $\mathcal{E}$ such that, for any topos $\mathcal{F}$, models of $\mathbf{T}$ in $\mathcal{F}$ correspond exactly to open geometric morphisms $\mathcal{F} \to \mathcal{E}$. We show that not every (infinitary) first-order theory has a classifying topos in this sense, but we characterize those which do by an appropriate 'smallness condition', and we show that every Grothendieck topos arises as the classifying topos of such a

theory. We also show that every first-order theory has a conservative extension to one which possesses a classifying topos, and we obtain a Heyting-valued completeness theorem for infinitary first-order logic.

[RS-97-19] Andrew D. Gordon, Paul D. Hankin, and Sren B. Lassen.
Compilation and Equivalence of Imperative Objects.
July 1997.
iv+64 pp. This report is superseded by the later report BRICS RS-98-12. Appears also as Technical Report 429, University of Cambridge Computer Laboratory, June 1997. Appears in Ramesh and Sivakumar, editors, *Foundations of Software Technology and Theoretical Computer Science: 17th Conference*, FST&TCS '97 Proceedings, LNCS 1346, 1997, pages 74–87.
Abstract: We adopt the untyped imperative object calculus of Abadi and Cardelli as a minimal setting in which to study problems of compilation and program equivalence that arise when compiling object-oriented languages. We present both a big-step and a small-step substitution-based operational semantics for the calculus and prove them equivalent to the closure-based operational semantics given by Abadi and Cardelli. Our first result is a direct proof of the correctness of compilation to a stack-based abstract machine via a small-step decompilation algorithm. Our second result is that contextual equivalence of objects coincides with a form of Mason and Talcott's CIU equivalence; the latter provides a tractable means of establishing operational equivalences. Finally, we prove correct an algorithm, used in our prototype compiler, for statically resolving method offsets. This is the first study of correctness of an object-oriented abstract machine, and of operational equivalence for the imperative object calculus.

[RS-97-18] *PostScript*, *PDF*, *DVI.*
Robert Pollack.
How to Believe a Machine-Checked Proof.
July 1997.
18 pp. Appears as a chapter in the book *Twenty Five Years of Constructive Type Theory*, eds. Smith and Sambin, Oxford University Press, 1998.
Abstract: Suppose I say "Here is a machine-checked proof of Fermat's last theorem (FLT)". How can you use my putative machine-checked proof as evidence for belief in FLT? This paper presents a technological approach for reducing the problem of believing a formal proof to the same psychological and philosophical issues as believing a conventional proof in a mathematics journal. I split the question of belief into two parts; asking whether the claimed proof is actually a derivation in the claimed formal system, and then whether what it proves is the claimed theorem. The first question is addressed by independent checking of formal proofs. I discuss how this approach extends the informal notion of proof as surveyable by human readers, and how surveyability of proofs reappears as the issue of feasibility of proof-checking.

The second question has no technical answer; I discuss how it appears in verification of computer systems.

[RS-97-17] *PostScript*, *PDF*, *DVI.*
Peter Bro Miltersen.
Error Correcting Codes, Perfect Hashing Circuits, and Deterministic Dynamic Dictionaries.
June 1997.
19 pp. Appears in *The Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '98 Proceedings, 1998, pages 556–563.

Abstract: We consider dictionaries of size n over the finite universe $U = \{0, 1\}^w$ and introduce a new technique for their implementation: error correcting codes. The use of such codes makes it possible to replace the use of strong forms of hashing, such as universal hashing, with much weaker forms, such as clustering.

We use our approach to construct, for any $\epsilon > 0$, a deterministic solution to the dynamic dictionary problem using linear space, with worst case time $O(n^\epsilon)$ for insertions and deletions, and worst case time $O(1)$ for lookups. This is the first deterministic solution to the dynamic dictionary problem with linear space, constant query time, and non-trivial update time. In particular, we get a solution to the static dictionary problem with $O(n)$ space, worst case query time $O(1)$, and deterministic initialization time $O(n^{1+\epsilon})$. The best previous deterministic initialization time for such dictionaries, due to Andersson, is $O(n^{2+\epsilon})$. The model of computation for these bounds is a unit cost RAM with word size w (i.e. matching the universe), and a standard instruction set. The constants in the big-O's are independent upon w. The solutions are weakly non-uniform in w, i.e. the code of the algorithm contains word sized constants, depending on w, which must be computed at compile-time, rather than at run-time, for the stated run-time bounds to hold.

An ingredient of our proofs, which may be interesting in its own right, is the following observation: A good error correcting code for a bit vector fitting into a word can be computed in $O(1)$ time on a RAM with unit cost multiplication.

As another application of our technique in a different model of computation, we give a new construction of perfect hashing circuits, improving a construction by Goldreich and Wigderson. In particular, we show that for any set $S \subseteq \{0, 1\}^w$ of size n, there is a Boolean circuit C of size $O(w \log w)$ with w inputs and $2 \log n$ outputs so that the function defined by C is 1-1 on S. The best previous bound on the size of such a circuit was $O(w \log w \log \log w)$.

[RS-97-16] *PostScript*, *PDF*, *DVI.*
Noga Alon, Martin Dietzfelbinger, Peter Bro Miltersen, Erez Petrank, and Gbor Tardos.
Linear Hashing.

June 1997.

22 pp. Appears in *The Journal of the Association for Computing Machinery*, 46(5):667-683. A preliminary version appeared with the title *Is Linear Hashing Good?* in *The Twenty-ninth Annual ACM Symposium on Theory of Computing*, STOC '97 Proceedings, 1997, pages 465–474.

Abstract: Consider the set $\mathcal{H}$ of all linear (or affine) transformations between two vector spaces over a finite field F. We study how good $\mathcal{H}$ is as a class of hash functions, namely we consider hashing a set S of size n into a range having the same cardinality n by a randomly chosen function from $\mathcal{H}$ and look at the expected size of the largest hash bucket. $\mathcal{H}$ is a universal class of hash functions for any finite field, but with respect to our measure different fields behave differently.

If the finite field F has n elements then there is a bad set $S \subset F^2$ of size n with expected maximal bucket size $\Omega(n^{1/3})$. If n is a perfect square then there is even a bad set with largest bucket size *always* at least $\sqrt{n}$. (This is worst possible, since with respect to a universal class of hash functions every set of size n has expected largest bucket size below $\sqrt{n} + 1/2$.)

If, however, we consider the field of two elements then we get much better bounds. The best previously known upper bound on the expected size of the largest bucket for this class was $O(2^{\sqrt{\log n}})$. We reduce this upper bound to $O(\log n \log\log n)$. Note that this is not far from the guarantee for a random function. There, the average largest bucket would be $\Theta(\log n / \log\log n)$.

In the course of our proof we develop a tool which may be of independent interest. Suppose we have a subset S of a vector space D over $\mathbf{Z}_2$, and consider a random linear mapping of D to a smaller vector space R. If the cardinality of S is larger than $c_\epsilon |R| \log |R|$ then with probability $1 - \epsilon$, the image of S will cover all elements in the range.

[RS-97-15] *PostScript*, *PDF*, *DVI*.

Pierre-Louis Curien, Gordon Plotkin, and Glynn Winskel.

Bistructures, Bidomains and Linear Logic.

June 1997.

41 pp. Appears in Plotkin, Stirling and Tofte, editors, *Proof, Language, and Interaction: Essays in Honour of Robin Milner*, MIT Press, 2000.

Abstract: Bistructures are a generalisation of event structures which allow a representation of spaces of functions at higher types in an order-extensional setting. The partial order of causal dependency is replaced by two orders, one associated with input and the other with output in the behaviour of functions. Bistructures form a categorical model of Girard's classical linear logic in which the involution of linear logic is modelled, roughly speaking, by a reversal of the roles of input and output. The comonad of the model has an associated co-Kleisli category which is closely

related to that of Berry's bidomains (both have equivalent nontrivial full sub-cartesian closed categories).

[RS-97-14] Arne Andersson, Peter Bro Miltersen, Sren Riis, and Mikkel Thorup.
Dictionaries on AC^0 RAMs: Query Time $\Theta(\sqrt{\log n/\log\log n})$ is Necessary and Sufficient.
June 1997.
18 pp. Appears in *37th Annual Symposium on Foundations of Computer Science*, FOCS '96 Proceedings, 1996, pages 441–450 with the title *Static Dictionaries on AC^0 RAMs: Query Time $\Theta(\sqrt{(}\log n/\log\log n))$ is Necessary and Sufficient.*

[RS-97-13] *PostScript*, *PDF*, *DVI.*
Jrgen H. Andersen and Kim G. Larsen.
Compositional Safety Logics.
June 1997.
16 pp.
Abstract: In this paper we present a generalisation of a promising compositional model checking technique introduced for finite-state systems by Andersen and extended to networks of timed automata by Larsen et al. In our generalized setting, programs are modelled as arbitrary (possibly infinite-state) transition systems and verified with respect to properties of a basic safety logic. As the fundamental prerequisite of the compositional technique, it is shown how logical properties of a parallel program may be transformed into necessary and sufficient properties of components of the program. Finally, a set of axiomatic laws are provided useful for simplifying formulae and complete with respect to validity and unsatisfiability.

[RS-97-12] *PostScript*, *PDF*, *DVI.*
Andrej Brodnik, Peter Bro Miltersen, and J. Ian Munro.
Trans-Dichotomous Algorithms without Multiplication — some Upper and Lower Bounds.
May 1997.
19 pp. Appears in Dehne, Rau-Chaulin, Sack and Tamassio, editors, *Algorithms and Data Structures: 5th International Workshop*, WADS '97 Proceedings, LNCS 1272, 1997, pages 426–439.
Abstract: We show that on a RAM with addition, subtraction, bitwise Boolean operations and shifts, but no multiplication, there is a trans-dichotomous solution to the static dictionary problem using linear space and with query time $\sqrt{\log n(\log\log n)^{1+o(1)}}$. On the way, we show that two w-bit words can be multiplied in time $(\log w)^{1+o(1)}$ and that time $\Omega(\log w)$ is necessary, and that $\Theta(\log\log w)$ time is necessary and sufficient for identifying the least significant set bit of a word.

[RS-97-11] *PostScript*, *PDF*, *DVI.*
Kārlis Čerāns, Jens Chr. Godskesen, and Kim G. Larsen.
Timed Modal Specification — Theory and Tools.
April 1997.
32 pp.

Abstract: In this paper we present the theory of *Timed Modal Specifications* (TMS) together with its implementation, the tool EPSILON. TMS and EPSILON are timed extensions of respectively *Modal Specifications* and the TAV system.

The theory of TMS is an extension of real–timed process calculi with the specific aim of allowing *loose* or *partial* specifications. Looseness of specifications allows implementation details to be left out, thus allowing several and varying implementations. We achieve looseness of specifications by introducing two modalities to transitions of specifications: a *may* and a *must* modality. This allows us to define a notion of *refinement*, generalizing in a natural way the classical notion of bisimulation. Intuitively, the more must–transitions and the fewer may–transitions a specification has, the finer it is. Also, we introduce notions of refinements abstracting from time and/or internal computation.

TMS specifications may be combined with respect to the constructs of the real–time calculus. "Time–sensitive" notions of refinements that are preserved by these constructs are defined (to be precise, when abstracting from internal composition refinement is not preserved by choice for the usual reasons), thus enabling compositional verification.

EPSILON provides automatic tools for verifying refinements. We apply EPSILON to a compositional verification of a train crossing example.

[RS-97-10] *PostScript*, *PDF*, *DVI*.
Thomas Troels Hildebrandt and Vladimiro Sassone.
Transition Systems with Independence and Multi-Arcs.
April 1997.
20 pp. Appears in Peled, Pratt and Holzmann, editors, *DIMACS Workshop on Partial Order Methods in Verification*, POMIV '96 Proceedings, American Mathematical Society Press (AMS) DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 1996, pages 273–288.

Abstract: We extend the model of transition systems with independence in order to provide it with a feature relevant in the *noninterleaving* analysis of concurrent systems, namely *multi-arcs*. Moreover, we study the relationships between the category of transition systems with independence and multi-arcs and the category of labeled asynchronous transition systems, extending the results recently obtained by the authors for (simple) transition systems with independence (cf. *Proc. CONCUR '96*), and yielding a precise characterisation of transition systems with independence and multi-arcs in terms of (*event-maximal*, *diamond-extensional*) labeled asynchronous transition systems.

[RS-97-9] *PostScript*, *PDF*, *DVI*.
Jesper G. Henriksen and P. S. Thiagarajan.
A Product Version of Dynamic Linear Time Temporal Logic.
April 1997.

18 pp. Appears in Mazurkiewicz and Winkowski, editors, *Concurrency Theory: 8th International Conference*, CONCUR '97 Proceedings, LNCS 1243, 1997, pages 45–58.

Abstract: We present here a linear time temporal logic which simultaneously extends LTL, the propositional temporal logic of linear time, along two dimensions. Firstly, the until operator is strengthened by indexing it with the regular programs of propositional dynamic logic (PDL). Secondly, the core formulas of the logic are decorated with names of sequential agents drawn from fixed finite set. The resulting logic has a natural semantics in terms of the runs of a distributed program consisting of a finite set of sequential programs that communicate by performing common actions together. We show that our logic, denoted $\mathrm{DLTL}^{\otimes}$, admits an exponential time decision procedure. We also show that $\mathrm{DLTL}^{\otimes}$ is expressively equivalent to the so called regular product languages. Roughly speaking, this class of languages is obtained by starting with synchronized products of (ω-)regular languages and closing under boolean operations. We also sketch how the behaviours captured by our temporal logic fit into the framework of labelled partial orders known as Mazurkiewicz traces.

[RS-97-8] *PostScript*, *PDF*, *DVI*.
Jesper G. Henriksen and P. S. Thiagarajan.
Dynamic Linear Time Temporal Logic.
April 1997.
33 pp. Full version of paper appearing in *Annals of Pure and Applied Logic* 96(1-3), pp. 187-207 (1999).

Abstract: A simple extension of the propositional temporal logic of linear time is proposed. The extension consists of strengthening the until operator by indexing it with the regular programs of propositional dynamic logic (PDL). It is shown that DLTL, the resulting logic, is expressively equivalent to S1S, the monadic second-order theory of ω-sequences. In fact a sublogic of DLTL which corresponds to propositional dynamic logic with a linear time semantics is already as expressive as S1S. We pin down in an obvious manner the sublogic of DLTL which correponds to the first order fragment of S1S. We show that DLTL has an exponential time decision procedure. We also obtain an axiomatization of DLTL. Finally, we point to some natural extensions of the approach presented here for bringing together propositional dynamic and temporal logics in a linear time setting.

[RS-97-7] *PostScript*, *PDF*, *DVI*.
John Hatcliff and Olivier Danvy.
Thunks and the λ-Calculus (Extended Version).
March 1997.
55 pp. Extended version of article appearing in *Journal of Functional Programming*, 7(3):303–319, May 1997.

Abstract: Plotkin, in his seminal article *Call-by-name, call-by-value and the λ-calculus*, formalized evaluation strategies and simulations using operational semantics and continuations. In particular, he showed how call-by-name evaluation could be simulated under call-by-value evaluation and vice versa. Since Algol 60, however, call-by-name is both implemented and simulated with thunks rather than with continuations. We recast this folk theorem in Plotkin's setting, and show that thunks, even though they are simpler than continuations, are sufficient for establishing all the correctness properties of Plotkin's call-by-name simulation.

Furthermore, we establish a new relationship between Plotkin's two continuation-based simulations $\mathcal{C}_n$ and $\mathcal{C}_v$, by *deriving* $\mathcal{C}_n$ as the composition of our thunk-based simulation $\mathcal{T}$ and of $\mathcal{C}_v^+$ — an extension of $\mathcal{C}_v$ handling thunks. Almost all of the correctness properties of $\mathcal{C}_n$ follow from the properties of $\mathcal{T}$ and $\mathcal{C}_v$. This simplifies reasoning about call-by-name continuation-passing style.

We also give several applications involving factoring continuation-based transformations using thunks.

[RS-97-6] *PostScript*, *PDF*, *DVI*.
Olivier Danvy and Ulrik P. Schultz.
Lambda-Dropping: Transforming Recursive Equations into Programs with Block Structure.
March 1997.
53 pp. Extended version of an article appearing in the 1997 *ACM SIGPLAN Symposium on Partial Evaluation and Semantics-Based Program Manipulation*, PEPM '97 Proceedings (Amsterdam, The Netherlands, June 12–13, 1997), ACM SIGPLAN Notices, 32(12), pages 90–106. This report is superseded by the later BRICS Report RS-99-27.

Abstract: Lambda-lifting a functional program transforms it into a set of recursive equations. We present the symmetric transformation: lambda-dropping. Lambda-dropping a set of recursive equations restores block structure and lexical scope.

For lack of scope, recursive equations must carry around all the parameters that any of their callees might possibly need. Both lambda-lifting and lambda-dropping thus require one to compute a transitive closure over the call graph:

- for lambda-lifting: to establish the Def/Use path of each free variable (these free variables are then added as parameters to each of the functions in the call path);
- for lambda-dropping: to establish the Def/Use path of each parameter (parameters whose use occurs in the same scope as their definition do not need to be passed along in the call path).

Without free variables, a program is scope-insensitive. Its blocks are then free to float (for lambda-lifting) or to sink (for lambda-dropping) along the vertices of the scope tree.

We believe lambda-lifting and lambda-dropping are interesting

per se, both in principle and in practice, but our prime application is partial evaluation: except for Malmkjr and rbk's case study presented at PEPM'95, most polyvariant specializers for procedural programs operate on recursive equations. To this end, in a pre-processing phase, they lambda-lift source programs into recursive equations. As a result, residual programs are also expressed as recursive equations, often with dozens of parameters, which most compilers do not handle efficiently. Lambda-dropping in a post-processing phase restores their block structure and lexical scope thereby significantly reducing both the compile time and the run time of residual programs.

[RS-97-5] *PostScript*, *PDF*, *DVI.*
Kousha Etessami, Moshe Y. Vardi, and Thomas Wilke.
First-Order Logic with Two Variables and Unary Temporal Logic.
March 1997.
18 pp. Appears in *Twelfth Annual IEEE Symposium on Logic in Computer Science*, LICS '97 Proceedings, 1997, pages 228–235.

Abstract: We investigate the power of first-order logic with only two variables over ω-words and finite words, a logic denoted by FO^2. We prove that FO^2 can express precisely the same properties as linear temporal logic with only the unary temporal operators: "next", "previously", "sometime in the future", and "sometime in the past", a logic we denote by unary-TL. Moreover, our translation from FO^2 to unary-TL converts every FO^2 formula to an equivalent unary-TL formula that is at most exponentially larger, and whose operator depth is at most twice the quantifier depth of the first-order formula. We show that this translation is optimal.

While satisfiability for full linear temporal logic, as well as for unary-TL, is known to be PSPACE-complete, we prove that satisfiability for FO^2 is NEXP-complete, in sharp contrast to the fact that satisfiability for FO^3 has non-elementary computational complexity. Our NEXP time upper bound for FO^2 satisfiability has the advantage of being in terms of the *quantifier depth* of the input formula. It is obtained using a small model property for FO^2 of independent interest, namely: a satisfiable FO^2 formula has a model whose "size" is at most exponential in the quantifier depth of the formula. Using our translation from FO^2 to unary-TL we derive this small model property from a corresponding small model property for unary-TL. Our proof of the small model property for unary-TL is based on an analysis of unary-TL types.

[RS-97-4] *PostScript*, *PDF*, *DVI.*
Richard Blute, Jose Desharnais, Abbas Edalat, and Prakash Panangaden.
Bisimulation for Labelled Markov Processes.
March 1997.
48 pp. Appears in *Twelfth Annual IEEE Symposium on Logic in Computer Science*, LICS '97 Proceedings, 1997, pages 149–158.

Abstract: In this paper we introduce a new class of labelled transition systems - Labelled Markov Processes - and define bisimulation for them. Labelled Markov processes are probabilistic labelled transition systems where the state space is not necessarily discrete, it could be the reals, for example. We assume that it is a Polish space (the underlying topological space for a complete separable metric space). The mathematical theory of such systems is completely new from the point of view of the extant literature on probabilistic process algebra; of course, it uses classical ideas from measure theory and Markov process theory. The notion of bisimulation builds on the ideas of Larsen and Skou and of Joyal, Nielsen and Winskel. The main result that we prove is that a notion of bisimulation for Markov processes on Polish spaces, which extends the Larsen-Skou definition for discrete systems, is indeed an equivalence relation. This turns out to be a rather hard mathematical result which, as far as we know, embodies a new result in pure probability theory. This work heavily uses continuous mathematics which is becoming an important part of work on hybrid systems.

[RS-97-3] *PostScript*, *PDF*, *DVI*.
Carsten Butz and Ieke Moerdijk.
A Definability Theorem for First Order Logic.
March 1997.
10 pp. Appears in *Journal of Symbolic Logic*, 64(3):1028–1036, 1999.

Abstract: For any first order theory T we construct a Boolean valued model M, in which precisely the T–provable formulas hold, and in which every (Boolean valued) subset which is invariant under all automorphisms of M is definable by a first order formula.

[RS-97-2] *PostScript*, *PDF*, *DVI*.
David A. Schmidt.
Abstract Interpretation in the Operational Semantics Hierarchy.
March 1997.
33 pp.

Abstract: We systematically apply the principles of Cousot-Cousot-style abstract interpretation to the hierarchy of operational semantics definitions—flowchart, big-step, and small-step semantics. For each semantics format we examine the principles of safety and liveness interpretations, first-order and second-order analyses, and termination properties. Applications of abstract interpretation to data-flow analysis, model checking, closure analysis, and concurrency theory are demonstrated. Our primary contributions are separating the concerns of safety, termination, and efficiency of representation and showing how abstract interpretation principles apply uniformly to the various levels of the operational semantics hierarchy and their applications.

[RS-97-1] *PostScript*, *PDF*, *DVI*.
Olivier Danvy and Mayer Goldberg.
Partial Evaluation of the Euclidian Algorithm (Extended Version).

January 1997.

16 pp. Appears in the journal *LISP and Symbolic Computation*, 10(2):101–111, July 1997.

Abstract: Some programs are easily amenable to partial evaluation because their control flow clearly depends on one of their parameters. Specializing such programs with respect to this parameter eliminates the associated interpretive overhead. Some other programs, however, do not exhibit this interpreter-like behavior. Each of them presents a challenge for partial evaluation. The Euclidian algorithm is one of them, and in this article, we make it amenable to partial evaluation.

We observe that the number of iterations in the Euclidian algorithm is bounded by a number that can be computed given either of the two arguments. We thus rephrase this algorithm using bounded recursion. The resulting program is better suited for automatic unfolding and thus for partial evaluation. Its specialization is efficient.