

REPORT ON BCTCS 2015
31st British Colloquium for
Theoretical Computer Science
Middlesex University, London, September 14-18, 2015

The British Colloquium for Theoretical Computer Science (BCTCS) is an annual forum in which researchers in Theoretical Computer Science can 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 to benefit from contact with established researchers.

BCTCS 2015 was hosted by Middlesex University, and held from 14th to 18th September, 2015. The event attracted over 50 participants, and featured an interesting and wide-ranging programme of seven invited talks and 12 contributed talks, including two Turing Award winners as well as a Fields Medallist, covering virtually all areas of the subject. Furthermore, the scheduling of BCTCS coincided with – and thus included an excursion to – a special Computer Science Day on Algorithms and Cryptography hosted on 17th September by the London Mathematical Society to celebrate its 150th anniversary and held at the Royal Society in London. Abstracts for all of the talks from BCTCS 2015 are provided below.

Tony Hoare (Microsoft Research, Cambridge) opened the colloquium discussing the interaction between concurrent and sequential processes in a Keynote Lecture on the Monday. Tony is well-known for developing the formal language CSP (Communicating Sequential Processes) to specify the interactions of concurrent processes. On Tuesday the semantics theme continued when Per Martin-Löf (Stockholm) opened proceedings with a talk on spreads, repetitive structures, and functional causal models. Per is best-known for the type theory that bears his name. In the afternoon Samson Abramsky (Oxford) gave a fascinating talk on contextuality at the borders of paradox. Contextuality is a key feature of quantum mechanics that permits quantum information processing and computation to transcend the boundaries of classical computation. Wednesday was the designated BCTCS maths day in which we welcomed Thomas Hales (Pittsburgh), our LMS Keynote Speaker in Discrete Maths. Tom is best-known for proving the Kepler Conjecture about the close packing of spheres, which he later verified correct in the theorem prover HOL light. The theorem-proving theme continued in the afternoon when the renowned mathematician Tim Gowers (Cambridge) spoke on his ideas of an extreme human-oriented, heuristic-based automated theorem prover that would be of use to everyday mathematicians. Tim, who is best known for his work in Functional Analysis and Combinatorics spoke eloquently about his

programme to remedy a lack of engagement between most working mathematicians and the automated theorem proving community. Friday saw another star of Computer Science, as Joseph Sifakis (Lausanne) spoke about rigorous system design, a talk that was of interest to both theorists and the more application-oriented alike. The colloquium finished with Andrei Krokhin (Durham) discussing recent research in the value constraint satisfaction problem.

BCTCS 2016 will be hosted by Queen's University, Belfast, from 22nd–24th March 2016. Researchers and PhD students wishing to contribute talks concerning any aspect of Theoretical Computer Science are cordially invited to do so. Further details are available from the BCTCS website at www.bctcs.ac.uk.

Invited Talks at BCTCS 2015

Samson Abramsky (Oxford University) Contextuality: At the Borders of Paradox Contextuality is a key feature of quantum mechanics that provides an important non-classical resource for quantum information and computation. Contextuality can be understood as arising where we have a family of data which is locally consistent, but globally inconsistent. From this point of view, it can be seen as a pervasive phenomenon, arising not only in quantum mechanics, but in many other areas, including databases and constraint satisfaction. There are also remarkably direct connections to logical paradoxes. One can say that contextual phenomena, which we must accept as key features of our picture of physical reality, lie at the very borders of paradox, but do not cross those borders. We present a sheaf-theoretic analysis of contextuality as a general phenomenon, and some of its applications, in quantum mechanics and beyond.

Tim Gowers (University of Cambridge) Extreme Human-Oriented Theorem Proving I will describe the philosophy behind a research programme that I have undertaken with Mohan Ganesalingam, in which our aim is to create an automatic theorem prover that “thinks like a human”. An important part of this philosophy is to disallow methods that exploit what are traditionally regarded as the advantages of computers, in particular the ability to do calculations very rapidly. It may sound perverse and pointless to try to write a computer program that is restricted to what humans can do already; I will try to explain why I strongly disagree with this assessment.

Thomas Hales (University of Pittsburgh) The formal proof of the Kepler conjecture In 1611, Kepler asserted that no packing of congruent balls in space can have density greater than the familiar cannonball arrangement, called the face-centered cubic packing. This statement, known as the Kepler conjecture, is now a theorem that has been formally verified. A theorem is formally verified if every step of the proof has been checked at the level of the primitive inference rules of logic and the foundational axioms of mathematics. This talk will present the Kepler conjecture, its proof, and background about the formal verification of the-

orems.

Tony Hoare (Microsoft Research) *Laws of Programming with Concurrency*
The basic Laws of Nature sought by many branches of science, as well as the basic axioms postulated in many branches of mathematics, have historically been expressed in great generality as algebraic equations, or occasionally as inequalities. Nowadays these equations provide the theoretical foundation for the design of automated tools which are widely used to help scientists and engineers in pursuit of their goals. This is the way in which Isaac Newton still contributes to mechanics, Blaise Pascal to statistics, James Clark Maxwell to electronics, and George Boole to computer Logic Design. How many Computer Scientists and Software Engineers are familiar with the laws which underlie their own professional practice? They are remarkably similar to the laws of arithmetic, taught even today to school children. I will present arguments that they are both generally true of computer programs, and provide the foundation for tools that are widely used in programming practice. And the laws of concurrent programming are no more complicated than those for sequential programming.

Andrei Krokhin (Durham University) *The complexity of general-valued CSPs*
An instance of the *Valued Constraint Satisfaction Problem (VCSP)* is given by a finite set of variables, a finite domain of labels, and a sum of functions, each function depending on a subset of the variables. Each function can take finite values specifying costs of assignments of labels to its variables or the infinite value, which indicates an infeasible assignment. The goal is to find an assignment of labels to the variables that minimizes the sum. The case when all functions take only values 0 and infinity corresponds to the standard *CSP*. We study (assuming that $P \neq NP$) how the complexity of *VCSP* depends on the set of functions allowed in the instances, the so-called constraint language. Massive progress has been made in the last three years on this complexity classification question, and our work gives, in a way, the final answer to it, modulo the complexity of *CSPs*. This is joint work with Vladimir Kolmogorov and Michal Rolinek (IST Austria).

Per Martin-Löf (Stockholm University) *Spreads, repetitive structures, functional causal models*
There is a notion of causal space-time, or sample space-time. It is related to Brouwer's spreads, the statistical notion of repetitive structure, and Judea Pearl's functional causal models. From the point of view of intuitionistic type theory, the way to think about a causal space-time is as an infinite context in which variables are not only typed but also, sometimes, defined, that is, assigned a value dependent on previous undefined variables.

Joseph Sifakis (RiSD laboratory, EPFL) *Rigorous System Design Today*, the development costs of high confidence systems explode with their size. We are far away from the solution of the so called, software crisis. In fact, the latter hides another much bigger: the system crisis. In my talk I will discuss rigorous system design as a formal and accountable process leading from requirements to

correct-by-construction implementations. I will also discuss current limitations of the state of the art and advocate a coherent scientific foundation for system design based on four principles: 1) separation of concerns; 2) component-based construction; 3) semantic coherency; and 4) correctness-by-construction. The combined application of these principles allows the definition of a methodology clearly identifying where human intervention and ingenuity are needed to resolve design choices, as well as activities that can be supported by tools to automate tedious and error-prone tasks. The presented view for rigorous system design has been amply implemented in the BIP (Behavior, Interaction, Priority) component framework and substantiated by numerous experimental results showing both its relevance and feasibility. I will conclude with a discussion advocating a system-centric vision for computing, and a deeper interaction and cross-fertilization with other more mature scientific disciplines.

Contributed Talks at BCTCS 2015

Leroy Chew (University of Leeds) Lower bounds: from circuits to QBF proof systems A general and long-standing belief in the proof complexity community asserts that there is a close connection between progress in lower bounds for Boolean circuits and progress in proof size lower bounds for strong propositional proof systems. Although there are famous examples where a transfer from ideas and techniques from circuit complexity to proof complexity has been effective, a formal connection between the two areas has never been established so far. Here we provide such a formal relation between lower bounds for circuit classes and lower bounds for Frege systems for *quantified Boolean formulas (QBF)*. Starting from a propositional proof system P we exhibit a general method how to obtain a QBF proof system $P+red$, which is inspired by the transition from resolution to Q -resolution. For us the most important case is a new and natural hierarchy of QBF Frege systems $C - Frege + red$ that parallels the well-studied propositional hierarchy of $C - Frege$ systems, where lines in proofs are restricted to a circuit class C . Building on earlier work for resolution (Beyersdorff, Chew, Janota STACS'15), we establish a lower bound technique via strategy extraction that transfers arbitrary lower bounds for the circuit class C to lower bounds in $C - Frege + red$. By using the full spectrum of state-of-the-art circuit lower bounds, our new lower bound method leads to very strong lower bounds for QBF Frege systems: (1) exponential lower bounds and separations for $AC_{[p]}^0 - Frege + red$ for all primes p ; (2) an exponential separation of $AC_{[p]}^0 - Frege + red$ from $TC^0 - Frege + red$; (3) an exponential separation of the hierarchy of constant-depth systems $AC_d^0 - Frege + red$ by formulas of depth independent of d . In the propositional case, all these results correspond to major open problems.

Ross Duncan (Strathclyde) Strong Complementarity in Quantum Computing Loosely speaking, a pair of quantum observables is called “complementary” when

knowledge of one implies ignorance of the other. Complementarity is responsible to much of the “weirdness” in quantum theory. The classic example is position and momentum, however finite dimensional examples such as the \mathcal{Z} and \mathcal{X} spins are used throughout quantum information processing. Thanks to a theorem of Coecke, Pavlovic and Vicary, quantum observables can be identified with certain Frobenius algebras; from this perspective complementary observables are those whose algebras satisfy some additional equations. For strongly complementary observables these equations have a succinct form: the Frobenius algebras jointly form a Hopf algebra. This purely algebraic characterisation belies their power: strongly complementary observables can be used for many purposes in quantum information processing, and as I will show, strong complementarity is at the heart of quantum non-locality.

Carl Feghali (Durham University) A reconfigurations analogue of Brooks’ theorem and its consequences (joint work with Matthew Johnson and Daniel Paulusma) Let \mathcal{G} be a simple undirected graph on n vertices with maximum degree d . Brooks’ Theorem states that \mathcal{G} has a d -colouring unless \mathcal{G} is a complete graph or a cycle with an odd number of vertices. To recolour \mathcal{G} is to obtain a new proper colouring by changing the colour of one vertex. We show an analogue of Brooks’ Theorem by proving that from any k -colouring, $k > d$, a d -colouring of \mathcal{G} can be obtained by a sequence of $O(n^2)$ recolourings using only the original k colours unless \mathcal{G} is a complete graph or a cycle with an odd number of vertices, or $k = d + 1$, \mathcal{G} is d -regular and, for each vertex v in \mathcal{G} , no two neighbours of v are coloured alike. We use this result to study the reconfiguration graph $R_k(\mathcal{G})$ of the k -colourings of \mathcal{G} . The vertex set of $R_k(\mathcal{G})$ is the set of all possible k -colourings of \mathcal{G} and two colourings are adjacent if they differ on exactly one vertex. We prove that for d at least 3, $R_d(\mathcal{G})$ consists of isolated vertices and at most one further component which has diameter $O(n^2)$. This result enables us to complete both a structural classification and an algorithmic classification for reconfigurations of colourings of graphs of bounded maximum degree.

Michael B. Gale (Cambridge University) Programming with Monadic Effect Hierarchies Effectful programming in purely-functional languages can be awkward to use when multiple effects are required at the same time. It is up to the programmer to set up a configuration of effects and to then correctly wire up their effectful computations so that they can run in a particular configuration. The more complicated the program, the more complicated is this process. Work in this area has largely focused on allowing programmers to write reusable code for arbitrary configurations of effects. We take a different point of view and propose a technique, inspired by object-oriented programming, for structuring programs around hierarchies of effects. Programmers using this technique write classes, consisting of functions for a particular effect configuration and extensions thereof, but do not have to worry about initialising or wiring them up. This has proved to work well

for monadic state, but our goal is to allow arbitrary effect configurations.

Bram Geron (University of Birmingham) Localised side-effects using binding handlers Sometimes, compilers can automatically remove or parallelise unused or independent sections of code, but for this the compiler must know that such sections are unused or independent. This is impossible to determine automatically in general. The programmer often does know such “side-effect information”: it is implicitly in her mind, sometimes even spelled out informally in documentation. However, this is useless for the compiler. Also, there is little protection against a co-worker modifying code elsewhere and accidentally breaking the documentation. We propose a more structured type of programming language, based on binding handlers. This is a new language primitive that “binds” all side-effects in the function that creates them. For instance, you might bind a new mutable variable, or a new static exception that you will handle. Functions are annotated with the enclosing bindings that they use, for instance what state they can modify. This can be useful documentation for co-workers, and its completeness is checked by the compiler. In practice, much side-effect-less code will be obviously side-effect-less. It should be easy for the compiler to check whether all exceptions will be handled somewhere, and to detect a class of unused or independent sections of code. Binding handlers are in their early stages. We have a monomorphic language with effects and handlers, a model, and axioms for optimisations, and we are finishing the soundness proofs. Our model describes how to evaluate programs, but there is no runnable implementation yet.

Tamas Kiespeter (Cambridge University) General Event Structures for Concurrent Games The theory of concurrency forms a fragmented picture when it comes to denotational semantics. One approach has been concurrent games underpinned by event structures. This approach appears to be well suited for various scenarios including probabilistic games, quantum mechanical and security models. Concurrent games modelled as prime event structures in particular have been robust under a number of extensions. However, prime event structures do not support probabilistic scenarios where parallel causes play a major role. These scenarios are expected for a unifying framework that these models set out to be. General event structures as applied to concurrent games appear to be an alternative worth exploring. They do support parallel causes. Most often the composition of strategies is realised by a form of synchronised parallel composition followed by an operation of hiding synchronisations. However this is not always the case. For example, in obtaining an operational semantics for strategies it is useful to have a composition without hiding. General event structures do not support hiding. Nevertheless, with this limitation, they have the potential to support a rich language of strategies, including probabilistic strategies, accompanied by an operational semantics.

Stefan Kuhn (Leicester University) A reversible process calculus for reactions

involving covalent bonding We introduce a simple process calculus with a new operator that allows us to model locally controlled reversibility. In our setting, actions can be undone spontaneously, as in other reversible process calculi, or as a part of pairs of the so-called concerted actions, where performing forwards a weak action forces undoing of another action, without the need of a global control or a memory. We model two examples from chemistry: the simple interaction of two water molecules and the hydration of formaldehyde in water into methanediol. We also present some properties of the calculus.

Andrew Lewis-Pye (LSE) Sex versus Asex The question as to why most higher organisms reproduce sexually has remained open despite extensive research, and has been called “the queen of problems in evolutionary biology.” Given the connections to optimisation problems and the improvement of genetic algorithms, this is also a question which has recently attracted interest in the computer science community. Theories dating back to Weismann have suggested that the key must lie in the creation of increased variability in offspring, causing enhanced response to selection. Rigorously quantifying the effects of assorted mechanisms which might lead to such increased variability, and establishing that these beneficial effects outweigh the immediate costs of sexual reproduction has, however, proved problematic. In recent work with Montalban, we introduced an approach which does not focus on particular mechanisms, influencing factors such as the fixation of beneficial mutants or the ability of populations to deal with deleterious mutations, but rather tracks the entire distribution of a population of genotypes as it moves across vast fitness landscapes. In this setting simulations show sex robustly outperforming asex across a broad spectrum of finite or infinite population models. Concentrating on the additive infinite populations model, we are able to give a rigorous mathematical proof establishing that sexual reproduction acts as a more efficient optimiser of mean fitness, thereby solving the problem for this model. Some of the key features of this analysis carry through to the finite population case.

Sepehr Meshkinfamfard (Durham University) Evolutionary Dynamics on Networks: An Introduction to the Inverse Moran Natural selection refers to the survival of individuals based on their fitness in a population. Evolutionary dynamics is used to study the evolution of homogeneous populations with no spatial structure, and the Moran Process is one of the most studied models. Given a homogeneous population with a single mutant individual, we consider the probability of the mutant descendants taking over the whole population; this probability is called the fixation probability. In Moran processes, the higher the fitness of the mutant, the greater the probability of fixation. Recently, Lieberman, Hauert and Nowak introduced the evolutionary graph theory, a generalisation of Moran processes in homogeneous populations to structured populations where individuals are arranged on a (connected) graph. In this setting the structure of the graph

plays an important role in the fixation process: fixation can be independent of fitness due to the structure of a graph, or the fixation state can be amplified in particular graphs. One important motivation behind structured Moran processes is the analysis of the behaviour of social networks, where some individuals are more influential than others, so studying how different individuals are influenced by their circle of friends is of importance. This is the motivation behind our approach, called the Inverse Moran process. Like structured Moran processes, our approach is also applied on a structured population; similarly, it starts by placing a mutant with fitness r in the population graph and ends with fixation or extinction of the mutant. However, the Inverse Moran iterative protocol is very different from the standard one; here, at each time step a vertex is replaced by the offspring of one of the neighbouring vertices chosen randomly based on their fitness.

Paulo Oliva (QMUL) Backward Induction and Unbounded Games (joint work with Martín Escardó) In this talk we present a generalisation of backward induction to well-founded games: those which terminate after a finite number of rounds, but whose length of play are not *a priori* bounded but can be arbitrarily long depending on how the game is played. We also allow for infinite sets of moves, so that the game tree can possibly be infinitely branching. Our definition of backward induction is completely formal, and relies on extensions of the simply typed λ -calculus. This is in stark contrast with current uses of backward induction in the literature, which allow for confusion when comparing different applications of the method to slightly different games. For instance, Boves writes that “there seems to be a similarity between the backward induction argument for the finite iterated prisoner’s dilemma and the surprise exam paradox and one cannot help but wonder whether the former is indeed no more than an instance of the latter.” We give a precise mathematical description of the backward induction method, together with a closed formula for the strategy profile in sub-game perfect equilibrium. A formal proof that the backward induction method leads to a strategy profile in sub-game perfect equilibrium has recently also been given by Aliprantis for finite games in extensive form. In our approach we consider games in normal form, but extend Aliprantis’ results in two ways: (a) We consider games whose game trees have infinitely many nodes but are nevertheless well-founded so that each play in the game eventually leads to an outcome. (b) We provide a closed formula describing the resulting optimal strategy profile.

Aris Pagourtzis (National Technical University of Athens) Reliable Message Transmission under Partial Knowledge and General Adversaries (joint work with Giorgos Panagiotakos and Dimitris Sakavalas) A fundamental primitive in distributed computing is *Reliable Message Transmission (RMT)*, which refers to the task of correctly sending a message from one party to another despite the presence of byzantine corruptions. In this work we address the problem in the general adversary model of Hirt and Maurer, which subsumes earlier models such as

the global or local threshold adversaries. Regarding the topology knowledge, we employ our recently introduced *Partial Knowledge Model* (DISC 2014), which encompasses both the full knowledge and the ad hoc model; the latter assumes knowledge of the local neighborhood only. Our main contributions are: (a) A necessary and sufficient condition for achieving *RMT* in the partial knowledge model with a general adversary; in order to show sufficiency, we propose *RMT-PKA*, a protocol that solves *RMT* whenever this is possible, therefore it is a unique protocol (Pelc, Peleg, IPL 2005). To the best of our knowledge, this is the first unique protocol for *RMT* against general adversaries in the partial knowledge model. (b) A study of efficiency in the case of the ad hoc network model: we show that either the *Z-CPA* protocol (DISC 2014) is fully polynomial or no unique fully polynomial protocol for *RMT* exists, thus introducing a new notion of uniqueness with respect to efficiency that we call poly-time uniqueness. To obtain our results we introduce, among others, a joint view operation on adversary structures, a new notion of separator (*RMT-cut*) appropriate for *RMT* in unreliable networks, and a self-reducibility property of the *RMT* problem, which we show by means of a protocol composition. The latter plays a crucial role in proving the poly-time uniqueness of *Z-CPA*.

Gadi Tellez (UCL) Temporal Separation Logic We present a proof system for formally verifying temporal properties of heap manipulating programs. Judgments in this system express a temporal property of a program when started from a given computational state satisfying a given precondition, which is expressed as a formula in separation logic. Proofs in our system are cyclic proofs: cyclic derivations in which some temporal operator is unfolded infinitely often along every infinite path, allowing us to discard these by an infinite descent argument. We implement our system in the Cyclist theorem prover framework and evaluate it on a variety of examples taken from the Windows Update System and the backend of the PostgreSQL database manager system. The results obtained show the viability of our approach in practical examples.