REPORT ON BCTCS 2007

The 23rd British Colloquium for Theoretical Computer Science

2-5 April 2007, St Anne's College, Oxford

Sharon Curtis, Jeremy Gibbons

The British Colloquium for Theoretical Computer Science (BCTCS) is an annual forum for researchers in theoretical computer science to meet, present research findings, and discuss developments in the field. It also provides an environment for PhD students to gain experience in presenting their work in a wider context, and benefit from contact with established researchers.

BCTCS 2007 was hosted jointly by Oxford Brookes University and the University of Oxford, and held at the delightful setting of St Anne's College, Oxford during 2–5 April 2007. The event attracted 92 participants, and featured an interesting and wide-ranging programme of 6 invited talks and 48 contributed talks; roughly half of the participants and speakers were PhD students. Abstracts for the talks are provided below; further details, including many on-line slides from the talks, are available via the BCTCS website at `http://www.bctcs.ac.uk`. The financial support of the Engineering and Physical Sciences Research Council (EPSRC), the London Mathematical Society (LMS), and the Formal Aspects of Computer Science Special Interest Group of the British Computer Society (BCS-FACS) is gratefully acknowledged.

One notable outcome from this year's meeting arose out of an update by Samson Abramsky on the formation of a Learned Society for Computer Science in the UK, which was discussed at last year's meeting. Little progress had been made over the past few months, because the proposal drafted by UKCRC for such a learned society had been rejected by the BCS Board of Trustees. This year's meeting gave a near-unanimous authorization to the BCTCS committee to develop the organisation as a Learned Society for Theoretical Computer Science.

Other highlights of the meeting included the opening invited talk, which was the public inaugural lecture of Georg Gottlob, marking his recent appointment as Professor of Computing Science at the University of Oxford. The invited talk of Professor Richard Jozsa, from the University of Bristol, was also of note, being the first invited talk to BCTCS on the topic of quantum computation.

BCTCS 2008 will be hosted by the University of Durham from 7–10 April 2008. Researchers and PhD students wishing to contribute talks concerning any aspect of theoretical computer science are warmly welcomed to do so. Further details are available from the BCTCS website at `http://www.bctcs.ac.uk`.

Bulletin of the EATCS no 92, pp. 165–186, June 2007

Invited Talks at BCTCS 2007

Dimitris Achlioptas, University of California, Santa Cruz, U.S.A.
Random Constraint Satisfaction Problems: from Physics to Algorithms

For a number of random constraint satisfaction problems, such as random k-SAT and random graph coloring, by now we have excellent estimates of the largest constraints-to-variables ratios for which they have solutions. These estimates imply that all known polynomial-time algorithms stop finding solutions long before solutions cease to exist. To understand the origin of this phenomenon we study the evolution of the structure of the space of solutions in random CSPs as constraints are added. In this talk we survey both what physicists hypothesize happens in this evolution, and what has been rigorously proved so far. We will see that the rigorous results are in agreement with the physics predictions and help demystify *survey propagation*, an extremely successful heuristic proposed by physicists for random CSPs.

Steven Alpern, London School of Economics and Political Science
Search Games and Utilitarian Postman Paths on Networks

For any network Q, one may consider the zero-sum search game $\Gamma(Q)$ in which the (minimizing) Searcher picks a unit speed path $S(t)$ in Q, the Hider picks a point H in Q and the payoff is the meeting time $T = \min\{t : S(t) = H\}$. We show first that if Q is symmetric (edge and vertex transitive), then it is optimal for the Hider to pick H uniformly in Q, so that the Searcher must follow a Utilitarian Postman path (one which minimizes the time to reach a random point). We then show that if Q is symmetric of odd degree, with n vertices and m unit length edges, the value V of $\Gamma(Q)$ satisfies $V \geq \frac{m}{2} + \frac{n^2-2n}{8m}$, with equality if and only if Q has a path $e_1, \ldots, e_{n-2}$ which includes $n-1$ vertices such that $Q - \{e_2, e_4, \ldots, e_{n-2}\}$ is a connected set of edges.

Julian Bradfield, University of Edinburgh
How User-Friendly is Independence-Friendly Logic?
(BCS-FACS Keynote Lecture in Formal Methods)

Independence-friendly logic was invented around fifteen years ago by Hintikka and Sandu, and has consistently attracted controversy about whether it is useful or interesting. The arguments are not helped by the numerous subtle details of its semantics that were unclear or disputed. Recently there has been a flurry of activity, with three Ph.D. theses devoted to the fundamentals of IF logic. In this talk, I will introduce IF logic and its many interesting, not to say weird, properties; review the recent work in philosophical logic; and mention my own work on computer science versions of IF logics. The audience may then decide whether IF logic is

user-friendly, where we are the users.

Georg Gottlob, University of Oxford
Living with Computational Complexity

Computer scientists fight a continuing battle against computational complexity. New problems are often not known to be tractable and are solved by exponential-time algorithms even though polynomial algorithms exist. It is a goal of Computer Science to find polynomial algorithms wherever possible. Even when a problem is known to be tractable, the goal is to find fast algorithms whose runtime is bounded by a low-degree polynomial. However, many practically relevant problems are NP-hard and thus presumably intractable. In this case there are several possibilities of practical solutions. For example, one may look at large tractable subclasses or use heuristics or randomized algorithms. This moderately technical talk will illustrate these different ways of countering complexity by various examples, mostly related to the speaker's current or previous work, in particular, XPath query processing, Web data extraction, constraint satisfaction, and combinatorial auctions. The talk will end with some quite daring philosophical considerations on computational complexity and nature.

Richard Jozsa, University of Bristol
Quantum Computation — Principles and Achievements

Quantum computation represents a synthesis of principles from theoretical computer science and quantum physics. This union leads to new physically realisable modes of computation and for some computational tasks (such as integer factorisation) it provides algorithms that are exponentially more efficient than any known classical algorithms. In this talk we will describe the computational model that underlies quantum computation, introducing also the relevant aspects of quantum theory. We will discuss its novel features (relative to classical computation) and describe some illustrative applications of quantum effects in information processing tasks, including remarks on the relation of quantum computational complexity to NP, as currently understood.

Kristina Vušković, University of Leeds
The Use of Decomposition in the Study of Graph Classes defined by Excluding Induced Subgraphs **(LMS Keynote Lecture in Discrete Mathematics)**

We consider finite and simple graphs, and say that one graph G *contains* a second F if F is isomorphic to an induced subgraph of G. G is *F-free* if it does not contain F. Let $\mathcal{F}$ be a (possibly infinite) family of graphs. A graph G is *$\mathcal{F}$-free* if it is F-free, for every $F \in \mathcal{F}$.

Many interesting classes of graphs can be characterized as being $\mathcal{F}$-free for some family $\mathcal{F}$. Most famous such example is the class of perfect graphs. A

graph G is *perfect* if for every induced subgraph H of G, $\chi(H) = \omega(H)$, where $\chi(H)$ denotes the chromatic number of H and $\omega(H)$ denotes the size of a largest clique in H. The famous Strong Perfect Graph Theorem states that a graph is perfect if and only if it does not contain an odd hole nor an odd antihole (where a *hole* is a chordless cycle of length at least four, it is *odd* if it contains an odd number of nodes, and an *antihole* is a complement of a hole).

In the last 15 years other classes of graphs defined by excluding a family of induced subgraphs have been studied, perhaps originally motivated by the study of perfect graphs. The kinds of questions this line of research focused on were whether excluding induced subgraphs affects the global structure of the particular class in a way that can be exploited for putting bounds on parameters such as χ and ω, constructing optimization algorithms (problems such as finding the size of a largest clique or a minimum coloring), recognition algorithms and explicit construction of all graphs belonging to the particular class. A number of these questions were answered by obtaining a structural characterization of a class through their decomposition (as was the case with the proof of the Strong Perfect Graph Theorem). In this talk we survey some of the key results in this area.

Contributed Talks at BCTCS 2007

Sara Adams, Oxford University
On Universality for Timed Automata with Minimal Resources

Timed automata were introduced by Alur and Dill in 1994 and have since become the most prominent modelling formalism for real-time systems. A fundamental limit to the algorithmic analysis of timed automata, however, results from the undecidability of the universality problem: does a given timed automaton accept every timed word? As a result, much research has focused on attempting to circumvent this difficulty, often by restricting the class of automata under consideration, or by altering their semantics.

In this talk, we study the decidability of universality for classes of timed automata with minimal resources. More precisely, we consider restrictions on the number of clocks, states, and clock constants, as well as the size of the event alphabet. Our main result is that universality remains undecidable for timed automata with a single state, over a single-event alphabet, and using constants 0 and 1 only.

This is joint work with Joel Ouaknine and James Worrell.

Thorsten Altenkirch, Nottingham University
How Not to Prove Strong Normalisation

Traditionally, decidability of the equational theory of typed lambda calculi is es-

tablished by showing strong normalisation and confluence of a small-step reduction relation. Here I want to advertise alternatives using either a big-step semantics or normalisation by evaluation, which are technically simpler and closer to the actual implementation.

Joachim Baran, Manchester University
Model-Checking Call-Stack Behaviour

Model-checking regular properties is a powerful verification technique for regular as well as context-free program behaviours. Context-free program behaviours can be used to model a program's call-stack, which permits an accurate abstraction of calls and returns as they occur in "real" software programs. Unfortunately, the call-stack can only be inspected using regular properties, so that a rather simple requirement such as "every call returns" cannot be verified. In fact, the latter requirement is a context-free property, for which model-checking is undecidable.

Recently, through the use of visibly pushdown languages, which are beyond regular languages but strictly included in the context-free languages, model-checking of properties beyond regular expressiveness was made possible. Model-checking visibly pushdown properties of visibly pushdown program behaviours is decidable (EXPTIME-complete). Since call-stack behaviour is expressible as a visibly pushdown property, matching occurrences of calls and returns of "real" software programs can be model-checked.

We propose a new representation of visibly pushdown languages, which allows us a natural specification of visibly pushdown properties. While the existing representations are based on automata and second order logic, we utilise grammars as a characterisation tool. From a specification viewpoint, the grammatical representation bears a greater resemblance to the to-be-checked properties of program behaviours.

Joao Filipe Belo, Manchester University
Dependently Sorted Logic

Logics over many-sorted languages have applications in computer science such as in the specification of computer systems. In this talk I present the generalisation of a system for many-sorted logic to dependently sorted logic. Joint consistency and interpolation will be discussed and their proofs will be shown to carry over to the general case.

Dénes Bisztray, Leicester University
Rule-Level Verification of Business Process Evolutions using CSP

Business Process Reengineering is one of the most widely adopted techniques to improve the efficiency of organisations. Transforming process models, we intend to change their semantics in certain predefined ways, making them more flexi-

ble, more restrictive, etc. To understand and control the semantic consequences of change we use CSP to capture the behaviour of processes before and after the transformation. Formalising process transformations by graph transformation rules, we are interested in verifying semantic properties of these transformations at the level of rules, so that every application of a rule has a known semantic effect. It turns out that we can do so if the mapping of activity diagram models into the semantic domain CSP is compositional, i.e., compatible with the embedding of processes into larger contexts.

Clive Blackwell, Royal Holloway, University of London
Using Bigraphs for Security Analysis and Modelling of the Physical and Logical Aspects of Computer Systems

We have developed a multilayer security model that allows the analysis of systems in their entirety including human and physical factors. The physical and logical layers of our security model and the interaction between them can be formalised with a novel use of Milner's bigraph model. The bigraphical structure composes two graphs with one to represent logical communication and the other physical mobility. A bigraph is used to model the security architecture of systems, where security mechanisms are represented as graph rewriting rules. The defender's objectives can be defined by invariants of the bigraph that hold in secure states of the system, and rewriting rules representing actions that should only be performed by the defender.

The model incorporates the physical dependence of cryptographic and other logical security mechanisms on secure physical locations and channels, which is not considered in most security models. We can also analyse the effects of different types of attacker based on their location and capabilities at both the physical and logical layers. An important general application of bigraphs is modelling the interaction between the control elements of a system and its functional components. We propose some extensions to the model to broaden its applicability and to map the semantics of security problems more faithfully. The levels can represent other types of conceptual entities, such as physically separated hosts or LANs using VPNs to communicate over untrustworthy intermediate networks, as long as they follow the model's semantics. Another extension is the inclusion of a third level that could represent higher-layer semantic entities, or an intermediate level that could be used to model virtualisation.

Jens Blanck, Swansea University
Reduction Between Domain Representations

We will look at reductions between domain representations and the interplay between reductions and intrinsic properties of domain representation.

William Blum, Oxford University
The Safe Lambda Calculus

Safety is a syntactic condition of higher-order grammars that constrains occurrences of variables in production rules according to their type-theoretic order. The *safe lambda calculus* is obtained by generalising the safety condition to the setting of the simply-typed lambda calculus. In this calculus there is no need to rename bound variables when performing substitution, as variable capture is guaranteed not to happen. In the same vein as Schwichtenberg's 1976 characterisation of the simply-typed lambda calculus, we show that the numeric functions representable in the safe lambda calculus are exactly the multivariate polynomials; thus conditional is not definable. We then give a game-semantic analysis of safety by showing that the game semantics of safe lambda-terms are succinctly representable.

Gary Broughton, Kingston University
A Computer Algebra Approach for the Reduction of Matrix Pseudo-Linear Equations

Computer algebra is the study and development of symbolic algorithms with the main purpose of manipulating mathematical expressions in symbolic form. A computer algebra package called ISOLDE has recently been implemented in Maple. ISOLDE is an open source project, containing functionality implementing the formal reduction and finding different types of solutions of linear differential systems of the form $x\frac{dy}{dx} = A(x)y$ where y is a vector with n components and $A(x)$ is an $n \times n$ matrix with coefficients in $\mathbb{C}(x)$ or $\mathbb{C}((x))$. Our goal is to find a general framework which would allow to extend the functionality of ISOLDE for use with other types of matrix linear equations such as difference equations, q-difference equations and also for use with the perturbed eigenvalue problem. Such equations are found in various fields such as digital imaging, physics and engineering.

We have identified pseudo-linear mappings as a potential theoretical framework, which encompasses all of the above types of matrix equations. In this talk we will present how some of the existing algorithms can be extended to this general framework of matrix pseudo-linear equations. Implementation of these algorithms in ISOLDE is in progress.

Bob Coecke, Oxford University
Pictures of the Quantum World

We discuss a graphical language (and the corresponding formal algebra) that supports high-level quantum reasoning. Physicists should welcome this language since it both refines and formalises the highly successful Dirac calculus in a very intuitive two-dimensional fashion. Computer scientists should welcome it since its algebra is in fact an extremely powerful logical system which enables auto-

mated design and verification. The main recent development in this research program is the ability to capture quantum measurements and classical data manipulations within the language which was initially designed to capture quantum entanglement. We are able, for example. to distinguish between classical nondeterminism, stochastic processes, reversible classical processes etc. At the core of all this lies an analysis of the abilities to clone and delete data in the classical world "from the perspective in the quantum world". In this view, the classical world looks surprisingly complicated as compared to the very simple quantum world.

Joey Coleman, Newcastle University
On Proving the Soundness of Rely/Guarantee Rules for Software Development

Reasoning about the implementations of programs during software development requires that there exists a formal semantics in which to frame the discussion. Using such a semantics alone, however, risks losing track of the larger context in all the fine detail. This talk describes some work with Cliff Jones that uses rely/guarantee rules to extend the logical framework used for reasoning, and specifically, the nature of the soundness proof that relates a given set of rely/guarantee rules to their target language.

Gavin Cox, Leicester University
Reversing CCS with SimCCSK

Reversible computation has a growing number of potential application areas such as debugging, testing and the modelling of biochemical systems. To look into the possible use of reversibility in such areas some form of computer aided simulation would be of great advantage. With this in mind I have created a prototype simulator SimCCSK for CCSK, a reversible version of CCS with communication keys by Phillips and Ulidowski.

Mike Dodds, York University
Graph Grammars and Separation Logic

Graph grammars and separation-logic formulae are two methods of defining languages of graph-like structures. Graph grammars are a natural extension of context-sensitive string rewrite languages to the world of graphs. The language defined by a graph grammar is the set of graphs derivable from some initial element. Separation logic is a recently-developed logic for reasoning about heap data. A separation-logic formula defines the language of heaps which satisfy it. These two approaches have essentially the same purpose, so we may ask how they are related. In this talk I will show that restricted classes of such formulas and grammars are equivalent, by presenting a translation between these classes. I will also discuss the application of this result to the field of shape safety, where

graph grammars and separation-logic formulas are used to define pointer structure properties.

Neil Ghani, Nottingham University
Indexed Datatypes

I recently heard Jeremy Gibbons giving a talk about indexed datatypes and thought "I know something about that". In this talk, I will attempt to say what I would have said had I given a talk on indexed data types.

In particular, I will talk about the semantics of nested types and our work on indexed containers. The former corresponds to "big"-indexing as found in impredicative type theories while the latter corresponds to "small"-indexing as found in dependent type theories.

Alexander Green, Nottingham University
A Quantum IO Monad for Haskell

The functional programming language Haskell provides monadic programming constructs to model computations that may involve side effects. One of the main monads provided in Haskell is the IO monad, which contains all the various IO functions that can be used in a Haskell program. In this talk I describe a Quantum IO (QIO) Monad, which can be used to construct, and evaluate, quantum computations. The quantum computations are constructed using the 'do' notation, and allow qubit registers to be created, operated on, and measured. The allowed operations are unitary transformations that relate to the quantum circuit model of quantum computations: single qubit rotations, conditional branches, and the swapping of qubits. Finally, the QIO Monad allows the evaluation of these quantum computations to proceed in one of two ways: either by simulating the measurements probabilistically; or by viewing the internal state of the system. In this talk I shall give a brief introduction to the QIO Monad, including some simple quantum computations, and explain how they are evaluated.

Stephen Gorton, Leicester University
Extending Business Process Models with Policies

Business Process Modelling (BPM) is an extension of workflows, used to define business processes. BPM is often coupled with Service-oriented Architecture (SoA) to provide flexible software solutions for a variety of applications. However, this approach is limited, as the design of workflows is done offline and therefore runtime information cannot be fully integrated. Furthermore, workflows can become too complex, with the needs and wants of the business user needing to be captured in the initial design.

We have proposed the idea of combining policies with (SoA-based) workflows to create policy-driven service-oriented business modelling. Policies are informa-

tion that systems react to without the need for recompiling or redeploying. From a business perspective, it is valuable to allow policies such as "if service X fails, then abort the workflow" to a workflow or set of workflows. We present some recent work about how and where policies that capture user information can be integrated into a workflow, and how they affect the workflow.

This work has been done with Stephan Reiff-Marganiec and Carlo Montangero under the EU Project SENSORIA, IST-2005-016004.

Matthew Hague, Oxford University

Symbolic Backwards-Reachability Analysis for Higher-Order Pushdown Systems

Higher-order pushdown systems (PDSs) generalise pushdown systems through the use of higher-order stacks, that is, a nested "stack of stacks" structure. We further generalise higher-order PDSs to higher-order Alternating PDSs (APDSs) and consider the backwards reachability problem over these systems. We prove that given an order-n APDS, the set of configurations from which a given regular set of configurations is reachable is itself regular and computable in n-EXPTIME. We show that the result has several useful applications in the verification of higher-order PDSs such as LTL model checking, alternation-free μ-calculus model checking, and the computation of winning regions of reachability games.

Temesghen Kahsai Azene, Swansea University

Testing from CSP-CASL Specification

We propose a new testing framework from the specification language CSP-CASL. Our approach is based on black box testing, in which we define a point of control and observation. Our test cases consist of a finite sequence of symbols of a vocabulary and a verdict which states the result of the test. We especially investigate how test verdicts behave under refinement.

Leonidas Kapsokalivas, Kings College, University of London

Survey of Genetic Algorithms for the Protein Folding Problem

In this survey we try both to summarise the main Genetic Algorithm (GA) approaches to the protein folding problem and to address critical design issues affecting the performance of such algorithms. We focus on the HP model where protein folding is reduced to an energy minimization problem. Even for this simple lattice model, though, protein folding is proved to be NP-complete. Search-based techniques are often the most suitable for such hard optimization problems, where an exhaustive exploration of the solution space is infeasible and thus we try to find the optimal solution through an intelligent traversal of the solution space. Genetic algorithms are very popular techniques of this kind, due to their wide applicability in various optimization problems. Their generality, though, can result in poor

performance as it is harder to fine tune them for the specific problem. Concerning protein folding, our aim is to discuss how various genetic algorithms cope with the representation of a solution-conformation, the mutation and the crossover. There is a wide variety of choices and in most cases it is hard to justify a specific choice with analytic arguments rather than intuitive ones. Nevertheless, a comparative study of results can give us a better idea of the key features an efficient genetic algorithm for the protein folding problem should include.

Oliver Kullmann, Swansea University
SAT and the Polya Permanent Problem

Since the problem of computing the number of perfect matchings in a bipartite graph is #P-complete, polynomial-time algorithms likely exist only for special classes. One approach is by reducing the underlying (hard) permanent-computation to some (easy) determinant-computation. This approach goes back to an exercise posed by Polya in 1913, and can be expressed in many closely related ways, for example as the "even digraph" problem (is there an even directed cycle in a given directed graph?), or whether some square matrix is sign-non-singular (all matrices with the same sign pattern are invertible).

Robertson *et al.* and McCuaig showed that all these problems are solvable in polynomial time. In our earlier work the qualitative-matrix-approach was expanded, and a natural embedding of the problem into some (boolean) satisfiability problem (exploiting autarky theory) was demonstrated. As an application, for example, it is shown that hypergraph 2-colourability is decidable in polynomial time if the maximum of the difference in the number of hyperedges and the number of vertices for all sub-hypergraphs is zero.

In this talk I introduce this old (and new) area, hopefully giving some feeling for the subject, and communicating open problems.

David Lundin, Surrey University
Running Fair and Accurate Electronic Voting Schemes that are Safe, Secure and Reliable

In the last election I assume you dropped your ballot into the box and then you went home. Did you turn up first thing in the morning and check that the ballot box was empty at the start of the election? Did you stay and observe the box all day to make sure that no one dropped false ballots in there? Did you follow it to the place it was emptied and watch the votes being counted? Did you check all those ballots that were thrown out because they were allegedly spoilt to make sure that they were, in fact, spoilt? If you did all those things, do you have good friends in all other constituencies who were able to do the same there? If not, how do you know that the election was fair and accurate? All these questions arise in a traditional paper-based election system and can be solved using a slightly different

electronic voting system – not an opaque system like the now infamous American Diebold system, but a transparent one that can be audited and checked by anyone.

One such system is Prêt à Voter. In this system you enter the booth with your piece of paper and mark your choices on the ballot form like you have always done. But then, before you leave the booth, you separate the form into two halves, one of which becomes your encrypted receipt. Instead of dropping this into the ballot box and trusting that it will be counted properly, you allow it to be scanned into a computer and then you take it home. After the election, the election authority publishes all the encrypted receipts on the Internet and you can check that yours is included. If a random selection of voters check that their receipts are included, you can be confident that no one has cheated.

The Electronic Voting Group at the University of Surrey develops the Prêt à Voter system, and has made the first ever implementation and used it to run the University of Surrey Students' Union's Sabbatical Elections 2007. The talk starts with the benefits of electronic voting, touches on the properties of Prêt à Voter and finishes with our experience of a large-scale implementation of an electronic voting system.

Greg Manning, York University
The GP Project: Environment and Implementation

GP (Graph Programs) is a rule-based programming language for solving graph problems at a high level of abstraction, freeing programmers from dealing with low-level data structures. The core of GP consists of three constructs: single-step application of a set of conditional graph-transformation rules; application of a rule set as long as possible; and sequential composition of programs. The addition of a few powerful branching and iterating constructs makes the language much easier to use whilst preserving the simple semantics.

As the implementation of GP nears maturity, we discuss the overall structure of the components and some of the issues which have arisen in the development of this highly non-deterministic graph-transformation based system.

Steve Matthews, Warwick University
Connections between Domain Theory and Fuzzy Sets

It is observed that the axioms for partial metrics with values in quantales coincide with the axioms for Q-sets (M-valued sets, sets with fuzzy equality, quantale-valued sets) for commutative quantales. Omega-sets (sets valued in complete Heyting algebras) correspond to the case of partial ultrametrics. Partial metrics arise in the context of denotational semantics, and Omega-sets arise in the context of sheaf theory. Some corollaries of this new link between partial metrics and sheaves are discussed.

Chris McCaig, Stirling University

An Algorithm for Deriving Mean Field Equations From Large Process Algebra Models

In theoretical biology models often describe systems at the level of the population. This is convenient for further analysis, but has the drawback that information about populations cannot be directly observed. In contrast, models at the level of the individual can be based on direct observation, but only limited algebraic analysis is available. Bridging the gap between these different levels is an important challenge as it allows the derivation of population level models which take into account the individual interactions which are fundamental to the behaviour of the population.

Here we present an algorithm which, given an individual-based model of a system in the process algebra WSCCS, can produce a system of mean field equations which describe the mean behaviour at the level of the population.

Divina Melomey, East London University

A Comparative Study of Modelling Languages for Agent Systems

Agent Oriented Software Engineering (AOSE) is one of the recent options that have emerged in the field of Software Engineering. This paradigm is based on the concept of an agent, an autonomous computing entity. Some of the benefits AOSE provides to system developers are concepts and notations that relate to real life situations. These concepts include knowledge, behaviour, beliefs and desires, as well as characteristics similar to human intelligence and mobility. Methodologies are therefore required to help systems developers to model complex solutions during the software development process. For a methodology to be complete there is the need to clearly model internal structure representation as well as the external behaviour using a clearly understood modelling language. The modelling language must be well defined semantically such that developers can easily map abstracts in the domain to underlying concepts and computational structures. One thing worth taking into consideration is that different problems may require different abstractions and hence an appropriate modelling language for modelling. This talk examines recent research in agent modelling languages and compares the modelling languages for modelling agents with respect to mobility. The criterion for comparison is based on the functions of a modelling language, its characteristics and semantics.

Zoltan Miklos, Oxford University

Generalized Hypertree Decompositions: NP-Hardness and Tractable Variants

The generalized hypertree width $GHW(H)$ of a hypergraph H is a measure of its cyclicity. Classes of conjunctive queries or constraint satisfaction problems

whose associated hypergraphs have bounded GHW are known to be solvable in polynomial time. However, it has been an open problem for several years if, for a fixed constant k and input hypergraph H, it can be determined in polynomial time whether $GHW(H) \leq k$. Here, this problem is settled by proving that even for $k = 3$ the problem is already NP-hard. On the way to this result, another long standing open problem, originally raised by Goodman and Shmueli in 1984 in the context of join optimization is solved. It is proven that determining whether a hypergraph H admits a tree projection with respect to a hypergraph G is NP-complete. Our intractability result on generalized hypertree width motivates further research on more restrictive tractable hypergraph decomposition methods that approximate general hypertree decomposition (GHD). We show that each such method is dominated by a tractable decomposition method definable through a function that associates a set of partial edges to a hypergraph. By using one particular such function, we define the new Component Hypertree Decomposition method, which is tractable and strictly more general than other approximations to GHD published so far.

Neil Mitchell, York University
Making Haskell First Order

Haskell is a higher order programming language, with functions as first class data values. Haskell has explicit lambdas, currying, monads and type classes – all of which introduce higher order functions into a program. While these higher order functions serve to make code more compact, they can prove awkward when it comes to program analysis.

This talk describes a method for removing all higher order functions from a Haskell program, keeping the structure of the original program wherever possible.

Arjan Mooij, Nottingham University
Constructing and Reasoning about Security Protocols using Invariants

We explore the use of the programming method of Feijen and van Gasteren for the construction of security protocols. This method addresses the derivation of concurrent programs from a formal specification, and it is based on common notions like invariants and pre- and post-conditions. We show that fundamental concepts like secrecy and authentication can be specified nicely in this way. Using some small extensions, the style of formal reasoning from this method can be applied to the security domain, and matches some earlier guidelines. To demonstrate our approach, we discuss an authentication protocol or a key distribution protocol.

Joseph Morris, Dublin City University
Models for Higher-Order Nondeterministic Functions

There has been renewed interest of late in the provision of general nondeterminacy

in functions, where by "general" we mean that the nondeterminacy may be demonic or angelic, and may be unbounded. Published approaches to nondeterministic functions employ semantic models based on one of predicate transformers, upclosed multirelations, or free completely distributive lattices. Each technique can be used to create a range of models. We show that there are many isomorphisms between the various models, to the extent that we can map any known model that uses one approach to an isomorphic model in either of the other two. We examine the various models from the perspective of higher-order functional programming, and show that the standard models in all approaches are inadequate. There are two known richer models based on predicate transformers and free completely distributive lattices, respectively, and they appear to support higher-order nondeterministic programming very well. We show that the two models are isomorphic. We also show how to construct a new higher-order model isomorphic to these using multirelations. Part of the evidence that these richer models are adequate is that they underpin an algebra for reasoning about nondeterminacy in programming language terms. We show that the standard *map* and *fold right* operations of functional programming extend to the nondeterministic case without change, and we use the algebra of nondeterminacy to prove that the well-known *list fusion* law holds without restriction even when the participating functions employ arbitrary nondeterminacy.

Peter Mosses, Swansea University
Formal Semantics Online

Over the past 40 years, various formalisms for semantics have been developed, and used to give semantic descriptions of some major programming languages. However, it appears that rather few semantic descriptions are currently available online. A comprehensive online archive of existing semantic descriptions would surely be a valuable resource in itself, and enhance the visibility of formal semantics in the computer science community. This talk proposes a new initiative to establish such an archive. We also consider the possibility of incorporating canonical descriptions of commonly-occurring individual constructs, which could serve as reusable components, and substantially lower the effort required for future semantic descriptions.

Long Nguyen, Oxford University
Security in Pervasive Computing: An analytical survey of authentication protocols based on human interaction

Recently, people have been doing a lot of research in the area of pervasive computing. One of the main challenges is how we can establish secure communication over an untrusted high-bandwidth network without any initial knowledge or a public-key infrastructure. An approach studied by a number of researchers is

to build security though human work creating a low-bandwidth empirical (or authentication) channel where the transmitted information is authentic and cannot be faked or modified. An example is conversation between the users of systems. In this talk, we give an analytical survey of authentication protocols of this type.

We start with non-interactive authentication schemes, and then move on to analyse a number of strategies used to build interactive pair-wise and group protocols that minimise the human work relative to the amount of security obtained as well as optimising the computation processing. Many of the protocols are based on the human comparison of a single short authentication string (perhaps 16 bits), transmitted over the empirical channel that is the output of a new cryptographic primitive termed a *digest*, which uniformly digests many kilobytes of information into a short authentication string.

Liam O'Reilly, Swansea University
Implementing CSP-CASL

Recently, CSP-CASL has been designed as a specification language tailored to the description of distributed systems, capturing process aspects as well as data properties. Here, we develop theorem proving support for CSP-CASL. To this end, we translate CSP-CASL specifications into the input language of the already established tool CSP-Prover.

Joel Ouaknine, Oxford University
The Cost of Punctuality

Alur, Feder, and Henzinger (JACM 1996) introduced Metric Interval Temporal Logic (MITL) as a fragment of the real-time logic Metric Temporal Logic (MTL) in which exact or punctual timing constraints are banned. Their main result showed that model checking and satisfiability for MITL are both EXPSPACE-Complete.

Until recently, it was widely believed that admitting even the simplest punctual specifications in any linear-time temporal logic would automatically lead to undecidability. Although this was recently disproved, until now no punctual fragment of MTL was known to have even primitive recursive complexity (with certain decidable fragments having provably non-primitive recursive complexity).

In this talk, we present a 'co-flat' subset of MTL that is capable of expressing a large class of punctual specifications and for which model checking (although not satisfiability) has no complexity cost over MITL. Our logic is, moreover, qualitatively different from MITL in that it can express properties that are not timed-regular. Correspondingly, our decision procedures do not involve translating formulae into finite-state automata, but rather into certain kinds of reversal-bounded Turing machines.

Rawle Prince, Nottingham University
Reasoning with Containers

Containers seek to capture those data types which can be thought of using the metaphor of 'shapes' and 'positions' where data can be stored. Moreover, there is a representation theorem which gives a unique representation for all polymorphic functions between containers. In this talk, I consider the prospects for using containers to formally verify properties of polymorphic functions between containers in a theorem prover.

Stephan Reiff-Marganiec, Leicester University
Logic-based Detection of Conflicts in APPEL

Appel is a general language for expressing policies in a variety of application domains with a clear separation between the core language and its specialisation for concrete domains. Policies can conflict, thus leading to undesired behaviour. We present a novel formal semantics for the Appel language based on ΔDSTL(x) (so far Appel has only had an informal semantics). ΔDSTL(x) is an extension of temporal logic to deal with global applications: it includes modalities to localize properties to system components, an operator to deal with events, and temporal modalities à la Unity. A further contribution of the talk is the presentation of techniques based on the semantics to reason about conflicts.

Robert Reitmeier, Nottingham University
Dependent Types by Example

We give an introduction to *dependent types*, accompanied by examples in a newly developed core language.

Ondrej Rypacek, Nottingham University
Categorical Design Patterns

The development of design patterns in object-oriented programming aims at capturing good software design in a re-usable generic form. However, design patterns are not expressible in conventional object-oriented programming languages. To address this shortcoming, we need to model and understand design patterns precisely. We achieve this by moving to the abstract setting of category theory. In this talk, we formalize two "beliefs" about Command and Visitor. Moreover we show how Decorator and Adapter are instances of Composite — a relationship previously undocumented.

Gift Samuel, Swansea University
Implementation of the Stable Revivals Model in CSP-Prover

The stable revivals model is a recently developed CSP model that allows one to reason about stuckness and responsiveness. In this talk, we discuss an implemen-

tation of the stable revivals model in CSP-Prover, an interactive theorem prover for CSP implemented using the theorem prover Isabelle, We talk about the implementation of the domain and the semantical functions, and how to prove various properties of them in Isabelle.

Sam Sanjabi, Oxford University
Studying The Semantics of Aspect Oriented Programs by Translation

An aspect allows the programmer to combine a piece of code with a predicate, such that the code executes whenever the predicate is true during the execution of some underlying program. The predicate may depend on the dynamic state of the executing program, while the aspect might prevent pieces of the underlying code from executing. First, I'll describe how the semantics of a language in which neither of these situations arises can be understood (via a fully abstract translation) in terms of a language of general references. Second, I'll describe how aspects which prevent code from executing can similarly be translated into a language with exceptions.

Aadya Shukla, Oxford University
Meta-modelling Challenges in Clinical Information Management

With the practice of medicine becoming more evidence-based and data-intensive, clinical information management presents its own challenges for computer scientists. Some of the important concerns for information system designers for clinical studies are: dealing with the complexity of the clinical domain, increased regulations, requirements for data sharing and exchange, data reuse, raised expectations of functionality, reliability and distribution of data, and issues of security and attribution. Additionally, information system design should be flexible enough to incorporate the volatile nature of clinical knowledge and realistic mappings between an information system and the clinical study, as clinical measurements hold meaning in a given context. Application of the MDA (Model Driven Architecture) approach for clinical information management has been useful but current meta-modelling tools and techniques are sometimes not sufficient for the clinical domain.

This talk focuses on highlighting important meta-modelling issues, which need to be addressed with respect to the concerns raised above. Realistic examples are drawn from the Breast Cancer Clinical Trial Information Management within the CancerGrid (`http://www.cancergrid.org`) project.

Iain Stewart, Durham University
Improved Upper and Lower Bounds on the Feedback Vertex Numbers of Grids and Butterflies

We improve upon the best known upper and lower bounds on the sizes of minimal

feedback vertex sets in butterflies. Also, we construct new feedback vertex sets in grids so that for a large number of pairs (n, m), the size of our feedback vertex set in the grid $M(n, m)$ matches the best known lower bound, and for all other pairs it differs from this lower bound by at most 2.

Chris Tofts, HP Labs Bristol
Exploiting Strong Attractors to Slaughter Monsters: Taming 10^{1500} States and Beyond

The 'holy grail' of automated state-based model checking is to build precisely the states needed to validate a property and no more. I will present a natural filter on automata which exploits the nature of the design of concurrent systems to order the state construction. We demonstrate that this can provide a sequence of approximating models which permit us to both address infinite systems and large finite systems (a 500 component 10^{1500} state system being effectively modelled). The models are optimal, in the number of states used, for the parameters (state occupancy probabilities and consequent rewards) they attempt to approximate. The filter on the states we deduce is an interesting variant on the near decomposable class of systems presented by Simon and Ando (1961), but one where the small value terms do not necessarily dominate the long run behaviour. The semantic nature of the decomposition avoids the need to instantiate any values, or introduce arbitrary numerical bounds during the automata construction, enabling us to derive bounds on permitted behaviour without an expensive rebuild of the automata. Whilst the main focus of the work is the evaluation of properties of probabilistic systems, probabilities are not required for the construction; consequently the approach can be exploited for qualitative arguments of system correctness. Seven examples of layering of various systems are presented. One of the most powerful properties of the approach is that it is reasonable to argue that either the approximation technique works, or the system is inherently badly designed.

Theodoros Tsokos, Birmingham University
Representing CSP in a Sequential Calculus

Communicating Sequential Processes (CSP) is a formal language for describing patterns of interaction in concurrent systems, a process calculus. In research, it is used as an abstract way to study concurrent systems, and in industry as a tool for verification.

I am introducing another calculus (Strategy Calculus). It is a kind of lambda-calculus, not including higher order functions. As its name suggests, it is connected to game semantics. A type is a game and a term plays a strategy. It has value types (types of opponent-first strategies) and computation types (types of proponent-first strategies).

In my talk, I shall give a brief description of Strategy Calculus and then I

shall give a translation from CSP to it. It seems plausible that this translation is adequate with respect to finite traces, divergence and infinite traces. I shall describe theorems and conjectures that hopefully will lead to such a result.

Emilio Tuosto, Leicester University
HD-automata with Distinctions

History Dependent automata (HDA) provides syntax-independent operational models of a class of history-dependent formalisms. States and transitions of HDA are enriched with (finite sets of) names and symmetries on them. Previously, HDA have been successfully applied to modelling early and late bisimulation in pi-calculus and hyperbisimulation in fusion calculus. However, current HDA are not adequate for modelling behavioural semantics where more sophisticated forms of name relationships are central.

This talk presents a variant of HDA built on top of named sets with distinctions. Specifically, we extend named sets, the basic building blocks of HDA, with a notion of distinction so that names can coalesce if the distinction allows it. As a case study, we show how HDA over named sets with distinctions can model open bisimulation of pi-calculus.

Terry Walcott, East London University
Predicting new markets for Small Businesses

Consumers demand that organisations cater for their specific needs; as such the firm that does not conform is likely to lose business and become non-competitive. This could be a "tall order," especially when consumers themselves are often confused by products that are being manufactured and assembled for their consumption. Therefore, firms will need to predict accurately in order to maintain and support existing and new customers.

Accurate forecasts will ensure that managers and investors can make operational, strategic and tactical decisions for business survival. In order for such decisions to be made, one has to consider a barrage of technological advances that are currently available. Ideally, this may be an easy option when your organisation is large and has a wealth of expertise, cash supplements and reserves. Even then, this is an intricate process that can become tedious with the sustained pressures of competitors.

For small firms to venture into the technological realms they must be comfortable enough to trust existing technologies. Small business investors and managers (this often is a dual role) are sceptical of wasting money that might not be available. Unfortunately, there are no full proof assurances in business but there are some techniques that if harnessed effectively can help smaller organisations.

Artificial Neural Networks (ANNs) have been fruitful for helping large firms to predict and classify patterns in data steadily over the last twenty years. The

Delphi technique has been applied in this research to identify current problems that small businesses face (at least from an expert point of view). It also suggests (proposes) an appropriate ANN solution model for the problems identified. This talk describes modelling of the requirements necessary for predicting new markets for small businesses using neural networks.

Yonghong Xiang, Durham University
Augmented k-ary n-cubes

We propose a new interconnection network, the augmented k-ary n-cube AQ_n^k. We prove that AQ_n^k is a Cayley graph; thus it is vertex transitive. Also, we prove that AQ_n^k is $(4n-2)$-regular and $(4n-2)$–connected. We also show how to construct $(4n-2)$ node-disjoint paths between any two given nodes in AQ_n^k (which is optimal).

Letting $D(G)$ denote the diameter of a graph G and let Q_n^k be the k-ary n-cube, we prove that $D(AQ_2^k)\lfloor 2k/3 \rfloor$, and that $D(AQ_n^k)$ is almost half of $D(Q_n^k)$; however, obtaining $D(AQ_n^k)$ exactly, for $n \geq 3$, appears to be difficult. Finding the diameter of $D(AQ_n^k)$ and an optimal routing algorithm in $D(AQ_n^k)$ is still open.

Yong Xie, Oxford University
Extensible Non-Cooperative Games

Game theory has been an active research area for the past few decades for studying the strategic interactions among players (agents); e.g., the Nash equilibrium is a set of (mixed) strategies, one for each player, such that no player has the incentive to unilaterally change his strategy. However, little has been known about how to use game theory in large systems such as the Internet, mainly due to the issues of lack of scalability and extensibility. To address these issues, we propose the concept of *extensible games*. We introduce two ways of composing games, namely additive and multiplicative composition, and show their soundness and completeness in preserving Nash equilibria. Motivating examples and potential applications will be discussed as well.

Hong Qing Yu, Leicester University
A Modified LSP Method for Dynamic Web Service Evaluation and Selection

The LSP method extends existing scoring techniques and provides a means for the development of complex criterion functions using continuous preference logic. It has been successfully used to evaluate hardware systems, software systems and websites. However, the current LSP method is not sufficient to help in selecting web services in the contexts of dynamic service composition. This area depends not only on static service functional and non-functional requirements (which are traditionally considered in LSP applications), but also on run-time context such as business policies and pre/succeeding service instances in workflow patterns.

Furthermore, web services are an open architecture, so it is difficult to predict evaluation functions and analyse aggregation structures of all elementary criteria. We propose a modified LSP method to target the problem of dynamic web service evaluation and selection. The key modifications are the addition of rule-based criteria creation, automated invocation of type-based unified evaluation methods and dynamic assignment of the aggregation structure.