

5th Wessex Theory Seminar

Tuesday 8th December 2009

VENUE: Lecture Theatre B (Level 1), Oxford University Computing Laboratory (OUCL), Wolfson Building, Parks Road, Oxford

[Map](#)

Programme

12:00 Depart for lunch from the OUCL Reception

2:00--2:40: Nathan Bowler (University of Cambridge)

2:45--3:25: Martin Churchill (University of Bath)

3:30--4:00 **coffee break**

4:00--4:40: James Laird (University of Bath)

4:40--5:40 **coffee break**

Attendance

From the Wessex Theory Seminar sites:

Bath: Martin Churchill, James Laird, Ana Martins

Cambridge: Nathan Bowler, Martin Hyland

Queen Mary, University of London: Paulo Oliva

Oxford: Christopher Broadbent, Matthew Hague, David Hopkins, Andrzej Murawski, Luke Ong, Stephen Ramsay, Jamie Vicary

Southampton: Toby Wilkinson

Swansea: Ulrich Berger, Oliver Kullmann

In addition:

Birmingham: Dan Ghica

Abstracts

[Nathan Bowler: Fc-multicategories as a framework for the categorical understanding of games.](#)

A common intuitive construction of categories of games encodes the basic combinatorics involved in various approaches to the study of games, for example Conway's recursively structured theory, or the specialised theory of hypergraph games. These ideas have provided some key examples of models of (various fragments of) intuitionistic and linear logic. Such categories have also been used to construct denotational semantics for various abstract programming languages, including the construction of a fully abstract model for PCF.

I'll begin by outlining an example of such a category (based on bicoloured digraphs) which allows the incorporation of some of the structure of the theory of hypergraph games. I'll explain how this construction points to the language of fc-multicategories as a natural setting for the development of additional structure in categories of games. This language also provides a new setting for the construction of existing categories of games. I'll illustrate how the constructions in this setting have a modular form, cleanly separating different aspects of the underlying combinatorics, and I'll sketch some possible applications to the development of new constructions.

[Martin Churchill: A Logic of Sequentiality](#)

Many logics and programming languages can be equipped with game semantics, whereby programs of type A (or proofs of proposition A) are represented by strategies over the game representation of A. In the case of proof semantics, the strategies in question are total. These games models often admit definability results — for a given system, we might provide some constraints on strategies such that each strategy satisfying said constraints is the denotation of some program (or proof.) In this work we consider a relatively constraint-free games model (in which one can embed many other games models, albeit a finitary version). We develop a proof system for this model that is "fully complete" in a strong sense: proofs of a formula are in 1-1 correspondence with total strategies over the game representation of that formula. Thus we can use the logic to represent strategies syntactically, and thus use it to model proofs in Multiplicative Linear Logic, imperative objects, and so on. The logic is polarised and contains many standard connectives (tensors, with, par, exponentials...) although the proof rules themselves are nonstandard — a key notion is that of a sequoid operator from Lai02, identifying the location of the next move (or focus) of the formula.

[James Laird: Games and Full Abstraction for Programming Languages with Generic Polymorphism](#)

We describe a setting in which rather simple and concrete games models for programming languages with higher-rank polymorphic types can be given. It is based on Hyland-Ong style games, in which languages with many computational features have been interpreted, and related techniques for model-checking them developed. In particular, we describe a model of a language with polymorphism and general references which is fully abstract, and also effectively presentable.

Closed types in the model denote HO-style arenas with an additional enabling relation on moves. Open types denote endofunctors on these arenas which are in essence "arenas with (explicit) holes", so that their action is given by plugging arenas and strategies into this hole. These form a (closed Freyd) category, in which morphisms are natural transformations which play copycat between the holes. Universal quantification plugs the one-move arena into each hole, and uses the new enabling structure to specify the copycat links. Proving full abstraction (and definability of finitary strategies as terms) is surprisingly simple based on a kind of "internal genericity" property.