

BRICS *Newsletter*

Basic Research in Computer Science

No 3, July 95

In this Issue


Welcome	1
BRICS Themes	2
Logic in Semantics '95	2
Verification '96	3
Reports on Conferences and Workshops	3
Reports on Courses	5
Newly Appointed Researchers and PhD's	7
Coming Events	7
Theme Events	7
Evolving Algebras	7
Analysis and Transformation of Set-Theoretic Languages	8
Secure Multiparty Computations	8
Compact Routing Methods in Parallel and Distributed Systems	8
New Reports	8
Notes Series	10
Lecture Series	10
Technical Contributions	12
A General Splitting Lemma	12
UPPAAL	13
Calendar of Events	16
BRICS Address and World Wide Web	16

Welcome

Welcome to the third issue of the BRICS newsletter. Its purpose is to inform you of appointments, publications, courses and other activities within

BRICS. Further details can be obtained by contacting the addresses below.

We stand now at the middle of our second year, having just completed a second successful theme in Logic in Semantics. To date we have 87 publications in our report series and 10 in the notes series. In addition, we have begun a Lecture Series for the propagation of high quality lecture notes on topics in the foundations of Computer Science—we presently have 3. These are all available electronically. All told there are many indications that BRICS is becoming widely regarded as a significant site for theoretical Computer Science. A part of our contribution is through hosting and chairing conferences (we have recently held TAPSOFT, shall host CSL '97 and Glynn Winskel is to be program chair for LICS '97). But, ultimately a more important contribution is the active participation of BRICS researchers on the international research scene. We are pleased by the level of activity of BRICS researchers and their visibility through presentations at prestigious conferences. Our visibility has led to a satisfyingly high number and broad international spread of well qualified applicants for BRICS positions.

Closer to home, we have made efforts to explain the enterprises of BRICS to our younger students. In particular, a course on “Pearls of theory”, described in more detail below, introduces students to a range of issues, techniques and problems in theoretical Computer Science. We will continue to work towards an exciting research environment that can be felt also at the undergraduate level. It's surely not just a coincidence that we recently attracted an impressive pool of applications for PhD's in the areas of BRICS. 

BRICS Themes

Logic in Semantics '95

The theme for '95 has been *Logic in Semantics*, concentrating on new developments in the semantics of programming languages: proof principles for reasoning about functional programs and recursive types (e.g., Andy Pitts' and Andy Gordon's courses), synthetic domain theory (e.g., Pino Rosolini's course, Edmund Robinson's visit), full abstraction and stable and sequential domain theory (e.g., Full Abstraction Workshop). Because of a similar programme at the Isaac Newton Institute at Cambridge late in the year, the BRICS theme took place before mid summer '95, with two activities (Andy Pitts' mini-course and the Peripatetic Seminar on Sheaves and Logic) happening already in late '94.

In connection with the theme we were very pleased to have Edmund Robinson (University of Sussex, soon to take the Chair of Computer Science at Queen Mary College, London) visit us for two months. Edmund is an expert in categorical logic and its applications to semantics and logics of programs. Edmund's stay, accompanied by longer visits of John Power, Bart Jacobs and Pino Rosolini, as well as many short-stay guests and the local BRICS expertise, generated an exciting, friendly and concentrated research environment on topics logical and categorical in the study of programming languages.

Guests of the theme included: Abramsky, Bucciarelli, Curien, Coquand, Ehrhard, Gandy, Geva, Gordon, Jagadeesan, Jung, Malacaria, Nickau, O'Hearn, Ong, Pani, Pavlović, Pitts, Power, Riecke, Robinson, Rosolini, Sazonov, Sieber, Stoughton, Streicher, Zawadowski.

A workshop on full abstraction for PCF and related languages was organised for the two weeks starting April 18, 1995. PCF is a simple functional programming language based on the typed lambda calculus with fixed point operators, and yet with enough features to make the understanding of its fully abstract model extremely difficult, as is underlined in our ignorance about

certain decidability issues to do with PCF. The workshop was lucky in attracting a good proportion of the world's expertise in the mathematical theory of functional languages. In organising the workshop a special effort was made to leave enough time for discussion, and joint work, a formula that apparently worked as several participants have commented on the workshop being the most productive and enjoyable that they have attended.

The workshop began with a broad discussion on the nature of models for PCF (Stoughton) which was followed by a fairly intensive two days on logical relations both for building and reasoning about the fully abstract model of PCF (Riecke, Sieber, Jung and Stoughton), as well as issues related to sequentiality in programming languages with local state (O'Hearn, Pitts). Sequential algorithms as well as their relations with the recent game semantics of PCF and Herbelin's work were discussed in the two broad sweeping talks of Curien. Interesting examples and problems, as well as historical insight were provided by work on the lines of traditional higher type recursion theory (Gandy, Pani, Nickau, Sazonov). The more leisurely style of the workshop gave room for fairly full expositions, amounting to mini-courses, on game semantics (Abramsky, Jagadeesan, Malacaria, Ong) and strong stability (Ehrhard, Bucciarelli)—the latter included a new method (Bucciarelli) for relating the equational theories arising from models—and time for a recapitulation on the open problems brought to light during the workshop.

One outcome of the meeting has been a rather nontraditional form of proceedings (but one which will surely gain currency) consisting of a WWW page assembling open problems and the relevant literature.

The workshop created an opportunity of another kind. Dr. Robin Gandy, formerly Reader at Oxford University and Alan Turing's only Ph.D. student, participated in the workshop. He kindly agreed to give a talk on personal recollections of

his supervisor and friend. Robin Gandy's lecture, delivered to a crammed lecture theatre, touched on many themes, personal and scientific, in the short life of Alan Turing, the great computer pioneer. Many of us carry life-long memories away from Robin Gandy's talk.

The workshop and lecture series were accompanied by a number of mini courses on the theme of Logic in Semantics:

- Inductive and Co-inductive Techniques in the Semantics of Functional Programs, by Andrew Pitts, Computer Laboratory, Cambridge University, December '94, BRICS Notes series, NS-94-5.
- The modal μ -Calculus, by Igor Walukiewicz, BRICS, February '95, BRICS note NS-95-1.
- Bisimilarity as a Theory of Functional Programming, by Andrew Gordon, Computer Laboratory, Cambridge University, March '95, BRICS Notes series NS-95-3.
- Synthetic Domain Theory, by Pino Rosolini, University of Genova, March '95. ☰

Verification '96

Following previous successful BRICS themes of complexity theory (94) and full abstraction (95), the topic of the 1996 theme will be *Verification*.

The exact form and schedule for this event has not been decided yet. But the activities are intended to cover verification of computing systems in a broad sense, spanning from foundational studies of specification formalisms and proof principles to the technology of automated tools for verification in practice.

The program will contain a special scheme for long and short term visitors, and small workshops focussing on specific subjects.

Kim G. Larsen and Mogens Nielsen are responsible for the detailed planning, which will take place during the autumn 1995, see our BRICS Activities Web page for up to date information and the October issue of the EATCS bulletin. Also, everybody potentially interested in taking part in the activities is welcome to contact us directly at BRICS@brics.aau.dk. ☰

Reports on Conferences and Workshops

Workshop on Full Abstraction of PCF and Related Languages

As part of this year's BRICS theme in "Logic and Semantics", Glynn Winskel hosted a two week workshop on "Full Abstraction of PCF and Related Languages" from 18 to 28 April, 1995. This focussed attention on recent developments in the semantics of functional languages, in particular, exploring in depth such topics as: game semantics and intensional full abstraction, new work on sequentiality, in particular, strong stability, the use of logical relations in reasoning and refining models, as well as related problems to do with local state. The workshop succeeded in attracting most of the experts in the area, participants

including: Abramsky, Bucciarelli, Curien, Coquand, Ehrhard, Gandy, Geva, Jagadeesan, Jung, Malacaria, Nickau, O'Hearn, Ong, Pani, Pitts, Riecke, Sazonov, Sieber, Stoughton. Based on the workshop, a collection of open problems is being assembled, along with a BRICS WWW page¹ collecting the relevant literature. ☰

Workshop on Tools and Algorithms for the Construction and Analysis of Systems

Aarhus, Denmark, 19–20 May 1995.

Carsten Weise, University of Technology Aachen, Germany has kindly proved us with a personal

¹<http://www.brics.aau.dk/BRICS/FA/>

report on the workshop.

In May 1995, Aarhus not only saw TAPSOFT '95, but also hosted – among other satellite meetings – the first TACAS Workshop. The aim of the TACAS workshop is to bring together researchers and practitioners with a strong interest in tools for construction, analysis, specification and verification of distributed systems.

Taking place on two sunny days, the workshop was a very vivid event and surely lived up to meet its objectives. A lot of interesting talks were given on these two days, ranging from case studies over tool and meta-tool presentations to the description of new and improved methodologies to tackle a variety of problems in the automatic and semi-automatic analysis and synthesis of systems.

Besides the essentially high quality of the talks, the atmosphere at the workshop was friendly and amicable, and we had a lot of fun during and after the talks. Let me just mention some of the highlights that come to my mind:

From Bill Roscoe we learned that he likes to build verification tools as he then can use them to solve puzzles like Solitaire and the like.

Wang Yi presented a new version of the fight between David and Goliath when he told Pei-Hsin Ho that the new UppAal Tool, a small and fast tool aiming at special subclasses of Hybrid Systems, verified Fischer's Protocol four hundred times faster than HyTech, Henzinger and Ho's universal tool for Hybrid Systems. This will probably encourage Pei-Hsin Ho to improve on the speed of his very fine HyTech tool.

Wil Jansen shared with us his knowledge of knowledge transitions, which produced a common knowledge of his knowledge in all of the TACAS participants.

Between the talks, there was the opportunity to witness demonstrations of the presented tools – among them FDR, HyTech, Isabelle, MONA and UppAal – so the participants could get an impression of the capabilities and possibilities of the specific tools. These demonstrations were an

important part of the TACAS workshop, as they gave the chance to look at the tools presented or used in the talks. It was even possible to try examples by oneself.

The TACAS workshop was also a success as a social event, and it was a pleasure to meet old friends and make new contacts with colleagues. The central event apart from talks and demonstrations was the joyful workshop dinner, which ended with the old Danish tradition of singing some funny songs, although I must admit that I did not understand all the words that were sung.

So although the meeting was announced at short notice and – as we heard from Kim Guldstrand Larsen in his dinner speech – had to be organized remotely, the first Workshop on Tools and Algorithms for the Construction and Analysis of Systems (short TACAS) was a clear success. I enjoyed visiting the workshop (and Aarhus, once again) and I profitted from attending it. So let me just thank the organizers of TACAS '95, and I am looking forward to TACAS '96 in Passau, where I hope to meet not only this year's participants again, but also many more.

—≡—

From *Kim G. Larsen* of the TACAS Programme Committee we have received the following.

This TAPSOFT satellite workshop, TACAS, brought together 46 researcher interested in the development and application of tools and algorithms for specification, verification, analysis and construction of distributed systems. The overall goal of the workshop was to compare the various methods and the degree to which they are supported by interacting or fully automatic tools.

During the two days 23 presentations were given covering a variety of topics including refinement-based and compositional verification, construction techniques, analysis and verification via theorem-proving, process algebras, temporal and modal logics, techniques for real-time, hybrid and probabilistic systems, and approaches for value-passing systems. In addition 9 tools were demonstrated. A selection of 13 papers are invited for publication in a future volume

of Springer Verlag's Lecture Notes in Computer Science series.

In light of the very short notice for call-for-paper the Programme Committee was extremely happy to experience such a high number of attendants. It was therefore decided to repeat the workshop next year March 27–29 at Passau University, Germany. For a preliminary call for papers for the TACAS '96 workshop consult the reference².

— ■ —

All papers of the proceedings, NS-95-2, are electronically available (see Abstract of 95-2 in Notes Series of Publications of BRICS WWW). ■■

Reports on Courses

Bisimilarity as a Theory of Functional Programming

Andrew D. Gordon, University of Cambridge Computer Laboratory, held a 2 lecture mini-course Wednesday, 22nd March and 23rd March 1995. The course explained, at a tutorial level, new developments in the theory of proof principles for functional programming languages, and was, in particular, very well-attended by beginning second part students. The course notes are published as BRICS note NS-95-3 below.

The modal μ -Calculus

Igor Walukiewicz, BRICS, held a mini-course of 5 lectures a week in February on the modal μ -calculus, culminating with his proof of the completeness of Kozen's proof system—a long-standing open problem. The course notes are published as BRICS note NS-95-1 below.

Sixth International Joint Conference on the Theory and Practice of Software Development

Aarhus, Denmark, 22–26 May 1995.

This summer BRICS was host for TAPSOFT '95. It was attended by some 125 participants from all over the world. Apart from the traditional three parts, Invited Lectures, CAAP (Colloquium on Trees in Algebra and Programming) and FASE (Colloquium on Formal Approaches in Software Engineering), this year's TAPSOFT had a new successful part on TOOLS. The invited lecture of Vaughan Pratt attracted the attention of the Danish press—his talk on the anatomy of the Pentium bug emphasized the importance of automated verification.

Michel Bidoit has most kindly agreed to report from the TAPSOFT conference in the next issue of the newsletter. ■■

Synthetic Domain Theory

Pino Rosolini, University of Genova, visited 5–15 March and gave a mini-course (5 lectures) on synthetic domain theory.

Pearls of Theory

In the spring of 1995 the kernel researchers at BRICS offered the introductory graduate level course *Pearls of Theory* based on a novel format. The aim was to expose younger students to the areas of theoretical computer science that are researched at the BRICS Centre.

Unlike other theory courses, *Pearls of Theory* contained a succession of introductory lectures on a variety of topics. Each lecture lasted only one or two weeks and was delivered at a quick pace

²<http://www.uni-passau.de/fmi/lehrstuehle/steffen/cfp/tacas.html>

focussing on a self-contained topical or classical subject. The students were required to hand in solutions to exercises and were challenged with the *Oyster of the Week*—a particularly hard problem.

The course was attended by some 20 students as well as BRICS personnel and other local faculty members. The lectures were given by both kernel, associated, and visiting researchers according to the following plan:

Sven Skyum: A Non-Linear Lower Bound
Jens Palsberg: Set Constraints
Mogens Nielsen: Binary Decision Diagrams
Michael I. Schwartzbach:
 Polymorphic Type Inference
Ivan Damgaard: Zero Knowledge
Olivier Danvy: Partial Evaluation
Erik Meineche Schmidt: Tradeoffs
Jaap van Oosten: Intuitionistic Logic
Glynn Winskel: Model Checking
Peter Mosses: Applications of Modal Logic
Mogens Nielsen: Model Checking - Decidable?
Prakash Panangaden: Data Flow
Sven Skyum: Backwards Analysis

Abstracts of the lectures are available on the Web³. See also LS-95-2 and LS-95-3 of the Lectures Series.

It is intended that similar courses will become permanent fixtures of the Spring curriculum. ☰

Randomness and Computation

Aravind Srinivasan, National Univ. of Singapore, gave May 2 – May 9 a mini-course intended to serve both as a tutorial introduction to the role of randomness in computation, specifically in algorithms and complexity, and as a forum for the discussion of the most recent research in the area. The course was organised into a sequence of 6 lectures: 1. Introduction, 2. Randomness in Distributed Computing, 3. Derandomisation, 4. Random Sampling, 5. The Lovasz Local Lemma, 6. Weak Random Sources.

Lecture notes are under preparation and will be available as part of the BRICS Lecture Notes series in a few months' time.

In addition, the course was supplemented by a talk on “Improved Approximation Algorithms for Packing and Covering Problems”.

The course was attended by researchers in the area at Aarhus, Copenhagen and Odense and by students at Aarhus interested in the area. ☰

Dynamic Graph Algorithms

Giuseppe (Pino) Italiano, University of Salerno, Italy, gave in the period May 29 to June 2 a basic course in the important area of dynamic graph algorithms. Professor Italiano is a well-known authority on the subject. The course was organised into four lectures, each lasting approximately two hours: 1. Introduction and Basic Dynamic Graph Algorithms, 2. Sparsification, 3. Semi-Dynamic Graph Algorithms, 4. Fully Dynamic Randomized Connectivity.

The first lecture was introductory and covered the seminal work of Frederickson [3]. The second lecture introduced the simple but powerful general method to speed up dynamic graph algorithms called sparsification introduced by [2]. The third lecture considered simplifications resulting from considering only partially dynamic algorithms. The final lecture covered the very recent work of [4]. A draft of a forthcoming book on the subject [1] was also circulated in addition to the covered literature. For further information, see the BRICS WWW page.

The course was very well attended by researchers and students from Aarhus and Odense.

References

- [1] D. Eppstein, Z. Galil and G.F. Italiano, *Dynamic Graph Algorithms*, preliminary draft of a book.

³<http://www.brics.aau.dk/BRICS/Activities/pearls>

[2] D. Eppstein, Z. Galil and G.F. Italiano and A. Nissenzweig, "Sparsification: A Technique for Speeding up Dynamic Graph Algorithms", FOCS, 1992.

[3] G.N. Frederickson, "Data Structures for On-line Updating of Minimum Spanning Trees,

with Applications", *SIAM J. Comput.*, 14:4, 1985.

[4] M. Rauch-Henzinger and V. King, "Randomised Dynamic Graph Algorithms with Polylogarithmic Time per Operation", STOC 1995. ☐

Newly Appointed Researchers and PhD's

Ian Stark

Ian Stark recently obtained his PhD from Cambridge University, where he was supervised by Dr. Andrew Pitts. His thesis work concentrates on the hard problem of semantics and proof principles for functional languages with local state. He joins BRICS in November '95 after a postdoctoral position at the University of Pisa.

We would also like to welcome our newly admitted PhD students.

Ålborg: Henrik Ejerbo Jensen and Josvan Kleist.

Aarhus: Søren Bøgh Lassen, Jesper Gulmann Henriksen, Thomas Troels Hildebrandt, Rune Bang Lyngsø, Christian Nørgaard Storm Pedersen and Anders Bækgaard Sandholm.

This summer BRICS is hosting two visiting summer PhD students from abroad. It has been a very positive experience and we expect it to be an recurring event in the summers to come.

Rowan Davies

Rowan Davies is a graduate student (originally from Australia) at Carnegie Mellon University, USA, under the direction of Frank Pfenning. He hasn't yet proposed a thesis, but his main project recently is the application of type systems based on modal logic to the division of programs into computation stages. He will graduate around 1998. This summer, during his visit to BRICS, He is working on the implementation of Mona, and also the extension of the modal type system to encompass the state of the art in partial evaluation.

Mayer Goldberg

Mayer Goldberg is a graduate student at Indiana University, in his last year, working under Prof. Daniel P. Friedman. His research is in the type-free lambda calculus, where he study a phenomenon called "application survival". He is extremely interested in solving problems involving self-application, where traditional methods have proven unsuccessful. ☐

Coming Events

Theme Events

Evolving Algebras

Egon Börger, University of Pisa, and Yuri Gurevich, University of Michigan, will be visiting BRICS for all of August 1995. They are working with an approach to semantics called "Evolving Algebras".

In the week from 7–11 August, Professor Gurevich will give a short course of four lectures on "Evolving Algebras from First Principles": 1. The EA computation model, 2. Sequential Ealgebras, 3. Parallel and distributed Ealgebras, EA Proofs. Professor Börger will follow on with a series of three seminars concerning applications of evolving algebra semantics: 1. The logical structure of pipelining in RISC architectures, 2. Correctness of Compiling Occam Programs to Transputer Code

3. A Mathematical Specification of the APE100 Parallel Architecture.

The course and lectures are open to all. BRICS can assist in arranging accommodation, and possibly with local expenses in a few needy cases. For further information, click on 'Activities' on the BRICS WWW page, or contact BRICS@brics.aau.dk.

Copies of lecture notes are available in the BRICS Notes Series, NS-95-4.

Analysis and Transformation of Set-Theoretic Languages

In the week of 14–18 August *Robert Paige*, Courant Institute, New York University, will give a short course on how types and transformations can be used to integrate algorithm design and analysis, program development, and high level compilation of set theoretic programming languages. Topics: 1. Introduction to SETL and sources of inefficiency; program improvement by finite differencing, 2. Program improvement by real-time simulation of a typed set machine (equipped with high level input/output) on a RAM, 3. Reconstruction and extension of the lin-

ear time fragment of Willard's database predicate retrieval theory, 4. Design of A linear time fixed point language; experiments in productivity of algorithm implementation (live demonstration). More information can be found under 'Activities' on the BRICS Web page. Please contact BRICS@brics.aau.dk in case you want to attend the course or have questions.

Secure Multiparty Computations

Michael Ben-Or, Hebrew University, Jerusalem, will give a mini-course on secure multiparty computations in the week 21–25 August. For further information and a detailed plan, please contact Sven Skyum, sskyum@brics.aau.dk.

Compact Routing Methods in Parallel and Distributed Systems

In September *Richard Tan*, University of Utrecht and university of Oklahoma, will give a course on compact routing methods in parallel and distributed systems. For further information and a detailed plan, please contact Sven Skyum, sskyum@brics.aau.dk.

BRICS Report Series for 1995

ISSN 0909-0878

39 Allan Cheng. *Petri Nets, Traces, and Local Model Checking*. July 1995. 32 pp. Full version of paper appearing in Proceedings of AMAST '95, LNCS 936, 1995.

38 Mayer Goldberg. *Gödelisation in the λ -Calculus*. July 1995. 7 pp.

37 Sten Agerholm and Mike Gordon. *Experiments with ZF Set Theory in HOL and Isabelle*. July 1995. 14 pp. To appear in *Proceedings of the 8th International Workshop on Higher Order Logic Theorem Proving and its Applications*, LNCS, 1995.

36 Sten Agerholm. *Non-primitive Recursive Function Definitions*. July 1995. 15 pp. To appear

in *Proceedings of the 8th International Workshop on Higher Order Logic Theorem Proving and its Applications*, LNCS, 1995.

35 Mayer Goldberg. *Constructing Fixed-Point Combinators Using Application Survival*. June 1995. 14 pp.

34 Jens Palsberg. *Type Inference with Selftype*. June 1995. 22 pp.

33 Jens Palsberg, Mitchell Wand, and Patrick O'Keefe. *Type Inference with Non-structural Subtyping*. June 1995. 22 pp.

32 Jens Palsberg. *Efficient Inference of Object Types*. June 1995. 32 pp. To appear in *Information*

- and Computation. Preliminary version appears in *Ninth Annual IEEE Symposium on Logic in Computer Science*, LICS '94 Proceedings, pages 186–195.
- 31 Jens Palsberg and Peter Ørbæk. *Trust in the λ -calculus*. June 1995. 32 pp. To appear in *Static Analysis: 2nd International Symposium*, SAS '95 Proceedings, 1995.
 - 30 Franck van Breugel. *From Branching to Linear Metric Domains (and back)*. June 1995. 30 pp. Abstract appeared in Engberg, Larsen, and Mosses, editors, *6th Nordic Workshop on Programming Theory*, NWPT '6 Proceedings, 1994, pages 444–447.
 - 29 Nils Klarlund. *An $n \log n$ Algorithm for Online BDD Refinement*. May 1995. 20 pp.
 - 28 Luca Aceto and Jan Friso Groote. *A Complete Equational Axiomatization for MPA with String Iteration*. May 1995. 39 pp.
 - 27 David Janin and Igor Walukiewicz. *Automata for the μ -calculus and Related Results*. May 1995. 11 pp. To appear in *Mathematical Foundations of Computer Science: 20th Int. Symposium*, MFCS '95 Proceedings, LNCS, 1995.
 - 26 Faith Fich and Peter Bro Miltersen. *Tables should be sorted (on random access machines)*. May 1995. 11 pp. To appear in *Algorithms and Data Structures: 4th Workshop*, WADS '95 Proceedings, LNCS, 1995.
 - 25 Søren B. Lassen. *Basic Action Theory*. May 1995. 47 pp.
 - 24 Peter Ørbæk. *Can you Trust your Data?* April 1995. 15 pp. Appears in Mosses, Nielsen, and Schwartzbach, editors, *Theory and Practice of Software Development. 6th International Joint Conference CAAP/FASE*, TAPSOFT '95 Proceedings, LNCS 915, 1995, pages 575–590.
 - 23 Allan Cheng and Mogens Nielsen. *Open Maps (at) Work*. April 1995. 33 pp.
 - 22 Anna Ingólfssdóttir. *A Semantic Theory for Value-Passing Processes, Late Approach, Part II: A Behavioural Semantics and Full Abstractness*. April 1995. 33 pp.
 - 21 Jesper G. Henriksen, Ole J. L. Jensen, Michael E. Jørgensen, Nils Klarlund, Robert Paige, Theis Rauhe, and Anders B. Sandholm. *MONA: Monadic Second-Order Logic in Practice*. May 1995. 17 pp.
 - 20 Anders Kock. *The Constructive Lift Monad*. March 1995. 18 pp.
 - 19 François Laroussinie and Kim G. Larsen. *Compositional Model Checking of Real Time Systems*. March 1995. 20 pp.
 - 18 Allan Cheng. *Complexity Results for Model Checking*. February 1995. 18pp.
 - 17 Jari Koistinen, Nils Klarlund, and Michael I. Schwartzbach. *Design Architectures through Category Constraints*. February 1995. 19 pp.
 - 16 Dany Breslauer and Ramesh Hariharan. *Optimal Parallel Construction of Minimal Suffix and Factor Automata*. February 1995. 9 pp.
 - 15 Devdatt P. Dubhashi, Grammati E. Pantziou, Paul G. Spirakis, and Christos D. Zaroliagis. *The Fourth Moment in Luby's Distribution*. February 1995. 10 pp. To appear in *Theoretical Computer Science*.
 - 14 Devdatt P. Dubhashi. *Inclusion–Exclusion(3) Implies Inclusion–Exclusion(n)*. February 1995. 6 pp.
 - 13 Torben Braüner. *The Girard Translation Extended with Recursion*. February 1995. 79 pp. Full version of paper to appear in Proceedings of CSL '94, LNCS 933, 1995.
 - 12 Gerth Stølting Brodal. *Fast Meldable Priority Queues*. February 1995. 12 pp.
 - 11 Alberto Apostolico and Dany Breslauer. *An Optimal $O(\log \log n)$ Time Parallel Algorithm for Detecting all Squares in a String*. February 1995. 18 pp. To appear in *SIAM Journal on Computing*.

- 10 Dany Breslauer and Devdatt P. Dubhashi. *Transforming Comparison Model Lower Bounds to the Parallel-Random-Access-Machine*. February 1995. 11 pp.
- 9 Lars R. Knudsen. *Partial and Higher Order Differentials and Applications to the DES*. February 1995. 24 pp.
- 8 Ole I. Hougaard, Michael I. Schwartzbach, and Hosein Askari. *Type Inference of Turbo Pascal*. February 1995. 19 pp.
- 7 David A. Basin and Nils Klarlund. *Hardware Verification using Monadic Second-Order Logic*. January 1995. 13 pp.
- 6 Igor Walukiewicz. *A Complete Deductive System for the μ -Calculus*. January 1995. 39 pp.
- 5 Luca Aceto and Anna Ingólfssdóttir. *A Complete Equational Axiomatization for Prefix Iteration with Silent Steps*. January 1995. 27 pp.
- 4 Mogens Nielsen and Glynn Winskel. *Petri Nets and Bisimulations*. January 1995. 36 pp. To appear in TCS.
- 3 Anna Ingólfssdóttir. *A Semantic Theory for Value-Passing Processes, Late Approach, Part I: A Denotational Model and Its Complete Axiomatization*. January 1995. 37 pp.
- 2 François Laroussinie, Kim G. Larsen, and Carsten Weise. *From Timed Automata to Logic - and Back*. January 1995. 21 pp.
- 1 Gudmund Skovbjerg Frandsen, Thore Husfeldt, Peter Bro Miltersen, Theis Rauhe, and Søren Skyum. *Dynamic Algorithms for the Dyck Languages*. January 1995. 21 pp. To appear in *Algorithms and Data Structures: 4th Workshop, WADS '95 Proceedings, LNCS, 1995*.

BRICS Notes Series for 1995

ISSN 0909-3206

- 4 Yuri Gurevich and Egon Börger. *Evolving Algebras. Mini-Course*. July 1995. iv+222 pp.
- 3 Andrew D. Gordon. *Bisimilarity as a Theory of Functional Programming. Mini-Course*. July 1995. iv+59 pp.
- 2 Uffe H. Engberg, Kim G. Larsen, and Arne

Skou, editors. *Proceedings of the Workshop on Tools and Algorithms for The Construction and Analysis of Systems, TACAS (Aarhus, Denmark, 19-20 May, 1995), May 1995*. vi+334 pp.

- 1 Igor Walukiewicz. *Notes on the Propositional μ -calculus: Completeness and Related Results*. February 1995. 54 pp.

BRICS Lecture Series for 1995

ISSN 1395-2048

- 3 Michael I. Schwartzbach. *Polymorphic Type Inference*. June 1995. viii+24 pp.

Abstract:

We will present a tiny functional language and gradually enrich its type system. We shall cover the basic Curry-Hindley system and Wand's constraint-based algorithm for monomorphic type inference; briefly observe the Curry-Howard isomorphism and notice that logical formalism may serve as the inspiration for new type rules; present the poly-

morphic Milner system and the Damas-Milner algorithm for polymorphic type inference; see the Milner-Mycroft system for polymorphic recursion; and sketch the development of higher type systems. We will touch upon the relationship between types and logic and show how rules from logic may give inspiration for new type rules. En route we shall encounter the curious discovery that two algorithmic problems for type systems, which have been implemented in popular programming languages,

have turned out to be respectively complete for exponential time and undecidable.

Contents

- 1 Type Checking and Type Inference
 - 2 A Tiny Functional Language
 - 3 A Simple Type System
 - 4 Simple Type Inference
 - 5 Types and Logic
 - 6 Polymorphic Types
 - 7 Polymorphic Type Inference
 - 8 Polymorphism and Recursion
 - 9 Higher Type Systems
 - 10 Problems
- 2 Sven Skyum. *Introduction to Parallel Algorithms*. June 1995. viii+16 pp.

Abstract:

The material in this note is used as an introduction to parallel algorithms in a third year course on algorithms and complexity theory in Aarhus. Basic data structures and algorithms including sorting and searching are introduced to the students the first year. For the analysis of algorithms the unit cost model was used. The RAM were not introduced, the analysis was based on the number of operations in a (programming) language and corresponds to unit cost at a RAM.

This note covers an introduction to various PRAM's, presents Brents scheduling principle and various algorithms such as prefix, merging and sorting building up to a general method of simulating a CRCW-PRAM on a EREW-PRAM.

Two weeks are spent on the subject which corresponds to a total of six 45 minutes lectures.

The first version of the note was written in 1993 and was inspired by a note written by Peter Bro Miltersen the year before. Some of the problems originate from it. The note has since undergone revisions each year. Some of them have been substantial.

Contents

- 1 Models
 - 2 Time, work and optimality
 - 2.1 Brent's scheduling principle
 - 3 Merging and Sorting
 - 3.1 Prefix computations
 - 3.2 Merging on a CRCW-PRAM
 - 3.3 Bucketsort on an EREW-PRAM
 - 4 Simulation of CRCW-PRAMs
 - 5 Problems
- 1 Jaap van Oosten. *Basic Category Theory*. January 1995. vi+75 pp.

Abstract:

This course was given to advanced undergraduate and beginning Ph.D. students in the fall of 1994 in Aarhus, as part of Glynn Winskel's semantics course. It is, in the author's view, the very minimum of category theory one needs to know if one is going to use it sensibly. Nevertheless, two topics are breathed on, which may be skipped: there is a glimpse of categorical logic, and there is a treatment of the λ -calculus in cartesian closed categories. These are there to give the reader at least a very rough idea of how the theory "works". The text contains a bit over hundred exercises, varying in difficulty, which supplement the treatment and are warmly recommended. There is an elaborate index.

Contents

- 1 Categories and Functors
 - 1.1 Definitions and examples
 - 1.2 Some special objects and arrows
- 2 Natural transformations
 - 2.1 The Yoneda lemma
 - 2.2 Examples of natural transformations
 - 2.3 Equivalence of categories; an example
- 3 (Co)cones and (co)limits

- 3.1 Limits
- 3.2 Limits by products and equalizers
- 3.3 Colimits
- 4 A little piece of categorical logic
 - 4.1 Regular categories and subobjects
 - 4.2 Coherent logic in regular categories
 - 4.3 The language $\mathcal{L}(\mathcal{C})$ and theory $T(\mathcal{C})$ associated to a regular category \mathcal{C}
 - 4.4 Example of a regular category
- 5 Adjunctions
 - 5.1 Adjoint functors
 - 5.2 Expressing (co)completeness by existence of adjoints; preservation of (co)limits by adjoint functors

- 6 Monads and Algebras
 - 6.1 Algebras for a monad
 - 6.2 T -Algebras at least as complete as \mathcal{D}
 - 6.3 The Kleisli category of a monad
- 7 Cartesian closed categories and the λ -calculus
 - 7.1 Cartesian closed categories (ccc's); examples and basic facts
 - 7.2 Typed λ -calculus and cartesian closed categories
 - 7.3 Representation of primitive recursive functions in ccc's with natural numbers object ☰

Technical Contributions

A General Splitting Lemma

by Devdatt Dubhashi⁴ and Desh Ranjan^{5 6}

In a paper related to combinatorial games Spencer [1] proved the following amusing lemma:

Lemma 1 (Splitting Lemma) *Let $x_1 \geq x_2 \geq \dots \geq x_r$ all be negative powers of 2 with sum $x_1 + \dots + x_r = 1$. Then there exists a partition of the x_i into two groups so that each group sums to precisely one half.*

In this note, we prove a more general version of this statement that is in some way the best one can hope to prove.

For reals x, y we shall say that x *divides* y , denoted $x|y$ if $y = ax$ for a positive integer a .

Lemma 2 (General Splitting Lemma) *Let $y_1 \geq y_2 \geq \dots \geq y_r > 0$ be any reals with sum $y_1 + \dots + y_r = S$ such that for all $i < r$, $y_{i+1}|y_i$ and $y_1|S$. Then there exists a partition of the y_i into two groups so that the sums of the groups differ by*

no more than y_r . Moreover, there exists a partition of the y_i into two groups so that each group sums to precisely $S/2$ iff S/y_r is even.

Proof. Essentially the argument of Spencer works for the General Splitting Lemma as well. Consider the following process: place the y_i into groups largest first, always placing the y_i into the group with currently smaller sum. We show that this process maintains the invariant that after placing $y_1 \dots y_l$ the absolute value of the difference of the sums is at most $(y_{l+1} + \dots + y_r) + y_r$. At the start, the invariant holds trivially. Assume that it holds after $y_1 \dots y_l$ have been placed. We show that the invariant holds after placing y_{l+1} . Let Δ_l denote the absolute value of the difference of the sums after $y_1 \dots y_l$ have been placed. Then by the induction hypothesis, $\Delta_l \leq (y_{l+1} + \dots + y_r) + y_r$. Now consider the two cases:

Case 1: The two sums are different. Then as y_1, \dots, y_l are all multiples of y_{l+1} the difference is a multiple of y_{l+1} . Therefore, if

⁴BRICS, dubhashi@daimi.aau.dk

⁵Work done while the author was visiting the Max-Planck-Institut für Informatik, and BRICS, University of Aarhus.

⁶Dept. of Comp. Sci., New Mexico State University, Las Cruces, New Mexico 88003, USA, dranjan@cs.nmsu.edu

$\Delta_l = ky_{l+1}$ then $\Delta_{l+1} = (k-1)y_{l+1}$ which implies that $\Delta_{l+1} \leq \Delta_l - y_{l+1} = (y_{l+2} \cdots y_r) + y_r$.

Case 2: The two sums are equal. In this case $\Delta_{l+1} = y_{l+1}$. Then, $y_{l+2} \cdots y_r = S - (y_1 + \cdots + y_{l+1})$ is divisible by y_{l+1} . This can be zero only if $l+1 = r$ in which case the invariant is true as $\Delta_r = y_r$. Otherwise, it is at least y_{l+1} and hence $\Delta_{l+1} \leq y_{l+2} + \cdots + y_r$ and the invariant is maintained.

Observe that at the end, the sums in both parts are divisible by y_r , hence so is their difference. Thus the difference is either 0 or y_r . If the difference is in fact y_r , then the sum $S = (2a+1)y_r$ for some positive integer a , so S/y_r is odd. Hence if S/y_r is even, then the difference must be 0. Finally, if S/y_r is odd then there cannot be an equal split. For this, note that for any $I \subseteq [n]$, $\sum_{i \in I} y_i$ is divisible by y_r . Therefore if $\sum_{i \in I} y_i = \sum_{i \in \bar{I}} y_i = ay_r$ for a positive integer a , then $S = \sum_{i \in [n]} y_i = 2ay_r$, that is S/y_r is even. ■

Remark 3 The Splitting Lemma of Spencer follows from the above by choosing $y_i = x_i$ and $S = 1$.

Remark 4 The General Splitting Lemma is optimal in the sense that if S/y_r is odd, then a split into equal parts is impossible.

References

- [1] J. Spencer, “Randomization, derandomization and antirandomization: three games”, *Theoretical Computer Science*, 131, pp. 415–429, 1994. ■■

UPPAAL — A Tool Suite for Verification of Real-Time Systems

by Kim G. Larsen

During the spring of 1995 a new automatic verification tool was presented to the research community as a result of intense research collaboration

between BRICS and Department of Computing Systems at Uppsala University. The new tool is called UPPAAL (for Uppsala and Aalborg) and is seriously contesting existing similar tools.

Background

For numerous practical systems the most important and critical aspect is that the services offered by the system are provided at the *right* moments in time. By their very nature, such real-time systems are not adequately described using the classical model of finite-state systems: additional information about timing-constraints is required. A by now well-established timed extension of finite-state systems (so-called timed automata) was introduced in 1990 by Alur and Dill, and in recent years efficient automatic verification algorithms have been obtained based on an accompanying symbolic technique (known as the state-region graph technique). However these algorithms are all faced with a potential explosion in the (symbolic) state-space occurring when considering networks of systems.

What is Uppaal

UPPAAL is a new tool for automatic verification of safety and bounded liveness properties of networks of timed automata implemented in C^{++} . The current version of UPPAAL deals with the above mentioned explosion problem by a new and coarser symbolic technique reducing the verification problem to that of solving simple linear constraint systems. Future versions of UPPAAL will integrate a similarly newly developed compositional technique.

UPPAAL contains a suit of tools and features including:

- A graphical interphase allowing networks of timed automata to be defined by drawing.
- An automatic compilation of the graphical definition into a textual format, a format which also serves as a basic programming

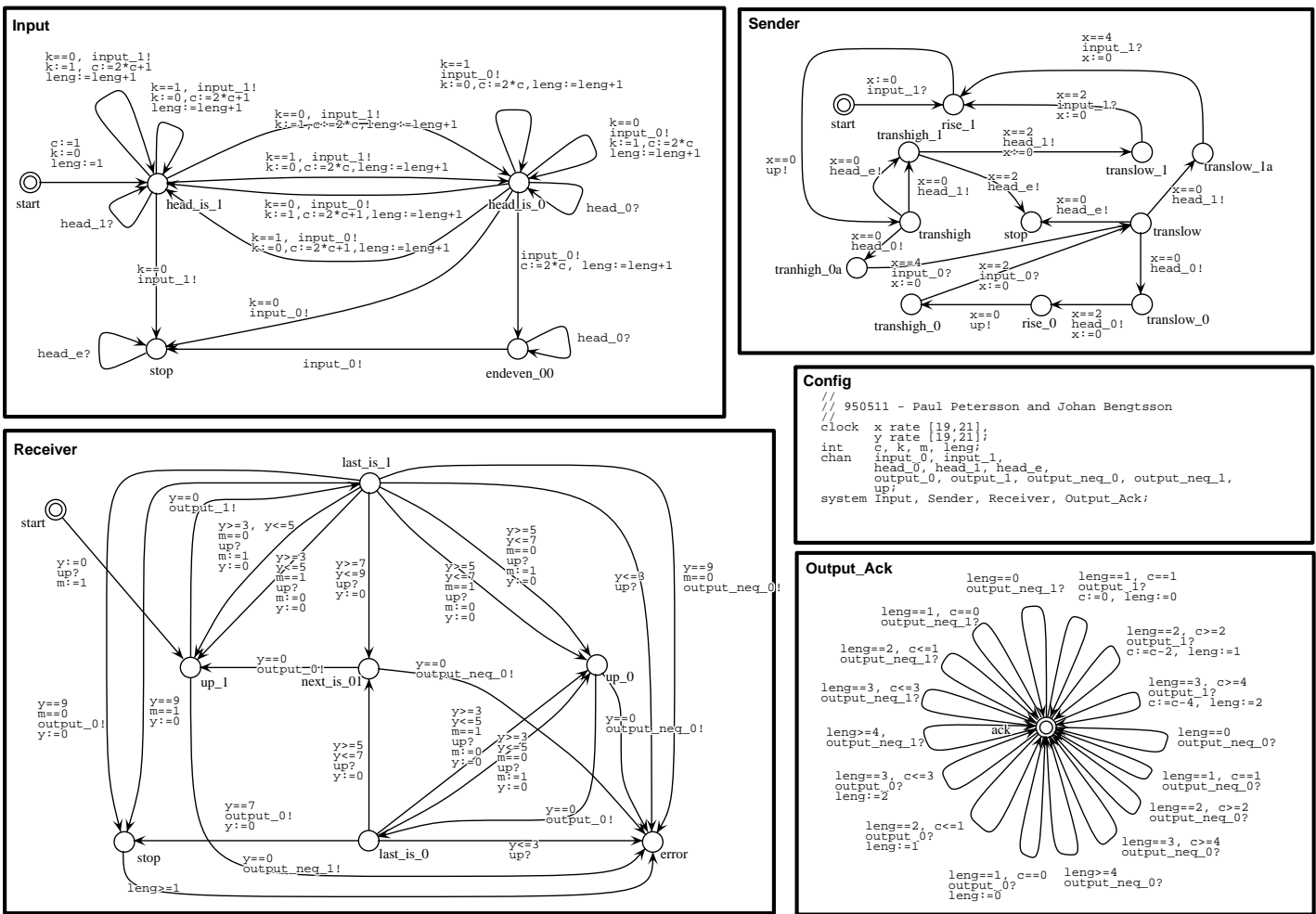


Figure 1: The Audio Protocol

language for timed automata. The automatic compilation of the graphical description ensures the important principle that "what you see is what you verify".

- UPPAAL is actually able to deal with certain types of so-called hybrid automata. In particular UPPAAL allows network of timed automata with varying time-speed. These hybrid automata are automatically compiled into equivalent timed automata, again to ensure the "what you see is what you verify" principle. To give an impression of the type of systems UPPAAL can deal with, Figure 1 provides the graphical description of an Audio Protocol. The verification time of UPPAAL was 30sec.
- A number of simple, but in practice extremely useful syntactical checks are made

before verification can commence.

- In case verification of a particular real-time system fails (which happens more often than not), a diagnostic trace is automatically reported by UPPAAL in order to facilitate debugging.

Who is working with UPPAAL

The UPPAAL tool kit is developed in collaboration between BRICS, The Centre of Basic Research in Computer Science, Aalborg University, Denmark and DoCS, The Department of Computer Systems, Uppsala University, Sweden. The people involved with the development are Wang Yi (Ph.D., Lecturer, DoCS), Kim G. Larsen (Professor, BRICS), Paul Pettersson (Ph.D. Student, DoCS), Arne Skou (Ph.D., Lecturer, BRICS), Kåre

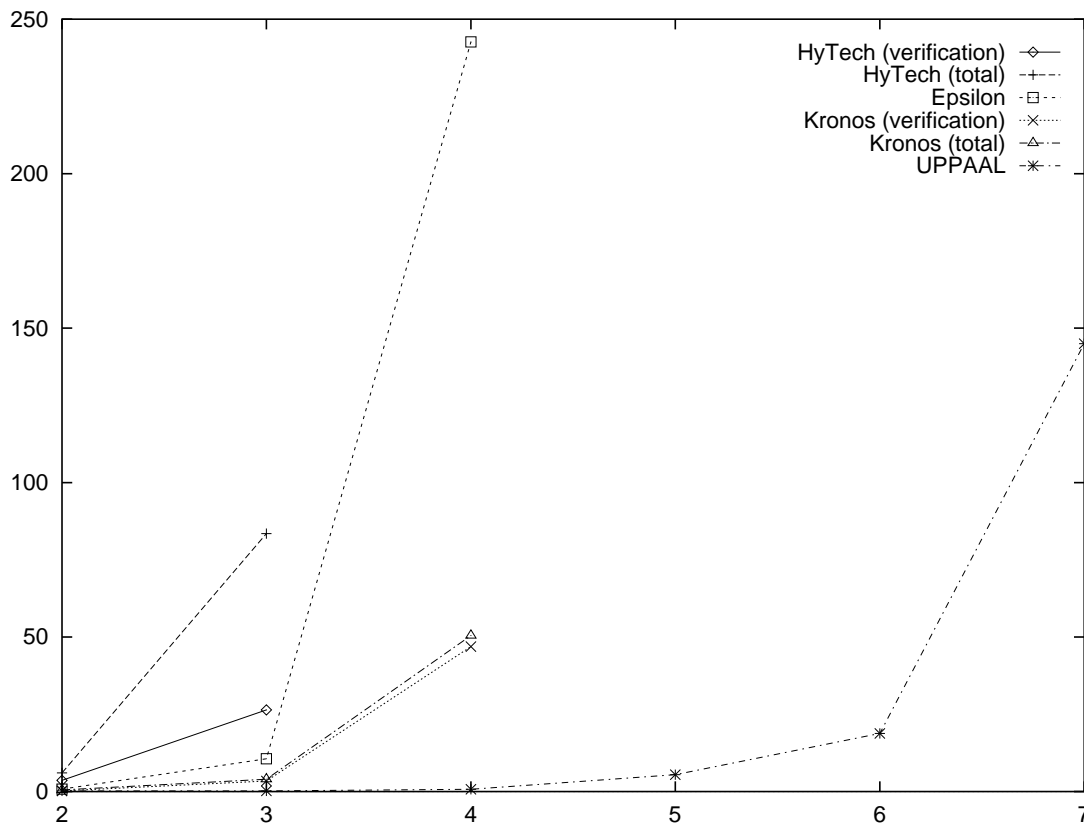


Figure 2: Comparison — Execution Times (seconds)

Kristoffersen (Ph.D. Student, BRICS), Francois Laroussinie (Ph.D. Researcher, BRICS), Johan Bengtsson (M.Sc. Student, DoCS) and Fredrik Larsson (M.Sc. Student, DoCS).

How good is UPPAAL

The current version of UPPAAL has been tested on the verification of Fischer's Protocol and compared with the performance of three other existing real-time verification tools. As can be seen from Figure 2 the experimental results show that UPPAAL is not only substantially faster than the other tools but also able to handle much larger systems (the unit on the x -axis is the number of components considered in the protocol).

How to learn more about UPPAAL

UPPAAL has a WWW homepage⁷, containing pointers to the published material on UPPAAL

and its theoretical foundation as well as complete information for installation.

The future of UPPAAL

The collaboration between BRICS and DoCS on UPPAAL continues and includes a joint tutorial on UPPAAL at the 16th IEEE Real-Time Systems Symposium, Pisa, Italy, 5–7 December this year. Our future plans for collaboration also includes frequent exchange of Ph.D. students as an important feature.

The funding of UPPAAL

The work on UPPAAL has been partly supported by the BRA project CONCUR2, NUTEK (The Swedish Board for Technical Development) and TFR (Swedish Technical Research Council).

⁷<http://www.docs.uu.se/docs/rtmv/uppaal>

Calendar of Events

Date	Event
Week 32 Aug	<i>Yuri Gurevich</i> , University of Michigan, and <i>Egon Börger</i> , University of Pisa; Evolving Algebras.
Week 33 Aug	<i>Robert Paige</i> , Courant Institute, New York University: Analysis and Transformation of Set-Theoretic Languages.
Week 34 Aug	<i>Michael Ben-Or</i> , Hebrew University, Jerusalem; lectures on: Secure Multiparty Computations.
Sep	<i>Richard Tan</i> , University of Utrecht and university of Oklahoma; Compact Routing Methods in Parallel and Distributed Systems.
1997	CSL '97 (Computer Science Logic).

BRICS Address and World Wide Web

For further information, hard copies of information material and reports of the BRICS Series as well as enrolment into the BRICS newsgroup, please contact **BRICS** at

Telephone: +45 8942 3360

Telefax: +45 8942 3255

Internet: <BRICS@daimi.aau.dk>

or, in writing, at

Department of Computer Science
University of Aarhus
Ny Munkegade, building 540
DK - 8000 Aarhus C
Denmark.

You can get access to information from BRICS through World Wide Web (WWW) and anonymous FTP. To connect to the BRICS WWW entry, open the URL:

<http://www.daimi.aau.dk/BRICS/>

The BRICS WWW entry contains updated information about most of the topics covered in the newsletter as well as access to electronic copies of information material and reports of the BRICS Series (look under Publications).

To access the information material and reports of the BRICS Series via anonymous FTP do the following:

```
ftp ftp.daimi.aau.dk
```

```
cd pub/BRICS
```

```
get README.
```



BRICS Newsletter

ISSN 0909-6043

Editors: Glynn Winskel & Uffe H. Engberg

Lay-out: Uffe H. Engberg

Publisher: BRICS

Print: Institute of Mathematics
University of Aarhus

© BRICS 1994