Report on the British Colloquium for Theoretical Computer Science (BCTCS17) April 9-12 2001, Glasgow, United Kingdom

Stephan Reiff-Marganiec*

The 17th Annual meeting of the British Colloquium for Theoretical Computer Science (BCTCS) was held from April 9th till April 12th at the Kelvin Conference Centre in Glasgow. The event was hosted by the Department of Computing Science, University of Glasgow supported by EPSRC, Microsoft Research and Hewlett Packard Labs (Bristol).

Vice Principal Professor Malcolm McLeod formally opened the meeting. The colloquium featured and interesting and wide ranging programme of invited and contributed talks. Invited talks were presented by Muffy Calder (University of Glasgow), Ursula Martin (University of St. Andrews), Faron Moller (University of Wales, Swansea), Joachim Parrow (KTH Teleinformatik, Sweden), Mike Paterson (University of Warwick), Simon Peyton-Jones (Microsoft UK) and Alexander Rabinovich (University of Edinburgh). The invited programme was complemented by 32 contributed talks.

The 18th British Colloquium for Theoretical Computer Science (BCTCS2002) will be held at Hewlett Packard Labs, Bristol from 7-10 April 2002. Anyone wishing to contribute talks concerning Theoretical Computer Science is warmly invited to do so. Further information regarding BCTCS2002 can be found at
`http://www-uk.hpl.hp.com/BCTCS18` The BCTCS webpages, giving details of previous colloquia are located at
`http://www.csc.liv.ac.uk/ ped/bctcs/summary.html`

1 Invited Talks

Prof Muffy Calder
University of Glasgow
A Day in the Life of a Spin Doctor

The design of any concurrent system, particularly one with synchronous and/or asynchronous communication, can be very difficult. What kind of investigations should you carry out to ensure that you have the got the system you intended, that what you intended is reasonable, and that you will get an answer within a reasonable period of time?

In this talk I will examine these questions within the context of G. Holzmann's Promela language and the model-checker Spin. Promela is a C-like language that supports synchronous and asynchronous communication, concurrency, and dynamic communication; Spin supports checking generic and domain specific properties, the latter as linear temporal logic. I wil survey some of the more interesting aspects of both with special consideration to practical tips for making reasoning tractable.

*Department of Computing Science, University of Glasgow, Glasgow G12 8RZ, Scotland, sreiff@acm.org

Ursula Martin
University of St. Andrews
Computational math: the new challenge for computational logic

Scientific computation based on continuous computational mathematics is widely used for applications as diverse as high energy physics, weather forecasting, environmental and financial modelling, and is becoming important in systems design for engineering applications like air-traffic control algorithms or synthesising code for control systems. Complex models and simulations are among the most demanding of computer power. The leading commercial software systems such as Maple 6, MATLAB and NAG have around 2 million users, and scientists and engineers have in-house code for many specialised applications.

Computational mathematics systems may be numeric or symbolic: both can be subject to error and problems of scale, and lack a framework for reasoning about the models they implement. Computational logic systems allow such reasoning, and do the underlying mathematics analytically, so we can rely on the answers they produce.

We give an overview of recent work at St Andrews and elsewhere in applying computational logic to computational mathematics, and set out some challenges:

- for its immediate application as an effective component of mathematical software such as Maple or for representing mathematical knowledge as part of endeavours like OpenMath/MathML
- for the use of computational logic techniques to support applied math and mathematical modeling, particularly in areas where the highest degreé of assurance is required, such as avionics
- for developing a strategy for the representation, validation and communication of mathematics and science in a future where the opportunities and challenges of digitally embodied knowledge may radically change scholarship and scientific discourse

Faron Moller
University of Wales Swansea
Techniques for decidability and undecidability for bisimilarity

In this tutorial we describe general approaches to deciding bisimilarity between vertices of (infinite) directed edge-labelled graphs. The approaches are based on a systematic search following the definition of bisimilarity. We outline (in decreasing levels of detail) how the search is modified to solve the problem for finite graphs, BPP graphs, BPA graphs, normed PA graphs, and normed PDA graphs. We complete this by showing the technique used in the case of graphs generated by one-counter machines. Finally, we demonstrate a general reduction strategy for proving undecidability, which we apply in the case of graphs generated by state-extended BPP (a restricted form of labelled Petri nets).

Joachim Parrow
KTH Teleinformatik
Tutorial: An introduction to the pi-calculus

The pi-calculus is a process algebra where agents communicate by sending communication ports, or capabililtes, to each other. In this introduction I shall present some of the basic ideas and intutions and give a small modelling example from mobile telephony

Recommended reading: "http://www.it.kth.se/ joachim/intro.ps"

Mike Paterson
University of Warwick
Getting your message across, but nicely : an introduction to contention resolution

When independent processes compete for a limited resource, such as a communication channel, interesting combinatorial and probabilistic questions are raised. Some old and some new results on contention resolution protocols will be described, though major open problems remaining.

Simon Peyton-Jones
Microsoft Research
Asynchronous Exceptions in Concurrent Haskell

There are some applications for which the ability to interrupt a computation cleanly is essential. Timeouts and user interrupts are two common examples. Recovering from such asynchronous exceptions is problematic, because they may interrupt the computation at a bad moment. So problematic, in fact, that most languages do not support asynchronous exceptions at all; instead the application must poll an alert flag at regular intervals.

In my talk I will present a design for asynchronous exceptions in Concurrent Haskell. Unusually, we have a complete operational semantics for the language, including the asynchronous-exception part. My hope is that Colloquium participants may be inspired to develop a usable theory for this semantics.

Alexander Rabinovich
University of Edinburgh
Temporal Logic over Branching Time: Expressiveness and Complexity

Many temporal logics were suggested as branching time specification formalisms during the last 20 years. These logics were compared against each other for their expressive power, model checking complexity and succinctness. Yet, unlike the case for linear time logics, no canonical temporal logic of branching time was agreed upon.

In this talk we offer an explanation for the multiplicity of temporal logics over branching time and provide an objective quantified 'yardstick' to measure these logics.

2 Contributed Talks

The speakers names are given in italics. The quoted institute is that of the speaker.

Richard Bruce
University of Leicester
Introduction to ad hoc wireless networks and a new fan-out broadcasting method

This talk shall introduce the relatively new area of mobile ad hoc networks (MANET). The talk will briefly review some of the current literature in this area such as unicasting and multicasting routing, discuss a new protocol aimed at fan-out multicast operations and finally present our simulated results.

Marius C. Bujorianu and *Manuela L. Bujorianu*
Constructive Foundations of Stochastic Analysis in Linear Logic

Linear logic (LL) is a constructive Logic invented by G.Y.Girard which has been applied to various domains in computer science, for example functional programming and concurrent systems. Some powerful implementations of LL are currently available. In this paper we propose a new application of LL namely in constructive mathematics. We propose a constructive theory for a very important branch of mathematics like Stochastic Analysis and especially for its core, Potential Theory (PT). PT theory has a long history, originated by the work of Gauss. Recent applications of PT include

Markov Processes, Financial Mathematics, Linear and Non-linear Partial Differential Equations, Stochastic Differential Equations, Dynamic Systems and Stochastic Differential Geometry. In this way we obtain a constructive logical approach to all these mathematical theories. We use a modern axiomatization of PT called Dirichlet Forms and the phase semantics of LL as a common semantic framework for both LL and PT. An important feature of our approach is that the stochastic and the deterministic cases are treated in a single elegant framework. We provide a sound and rich semantic domain for the integration of theorem proving with Constructive Mathematics and Computer Assisted Algebra. Possible further developments are investigated, especially the use of Girard's Geometry of Interaction as a unified model for the discrete and continuous computations and the extensions to the non-commutative case.

Marius C. Bujorianu and Manuela L. Bujorianu
University of Kent
An Abstract Domain for Probabilistic Hybrid Systems

Hybrid Systems (HS) incorporate both discrete and continuous dynamics; HS are everywhere: they arise in air traffic control, automobiles, robotics, consumer electronics (e.g. VCR, microwave oven, heater...). We are particularly interested in the biological systems. Interest in HS has grown in recent years, mainly because of their direct relation with embedded systems (systems that interact with the continuously changing real world). These systems often arise in safety critical situations, so formal verification methods have been the subject of intense research. Different models of HS are extensively investigated. The continuous aspects of these models require incursions into differential equations, which have very little in common with existing tools of automata theory and logic. Abstract interpretation of the continuous environment could help in reasoning about the system properties. This is why we propose an abstract domain in order to obtain a algebraic characterisation of the continuous part of an hybrid system. The abstract domain is powerful enough to deal with stochastic features. The discrete situations can be easily obtained as a particular case. The dynamics of the system are modelled as causal relations and the (possible) concurrent aspects are expressed as partial orders. We introduce and formalise a new interaction operator, named superposition, between multiple environments. The possible biological applications are stressed. For this we use the classical potential theory as a intermediate level. A case study in cardiac electrogenesis is detailed presented.

Jonathan Burton, Maciej Koutny and Guiseppe Pappalardo
University of Newcastle upon Tyne
Implementation Relations in the Event of Interface Difference

Standard process algebraic equivalences or refinements assume that both the implementation and specification systems are expressed at the same level of abstraction or using the same alphabet. However, it may often be useful to relate processes which are expressed at different levels of abstraction. This is the case when developing a system stepwise, where a specification is refined into a more concrete process at a lower level of detail; in particular, it has relevance to the modelling of fault tolerance. Moreover, dealing with interface difference in this way allows us to model check refinement or equivalence compositionally, even when the respective component specification and implementation processes have different interfaces. We describe some of the issues involved in this area and describe work done in the context of CSP to solve this problem.

Denham Coates-Evelyn
Kings College, University of London
An analysis of Kronrod's and related merge algorithms

We consider the problem of merging two sorted lists of m and n keys each in-place. We develop algorithms that uses (1) Optimum comparisons and no more than $3(n+m)+o(n+m)$ data moves for unstable merge, and (2) Optimum comparisons and no more than $(5.5n + 7.5m + o(n + m)$

data moves for stable merge. Implementation of our algorithm is stable and less complex than that given by Geffert et al who have derived an unstable algorithm that does optimum cmparisons and $3(n+m)+o(m)$ data moves and a stable algorithm that does optimum comparisons and $5n+12m+o(m)$ data moves. We then show that with the use of further constant memory k we can do modification to the algorithm to achieve optimum comparisons and $2(n+m)+3n/k+o \quad (m)$ data moves for stable merge. We use our main result to implement a stable merge-sort that does $(1+\frac{\lg(k)}{k})N\lg(N)-(N-1)$ comparisons and $(2+\frac{3}{2k})N\lg(N)$ data moves to sort a list of N data values in-place.

Denham Coates-Evelyn
Kings College, University of London
An optimum in-place Merge

In this report we consider the problem of merging two sorted lists of m and n keys each in-place. We survey known techniquesfor this problem, focussing on correctness and the attributes of stability and practicality. We demonstrate a class of unstable in-place merge algorithms that actually does not merge in the presence of sufficient duplicate keys of a given value. We show four (4) relatively simple block sorting techniques that can be used to correct these algorithms. In addition, we show relatively simple and robust techniques that does stable local block merge followed by stable block sort to create a merge. Using block size of $O(\sqrt{n+m})$ we achieve complexity of no more than $1.5(n+m)+O(\sqrt{n+m}\lg(n+m))$ comparisons and $4(n+m)+O(\sqrt{n+m}\lg(n+m))$ data moves. Using block size of $O((n+m)/\lg(n+m))$ gives us complexity of no more than $n+m+O((n+m)(1-\frac{\lg(\lg(n+m))}{\lg(n+m)})$ comparisons and $5(n+m)+O((n+m)(1-\frac{(\lg(\lg(n+m))}{\lg(n+m)})$ moves. Our algorithm is stable except for the case of buffer permutation during the merge, its implementation is much less complex than that given by V. Geffert et al who have derived an unstable algorithm that does optimum comparisons and $3(n+m)+o(n)$ data moves.

A.Cherubini, *S.Crespi Reghizzi*, M.Pradella, P.San Pietr
Politecnico di Milano
Associative Language Descriptions versus Context-Free models

The generative capacity of Context-Free grammars, notoriously insufficient for computer (or natural) languages, is also misdirected towards languages characterised by unnatural mathematical properties. The recent ALD model [1,2] combines locally testable and constituent structure ideas into a model simpler than CF yet adequate for programming languages (e.g. Pascal in [3]). ALD is a nonterminal-free formalism as regular expressions or Marcus' Contextual Grammars.

Consider a set of trees with terminal labelled leaves (the terminal string) and unlabeled internal nodes. Let k be an integer "degree". Simply stated the model first classifies trees into classes ("patterns") characterised by the root siblings string. Then for a subtree in a pattern class a "context" of occurrence is specified by two terminal strings of length k: the ones immediately preceding and following the frontier of the subtree. A tree is in the ALD iff any subtree meets the context specification.

ALDs are strictly included in the CFs, yet they contain Greibach's hardest CF language. ALDs make a strict hierarchy ordered by degree. Since ALD trees are included into the Non-Counting parenthesized CF languages [4], unnatural mathematical sets based on counting are excluded. Inclusion of regular languages by ALDs is open. Present research focuses on structural properties, e.g. that any CF language is the alphabetical non-erasing homomorphism of an ALD (which can be interpreted as a marked-up or tagged encoding) and on deterministic parsing. ALDs originated from search for a syntax model consistent with current views on brain organisation [5].

References:

[1] A. Cherubini, S. Crespi Reghizzi, P. San Pietro, Languages Based on Structural Local Testability, in C. S. Calude and M.J. Dinnen (eds.), Combinatorics, Computation and Logic, Proc. DMTCS99, 1999, pp. 159-174, Springer-Verlag.

[2] A. Cherubini, S. Crespi Reghizzi, P. San Pietro, "Associative Language Descriptions, TCS, 2001.

[3] S. Crespi Reghizzi, M. Pradella and P.L. San Pietro, "Associative definitions of programming languages", Computer Languages, 27, 2001.

[4] S. Crespi-Reghizzi, G. Guida and D. Mandrioli, Non-counting context-free languages, JACM, 25, 1978, 4, 571-580.

[5] S. Crespi-Reghizzi and V. Braitenberg, 'Towards a brain compatible theory of syntax based on local testability' in C. Martin-Vide and V. Mitrana (eds) Grammars and Automata for String Processing: from Mathematics and Computer Science to Biology, and Back. Gordon and Breach, London, 2001.

Graham Farr and Keith Edwards
Monash University

Planarization and fragmentability for graphs of bounded degree

We describe an algorithm which, for a graph with n vertices and maximum degree d, finds a set of at most n (d-2)/(d+1) vertices whose removal leaves behind a planar graph. (This is thus a heuristic for the Maximum Induced Planar Subgraph problem.) For $d \leq 3$, this performance guarantee is, in a sense, best possible. We also mention connections with the notion of fragmentability, which measures how easily graphs in some class can be broken up into small (bounded size) pieces by removal of vertices.

M. Carmen Fernandez-Gago
University of Liverpool

Algorithms for Guiding Clausal Temporal Resolution

Temporal logic is a variety of non-classical logic used in a range of areas within Computer Science and Artifcial Intelligence. Consequently, different proof methods have been developed, implemented and applied.

Here we use a proof method for temporal logics based upon the resolution. The resolution procedure is characterised by the translation to a normal form, classical resolution on formulae that occur at the same moment in time (step resolution), and temporal resolution between states. Although the clausal temporal resolution method has been defined, proved correct and implemented it sometimes generates an unneccesarily large set of formulas that may be irrelevant to the refutation. Not only that, but temporal resolution operations occur only after all step resolution inferences have been carried out. This means that cases where a large amount of step resolution can occur the method may be very expensive. The temporal resolution operation requires the detection of a set of clauses which satisfy a specified condition. As the search for candidates for this operation is likely to be required, our intention is to try to avoid all unnecessary step resolution operations, and apply temporal resolution earlier. In this sense we propose an algorithm based on the outputs of a refined temporal operation to guide the search for such candidates.

Simon Gay
University of Glasgow

A Framework for the Formalisation of Pi Calculus Type Systems in Isabelle/HOL

We present a formalisation, in the theorem proving system Isabelle/HOL, of a linear type system for the pi calculus, including a proof of runtime safety of typed processes. The use of a uniform encoding of pi calculus syntax in a meta language, the development of a general theory of type environments, and the structured formalisation of the main proofs, facilitate the adaptation of the Isabelle theories and proof scripts to variations on the language and other type systems.

David Haniff
University of Birmingham
Augmenting Reality

In 1961 Douglas C. Engelbart at Stanford University proposed research into *Augmented Human Intellect.* Augmented Reality enhances the intellect of the individual by providing virtual annotations through our perceptual systems (e.g. head-mounted display or head-phones). However, in order to truly supply 'augmentation' the features of the augmented reality systems need to be carefully designed. Formal methods can provide a means to provide a detailed description of augmented reality components to identify potential clashes. For example, many augmented reality systems are implemented on wearable computers situated in various environmental conditions, these conditions may conflict with the form of passive (sensor-based) input. A basic image recognition system will not be suitable under certain lighting conditions or electro-magnetic interference could lead to inaccuate sensor readings; intelligent sensor systems may therefore be required. Furthermore, the user-interface and the interpretation of the information presented to them need rigorous treatment to *enhance* and not detract from reality. *Intellect* presupposes cognitive functioning. A formal approach can be used to describe the intellectual process of completing a task as well modelling the concurrent system constraints. The technology itself is also subject to problems due to system lag and registration of virtual objects with real world objects. All of these factors need to be taken into account in the design of an augmented reality system that provide *augmented human intellect.*

Graham Hutton
University of Nottingham
The Generic Approximation Lemma

The approximation lemma is a simplification of the well-known take lemma, and is used to prove properties of programs that produce lists of values. In this talk I will show how the approximation lemma, unlike the take lemma, can naturally be generalised from lists to a large class of datatypes, and present a generic approximation lemma that is parametric in the datatype to which it applies.

Robert W. Irving and *David F. Manlove*
University of Glasgow
The Stable Roommates Problem with Ties

An instance of the well-known Stable Roommates problem (SR) involves a set P of 2n participants, and each participant ranks the others in strict order of preference. A solution is a stable matching M, i.e. a partition of P into n disjoint pairs, such that there is no pair x,y, each of whom prefers the other to his/her partner in M. In 1985, Irving formulated a linear-time algorithm for the problem of finding a stable matching if one exists, given an instance of SR.

In this talk, we consider the variant of SR in which participants are permitted to express ties in their preference lists. The problem in this setting has been studied under two possible stability criteria: so-called super-stability and weak stability. We present a linear-time algorithm for finding a super-stable matching if one exists, given a Stable Roommates instance with ties. This contrasts with the known NP-hardness of the analogous problem under weak stability.

Mark Jerrum, **Alistair Sinclair and Eric Vigoda**
University of Edinburgh
A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries

The permanent of a matrix is a multivariate polynomial akin to the more familiar determinant, except that all monomials are given positive sign. In contrast to the determinant, which can be evaluated efficiently using Gaussian elimination, the permanent is known to be #P-complete, even

in the special case of a 0,1-matrix. This classical result of Valiant almost certainly rules out a polynomial-time algorithm for computing the permanent of such a matrix exactly.

However, the question of whether the permanent of a 0,1-matrix can be efficiently approximated has been open for some time. The question has recently been resolved positively. I shall present a fully-polynomial randomised approximation scheme for the permanent of a 0,1-matrix (more generally, an arbitrary matrix with non-negative entries).

Victor Khomenko **and Maciej Koutny**
University of Newcastle upon Tyne
Verification of Bounded Petri Nets Using Integer Programming

A distinctive characteristic of reactive concurrent systems is that their sets of local states have descriptions which are both short and manageable, and the complexity of their behaviour comes from highly complicated interactions with the external environment rather than from complicated data structures and manipulations thereon. One way of coping with this complexity problem is to use formal methods and, especially, computer aided verification tools implementing model checking - a technique in which the verification of a system is carried out using a finite representation of its state space.

The main drawback of model checking is that it suffers from the state space explosion problem. That is, even a relatively small system specification can (and often does) yield a very large state space. To help in coping with this, a number of techniques have been proposed, which can roughly be classified as aiming at an implicit compact representation of the full state space of a reactive concurrent system, or at an explicit generation of its reduced (though sufficient for a given verification task) representation.

Techniques aimed at reduced representation of state spaces are typically based on the independence (commutativity) of some actions, often relying on the partial order view of concurrent computation. Such a view is the basis for algorithms employing Petri Net unfoldings, where the entire state space of a system specified by a Petri Net is represented implicitly, using an acyclic net to represent system's actions and local states. Melzer and Roemer suggested a reduction of the deadlock checking problem to a mixed integer linear programming problem (MIP). Having built a finite and complete prefix of a Petri net unfolding, they generate a system of constraints, which is feasible iff the original Petri net has a deadlock, and then use a standard MIP solver to solve it. This technique reduces memory requirements, but general-purpose solvers work relatively slow, since the system of constraints may be very large.

We present a further development of this approach. The essence of the proposed modifications is to use a specialised solver, which takes into account the information about causality and conflicts between events involved in an unfolding, and exploits specific properties of the system of constraints. The approach can easily be generalised to other verification problems, such as checking mutual exclusion or marking reachability and coverability. Our experiments demonstrate that the resulting algorithms can achieve significant speedups.

Dietrich Kuske
University of Leicester
Languages of series-parallel pomsets

By Kleene's Theorem, recognizable word languages model the behavior of systems obtained by sequential composition, choice, and sequential iteration. To deal with the parallel composition, series-parallel or N-free pomsets were proposed. Lodaya and Weil initiated the consideration of recognizable sets of series-parallel pomsets. In particular, they showed how Kleene's Theorem and Myhill-Nerode's Theorem can be extended to this setting (i.e., they related "rational", algebraically recognizable, and sets of N-free pomsets accepted by "branching automata"). In their full strength, these theorems generalize only to sets of uniformly width-bounded series-parallel pomsets. In the talk, I will show that also Büchi's Theorem can be generalized to sets of series-parallel pomsets: A set of series-parallel pomsets is algebraically recognizable if and only if it is

axiomatizable in monadic second order logic. This equivalence holds also for unbounded sets of series-parallel pomsets (which strengthens a recent result of mine, cf. ICALP'00). Furthermore, I give an extension of Schützenberger's and of McNaughton & Papert's results on aperiodic word languages: Any starfree set of N-free pomsets is first-order axiomatizable, any first-order axiomatizable set is "aperiodic", and any aperiodic and width-bounded set is starfree (cf. STACS'01). The restriction to width-bounded sets in the last implication is necessary. But it is open whether any first-order axiomatizable set is starfree.

Steve Lakin
University of Leicester
Context-sensitive decision problems in groups

The seemingly distinct areas of group theory, complexity theory, and formal language theory interact in an important way when one considers decision problems in groups, such as the word problem - the question of whether a given word in the generators of the group represents the identity or not - or the conjugacy problem - the question of whether two words are conjugate by some element of the group. In general these problems are undecidable, but we are interested in cases where they do have algorithms to solve them - in particular where these algorithms are context-sensitive (ie require space linear in the length of the input). As yet a classification of groups with context-sensitive word problem is still unknown (and appears to be very difficult), however there are many results and questions of interest, amongst these connections with automatic groups, and a question regarding finite extensions of groups with context-sensitive conjugacy problem. This will be an overview talk of some of these questions, the aim being to give a summary of some of the important issues and a general idea of the research.

Martin Lange
University of Edinburgh
Satisfiability Games and Completeness of Temporal Logic

Tableaux-like methods to solve satisfiability or the model checking problem for certain temporal logics face a very particular difficulty: to capture correctly the regeneration of fixed point constructs. This arises for example in CTL* model checking with both least and greatest fixed points, and in checking satisfiability for LTL, CTL, or even the dynamic logic PDL with least fixed points.

We show how a rather novel approach, called focus, can be employed to overcome these difficulties elegantly. The decision procedures are formulated in terms of two-player games taking choices on sets of formulas such that one of the players has the ability to focus on a particular formula and, hence, follow fixed point constructs while they are unfolded.

As opposed to automata-based satisfiability checking this way yields a very simple technique to prove completeness of these logics, i.e. to exhibit an axiomatisation such that every consistent formula of this system is satisfiable. The constructed axiom systems reflects the game rules and the winning strategies for one of the players.

Further work consists of extending these methods to satisfiability of CTL* and the modal mu-calculus, since the proofs of their completenesses are very delicate. Moreover, there are other temporal logics over non-standard structures which are not known to be complete yet.

Gerald Luettgen
University of Sheffield
A Faster-than Relation for Asynchronous Processes

This talk introduces a novel (bi)simulation-based faster-than preorder which relates asynchronous processes with respect to their worst-case timing behavior. The underlying studies were conducted for a conservative extension of the process algebra CCS, called TACS (Timed Asynchronous Communicating Systems), which permits the specification of maximal time bounds of actions. TACS complements work in plain process algebras which compares asynchronous processes with respect

to their functional reactive behavior only, and in timed process algebras which focus on analyzing synchronous processes.

The most unusual contribution of the reported work are results showing that the proposed faster-than preorder coincides with two other and at least equally appealing preorders, one of which considers the absolute times at which actions occur in system runs. The talk also presents the semantic theory of TACS, including a characterization of the largest precongruence contained in the faster-than preorder and its axiomatization in a fragment of the algebra. A small example relating two implementations of a simple storage system testifies to the practical utility of the new theory.

The research, on which this talk is based, has been jointly conducted with Walter Vogler, The University of Augsburg, Germany.

Florent Madelaine
University of Leicester
A family of Colouring problems that are not Homomorphism Problems

A logic capturing (more or less) the class of Homomorphism Problems has been given by Feder and Vardi. This logic is related to some colouring problems I shall introduce. Among those Colouring Problems, some are not Homomorphism problems. They split the two classes. I shall give a few examples of those so-called separating problems and might attempt to give the idea of their construction for a large class of them.

Aileen McLoughlin
Dublin City University
New Noncommutative Bilinear Algorithms for 3 by 3 Matrix Multiplication

In previous work R. Johnson and A. McLoughlin, by a computer-aided search, found new noncommutative bilinear algorithms for 3 by 3 matrix multiplication that require only 23 essential multiplications rather than the 27 required by the conventional method—the same complexity as the algorithm found earlier by J. Laderman, but inequivalent to it (and each other) in a sense that will be made precise. Such algorithms, like Strassen's algorithm for the 2 by 2 case, lead to fast algorithms for matrices of arbitrary size. We report here the discovery of still further algorithms for the 3 by 3 case, requiring 23 essential multiplications, but inequivalent to those mentioned above. In addition we mention new results for some other small matrix sizes.

Alice Miller
University of Glasgow
Using SPIN to Analyse the Tree Identification Phase of the FireWire Protocol

We describe initial attempts to model the Tree Identification phase of the IEEE 1394 High Performance Serial Bus (aka "FireWire") protocol in Promela and to verify properties of the protocol using SPIN. We demonstrate the analysis techniques that are available with SPIN and discuss optimisation techniques that we employ to maintain the tractability of the state-space.

Claudia Nalon
University of Liverpool
Theorem Proving for Synchronous Systems with No Learning

Combinations of non-classical logics is an area of increasing interest as such logics have been shown to be useful for reasoning about complex situations, such as, for instance, the specification and verification of distributed and multi-agent systems. Here, we examine a Temporal Logic of Knowledge, in which the dynamic and temporal components of such systems can be represented. In particular, we examine synchronous systems with no learning: the agent has access to a common clock and if, at a certain moment in time, she is not able to distinguish between two situations, so she will never able to distinguish between them. Then, we present a resolution-based method for

temporal logics of knowledge with synchrony and no learning. The approach is clausal, i.e. the negation of a formula that we wish to prove valid is firstly translated to a normal form. Clauses fall into four classes (initial, literal, modal, or temporal), each of them representing a different constraint/dimension to which resolution is applied. We show that, for the single-agent case, by introducing a set of new temporal clauses, but without adding any resolution rules, we achieve a complete method for theorem-proving in the logic corresponding to no learning and synchrony.

Ranko Lazic, *Tom Newcomb* and Bill Roscoe
Oxford University
On model checking data-independent systems with arrays

A system is data-independent with respect to a data type X iff the only operation it can perform on values of type X is equality testing. The system may also store, input and output values of type X. This can be formalised with X being a type variable. We study model checking of systems which are data-independent with respect to two distinct type variables X and Y, and may in addition use arrays with indices from X and values from Y. The main problem of interest is whether a given system satisfies a given temporal-logic formula for all instances of X and Y. We investigate decidability of this problem depending on the temporal logic used, and on characteristics of the system such as available array operations, number of arrays and available kinds of equality testing.

Practical motivation for model checking data-independent systems with arrays includes verification of memory systems of shared-memory multiprocessors, where X is the type of memory addresses, and Y the type of storable values.

Barry Norton
University of Sheffield
Clocked Transition Systems and the Compositional Modelling of Reactive Components under Synchronous Scheduling

We consider a style of software development where systems are constructed from components with reactive behaviour. Although individually asynchronous, they must form an application that fulfils the synchrony hypothesis and can be scheduled on a monoprocessor. In modelling such systems, we describe the reactive behaviour of components alongside their 'scheduling' behaviour, via ¡I¿clocked transition systems¡/I¿. A notion of composition that includes ¡I¿maximal progress¡/I¿ then allows the derivation of application models where the synchronous scheduling is implicit. Initial results in the use of such models for analysis and execution of such systems are used to motivate future work on the semantics for systems with hierarchy — used in the definition of synchronous subsystems — and concurrency — between them. Thus we plan to design a novel component-based development style for reactive systems, providing the definitional conveniences of Statecharts but with a clear compositional semantics.

Aris T. Pagourtzis
University of Liverpool
Counting the leaves of a nondeterministic computation tree

We investigate properties of the recently defined counting class TotP which contains functions that count the number of all computational paths of PNTMs (polynomial-time bounded nondeterministic Turing machines). TotP is closely related to Valiant's class #P: the two classes are equivalent under 1-Turing reductions. However, TotP is a proper subset of #P unless P=NP. We also define the class #PE (#P "Easy") which is shown to be an intermediate class between TotP and #P. This new class contains all functions f such that: (a) f is in #P, and (b) for any input x the question $f(x) > 0$? is polynomial time decidable. In other words, #PE contains the "hard-to-count-easy-to-decide" counting problems. Some of them are well-known ones, e.g. #PerfectMatchings (computing the number of perfect matchings of a bipartite graph - equivalent to computing the Permanent), #DNF-SAT (counting the number of satisfying assignments of a

DNF Boolean formula), and many more. Here we prove that most of these #PE problems are also contained in TotP. Nevertheless, it seems that this is not the case for all of them—it is shown that TotP is a proper subset of #PE, unless P=NP. We give a sufficient condition for a #PE function to be also in the class TotP; we describe this condition in terms of a self-reducibility property.

Finally, completeness for TotP and #PE is discussed with respect to two reducibility notions: 1-Turing and functional many-one. While several natural problems are complete for both classes under the former reducibility it is open if any of them is also complete (for any of the two classes) under the latter. We show, however, that TotP possesses some (not natural though) complete problems under many-one reductions.

Antonio Puricella **and Iain A. Stewart**
University of Leicester
A generic greedy algorithm, partially-ordered graphs and NP-completeness

Let π be a fixed polynomial-time testable, non-trivial, hereditary property of graphs. Miyano proved that the problem of finding the lexicographically first maximal subgraph of a graph G satisfying property π, when the vertices of G are linearly ordered, is P-hard. Suppose that the vertices of a graph G are not necessarily linearly ordered but partially ordered, where we think of this partial order as a collection of (possibly exponentially many) linear orders, represented by a directed acyclic graph. We prove that the problem of deciding whether a lexicographically first maximal subgraph of G satisfying π, with respect to one of these linear orders, contains a specific given vertex is NP-complete. This result still holds even if we restrict the input graph to be planar bipartite. If the partial order is an out-tree then we prove that the problem is in P.

Peter Saffrey
University of Glasgow
State Space Reduction by Specification Modification

Model checking is a technique used to verify the property of systems. Systems are specified using a specification language and model checking tools exhaustively search the behaviour of the specification to check for conformance to given properties. Exhaustive search can be expensive, so reduction techniques are often needed to make a search space tractable.

This talk will describe one possible reduction technique based on optimising at the specification level. Since a specification is written by a system designer there is potential for the code to be inefficient and produce needlessly large state spaces. Our methods attempt to deduce generic methods for improving inefficient specifications. We deal particularly with concurrent systems and optimising the communication paths, or channels, between concurrent processes.

Paul Sant, **Pierluigi Frisco and Hendrik Jan Hoogeboom**
University of Liverpool
A Direct Construction of a Universal P System

We present a direct universal P system based on splicing. P systems were recently introduced as distributed parallel computability models. They are based on a hierarchically arranged, finite cell-structure consisting of several cell-membranes embedded in a main membrane called the skin. Several variants of these systems have been considered and most of them have been shown to generate recursively enumerable sets or vectors of natural numbers.

The objects of our investigations are P systems using vectors evolving by splicing. A computability model C is computationally complete if the devices in C have the power of Turing machines (or any other type of equivalent device), that is they can generate and/or recognize recursively enumerable languages. A related property is universality. This means that a fixed element of C can simulate any other given device in C. Such an element is called universal.

Any model that is computationally complete has a universal device. This means that it can simulate, for example, a universal Turing machine, which in turn may simulate any element of the

original model. If a P system receives as input the coding of another P system and simulates it then we have a direct construction of a universal P system.

A P system that takes as input an encoding of a Turing machine (which in turn takes as input an encoding of another P system) and simulates it is called a indirect universal P system. In this talk we describe the construction of a direct universal splicing P system.

Carron Shankland
University of Stirling
Prototyping Model Checkers over Symbolic Transition Systems

Model checking is a popular validation technique because it is seen as easy to use. The main problem is that only specific scenarios of finite size can be checked, therefore lots of people are tackling the problem of how to generalise the results from model checking to infinite state, or how to make the model checkers deal with infinite state in some way. We have developed a theory in which transition systems can be represented in a finite way, namely Symbolic Transition Systems (STSs). We've also defined a logic over STSs. In this talk I'll present work on implementing prototype model checkers using a variety of different techniques: XTL and the CADP toolkit, the theorem prover Ergo, which allows interactive proofs in a generalised sequent calculus style, and Rewriting Logic using Maude.

Rick Thomas
University of Leicester
String rewriting in groups

The purpose of this talk is to explore some connections between groups and string rewriting systems. If R is a rewriting system over a set Σ, then R induces a congruence on the set of all strings over Σ called the *Thue congruence* of R. If a group G has a finite complete (i.e. Noetherian and confluent) rewriting system, then G has solvable word problem. In fact, it is sufficient for G to have a solvable word problem that it has a Noetherian rewriting system that is confluent on the Thue congruence class containing the empty word λ; we call such a rewriting system $[\lambda]$-*complete.* Given this, it is natural to ask which groups have a rewriting system R that is $[\lambda]$-complete. We consider a particular case of this question where the right-hand side of every rule in R is the empty word (so that R is automatically Noetherian); such a rewriting system is said to be *special.* It is an open question as to precisely which groups have a special $[\lambda]$-complete rewriting system and we will survey some of what is currently know about this problem.

Michele Zito
University of Liverpool
A Remark on the Space Complexity of Random Formulae in Resolution

The importance of studying the complexity of (propositional) proof systems comes from its close relationship with long-standing open problems in Complexity Theory such as NP=?Co-NP. The complexity measure related to the classical notion of time is the size of a proof, ie the number of lines used in the proof. Recently Esteban and Toran suggested a measure for the space complexity of refusing an unsatisfiable formula in a proof system called resolution.

Although several results are, by now, known on the space complexity of various classes of formulae, a quantitative analysis of the space needed to prove the unsatisfiability of random formulae remained, until recently, somewhat elusive. As an initial step towards the solution of this problem, we point out that a combination of a variant of the classical Davis-Putnam algorithm and a linear time algorithm for 2-SAT outputs resolution refutations of any unsatisfiable random formula within the space bounds stated in the following Theorem.

Theorem 1. For each $k > 1$ there are constants c_1 and c_2 such that almost all unsatisfiable k-CNF formula F on n variables and $m = \Delta n$ clauses with $\Delta > c_1$ can be refused in space $kc_2 n/\Delta^{1/(k-2)}$.

REMARKS:

(1) Tight upper-bounds on the probability that a random 2-CNF formula is satisfiable are instrumental to the proof of Theorem 1. Therefore this Theorem represents a nice application of results on the sharp phase-transition phenomenon of random 2-SAT.

(2) For $k = 3$, $c_1 = 20$ and $c_2 = 1.1$, but no attempt has been made to optimise these values.

(3) Theorem 1 pairs up with the lower bounds proved recently by Ben-Sasson and Galesi, showing the optimality of their result for sufficiently large values of c_1.