17th Wessex Theory Seminar

17th Wessex Theory Seminar

The seventeenth meeting of the Wessex Theory Seminar will take place at Queen Mary, University of London on 20th September 2012.

Venue

The talks will be held in room BR3.01 of QMUL http://goo.gl/maps/vBkgo

Programme

11:00 coffee at the Hub, next to the conference room

11:30 **Radu Grigore** (Queen Mary), Register Automata and Java

12:15 **Stefan Kiefer** (Oxford), On the equivalence problem for probabilistic automata

13:00 lunch (not provided, but we suggest the Curve or Drapers)

14:15 **Tony Tan** (Warsaw), An Automata Model for Trees with Ordered Data Values

15:00 **Nobuko Yoshida** (Imperial), Multiparty Session Automata and their application in large distributed systems

15:45 coffee at the Hub

16:15 **Jules Villard** (UCL), The Ramifications of Sharing in Data Structures

17:00 **Nick Benton** (Microsoft Research), High-Level Separation Logic for Low-Level Code

17:45 pub/dinner

Abstracts

Radu Grigore

Register Automata and Java

Is it possible to use register automata to specify Java programs with dynamic allocation of resources? Intuitively the answer should be affirmative. Register automata denote languages over infinite alphabets, and program executions are traces of events, which may mention an unbounded number of resources. However, details sometimes hide devils. Would such specifications be convenient and useful? I will try to convince you that the answer is indeed affirmative, with examples and details. I introduce TOPL automata, which are equally expressive to register automata, but more convenient for this task. I will demonstrate how they are used for runtime verification. Finally, I will show some preliminary work on how to use TOPL automata for static analysis. This is joint work with Dino Distefano, Rasmus Lerchedahl Petersen, and Nikos Tzevelekos.

Stefan Kiefer

On the equivalence problem for probabilistic automata

Deciding equivalence of probabilistic automata is a key problem for establishing various behavioural and anonymity properties of probabilistic systems. In particular, it is at the heart of the tool APEX, an analyser for probabilistic programs. APEX is based on game semantics and analyses a broad range of anonymity and termination properties of randomised protocols and other open programs.

In recent experiments a randomised equivalence test based on polynomial identity testing outperformed deterministic algorithms. We show that polynomial identity testing yields efficient algorithms for various generalisations of the equivalence problem. First, we provide a randomized NC procedure that also outputs a counterexample trace in case of inequivalence. Second, we consider equivalence of probabilistic cost automata. In these automata transitions are labelled with integer costs and each word is associated with a distribution on costs, corresponding to the cumulative costs of the accepting runs on that word. Two automata are equivalent if they induce the same cost distributions on each input word. We show that equivalence can be checked in randomised polynomial time. Finally we show that the equivalence problem for probabilistic visibly pushdown automata is logspace equivalent to the problem of whether a polynomial represented by an arithmetic circuit is identically zero.

Tony Tan

An Automata Model for Trees with Ordered Data Values

Data trees are trees in which each node, besides carrying a label from a finite alphabet, also carries a data value from an infinite domain. They have been used as an abstraction model for reasoning tasks on XML and verification. However, most existing approaches consider the case where only equality test can be performed on the data values.

In this paper we study data trees in which the data values come from a linearly ordered domain, and in addition to equality test, we can test whether the data value in a node is greater than the one in another node. We introduce an automata model for them which we call ordered-data tree automata (ODTA), provide its logical characterisation, and prove that its emptiness problem is decidable in 3-NEXPTIME. We also show that the two-variable logic on unranked trees, studied by Bojanczyk, Muscholl, Schwentick and Segoufin in 2009, corresponds precisely to a special subclass of this automata model.

Then we define a slightly weaker version of ODTA, which we call weak ODTA, and provide its logical characterisation. The complexity of the emptiness problem drops to NP. However, a number of existing formalisms and models studied in the literature can be captured already by weak ODTA. We also show that the definition of ODTA can be easily modified, to the case where the data values come from a tree-like partially ordered domain, such as strings.

Nobuko Yoshida

Multiparty Session Automata and their application in large distributed systems

Communicating finite state machines (CFSMs for short) abstract processes which communicate by asynchronous exchanges of messages via FIFO channels. Their major impact has been in characterising essential properties of communications such as freedom from deadlock and communication error, and buffer boundedness. CFSMs are known to be computationally hard: most of these properties are undecidable even in restricted cases. On the other hand, multiparty session types are a recent typed framework whose main features are its ability to efficiently enforce these properties for mobile processes and programming languages.

This talk ties the links between the two frameworks to achieve a two-fold goal. On one hand, we present a generalised variant of multiparty session types that have a direct semantical correspondence to CFSMs. Our calculus can treat expressive forking, merging and joining protocols that are absent from existing frameworks, and the typing system can ensure properties such as safety, boundedness and liveness on distributed processes in polynomial time.

On the other hand, multiparty session types generate a new class of CFSMs that automatically enjoy the aforementioned properties. This solves an open question on CFSMs, generalising Gouda et al's work in 1984, which covered the two-machine case and presented polynomial time algorithms.Our framework works with an arbitrary number of machines, still offering a polynomial time algorithm.

I also talk about a summary of our recent collaborations based on multiparty session automata, with industry partners and a major, long-term, NSF-funded program which provides a ultra large scale cyberinfrustracture for 25-30 years of sustained ocean measurements to study climate variability, ocean circulation and ecosystem dynamics.

Jules Villard

The Ramifications of Sharing in Data Structures

Programs manipulating mutable data structures with intrinsic sharing present a challenge for modular verification. Deep aliasing inside data structures dramatically complicates reasoning in isolation over parts of these objects because changes to one part of the structure (say, the left child of a dag node) can affect other parts (the right child or some of its descendants) that may point into it. The result is that finding intuitive and compositional proofs of correctness is usually a struggle. We propose a compositional proof system that enables local reasoning in the presence of sharing.

While the AI "frame problem" elegantly captures the reasoning required to verify programs without sharing, we contend that natural reasoning about programs with sharing instead requires an answer to a different and more challenging AI problem, the "ramification problem": reasoning about the indirect consequences of actions. Accordingly, we present a Ramify proof rule that attacks the ramification problem head-on and show how to reason with it. Our framework is valid in any separation logic and permits sound compositional and local reasoning in the context of both specified and unspecified sharing. This talk will be illustrated by proofs of examples manipulating dags, graphs, and overlaid data structures.

Nick Benton

High-Level Separation Logic for Low-Level Code

Separation logic is a powerful tool for reasoning about structured, imperative programs that manipulate pointers. However, its application to unstructured, lower-level languages such as assembly language or machine code remains challenging. In this paper we describe a separation logic tailored for this purpose that we have applied to x86 machine code programs.

The logic is built from an assertion logic on machine states over which we construct a specification logic that encapsulates uses of frames and step indexing. The traditional notion of Hoare triple is not applicable directly to unstructured machine code, where code and data are mixed together and programs do not in general run to completion, so instead we adopt a continuation-passing style of specification with preconditions alone. Nevertheless, the range of primitives provided by the specification logic, which include a higher-order frame connective, a novel read-only frame connective, and a `later¿ modality, support the definition of derived forms to support basic-block-style reasoning for common cases, in which standard rules for Hoare triples are derived as lemmas. Furthermore, our encoding of assembly language labels in terms of the more primitive code pointers lets us encapsulate local usage of labels and the definition and rules for assembly-language `macros¿ such as while loops and conditionals.

We have applied the framework to a model of x86 machine code built entirely within the Coq proof assistant, including tactic support based on computational reflection.

This is joint work with Jonas Jensen and Andrew Kennedy.

Attendance

From Bath:

- John Power
- Cai Wingfield

From Birmingham:

- Liang-Ting Chen
- Martin Escardo
- Olle Fredriksson
- Dan Ghica
- Chuangjie Xu

From Imperial:

- Runa Jesmin
- Nobuko Yoshida

From Microsoft Research:

- Nick Benton

From Oxford:

- Stefan Kiefer

From Queen Mary:

- Yasin Anbar
- Rob Arthan
- Tzu-Chun Chen
- Nhat Anh Dang
- Dino Distefano
- Luca Fossati
- Radu Grigore
- Jules Hedges
- Paulo Oliva
- Quoc San Phan
- Edmund Robinson
- Nikos Tzevelekos
- Graham White

From Rennes:

- Delphine Demange

From University College London:

- Jules Villard

From Warsaw:

- Tony Tan