Preface

A general abstract theory for computation involving shared resources is presented. We develop the models of *sharing graphs*, also known as term graphs, in terms of both syntax and semantics.

According to the complexity of the permitted form of sharing, we consider four situations of sharing graphs. The simplest is first-order acyclic sharing graphs represented by let-syntax, and others are extensions with higher-order constructs (lambda calculi) and/or cyclic sharing (recursive letrec binding). For each of four settings, we provide the equational theory for representing the sharing graphs, and identify the class of categorical models which are shown to be sound and complete for the theory. The emphasis is put on the algebraic nature of sharing graphs, which leads us to the semantic account of them.

We describe the models in terms of the notions of symmetric monoidal categories and functors, additionally with symmetric monoidal adjunctions and traced monoidal categories for interpreting higher-order and cyclic features. The models studied here are closely related to structures known as notions of computation, as well as models for intuitionistic linear type theory. As an interesting implication of the latter observation, for the acyclic settings, we show that our calculi conservatively embed into linear type theory. The models for higher-order cyclic sharing are of particular interest as they support a generalized form of recursive computation, and we look at this case in detail, together with the connection with cyclic lambda calculi.

We demonstrate that our framework can accommodate Milner's *action calculi*, a proposed framework for general interactive computation, by showing that our calculi, enriched with suitable constructs for interpreting parameterized constants called controls, are equivalent to the closed fragments of action calculi and their higher-order/reflexive extensions. The dynamics, the computational counterpart of action calculi, is then understood as rewriting systems on our calculi, and interpreted as local preorders on our models.

Preface to the Present Edition

This book contains the author's PhD thesis written under the supervision of Rod Burstall (first supervisor), Philippa Gardner and John Power (second supervisors) at Laboratory for Foundations of Computer Science, University of Edinburgh. The thesis was examined by Martin Hyland (Cambridge) and Alex Simpson (Edinburgh). Except for correcting minor mistakes and updating the bibliographic information, the text agrees with the examined version of the thesis.

Some parts of the book have been published elsewhere in [13, 35, 38]. Since the examination of the thesis, a number of works related to this research have appeared. I take this opportunity to mention some of them.

- An independent work by Corradini and Gadducci [25] used essentially the same categorical structure described in Chapter 3 for modeling acyclic graph rewriting systems (with Cat-enrichment rather than Preord-enrichment). Miyoshi [70] translated the results in Chapter 6 to their setting and reformulated the cyclic sharing theories as a rewriting logic.

- While the model construction techniques in Chapter 5 show the conservativity of syntactic translations, further techniques for showing the fullness (or full completeness) of the translations have been developed by the author, as reported in [39].

- A direction progressing rapidly is the investigation of traced monoidal categories as a foundation of recursive computation, as claimed in Chapter 7. Some fundamental issues on traced monoidal categories are studied in Abramsky, Blute and Panangaden [4] and Blute, Cockett and Seely [23]; the latter contains a fixpoint theorem related to those in Chapter 7. As an interesting case study, Ryu Hasegawa [40] related the fixpoint operator in a model of (typed and untyped) lambda calculus and the Lagrange-Goodman inversion formula in enumerative combinatorics in terms of trace. The relation to axiomatic domain theory has been studied by Plotkin and Simpson [74].

- In Chapter 9 the possibility of developing the premonoidal variant of the sharing theories and their models was suggested. Related to this, Jeffrey [46] has introduced a semantics of the graphically-presented imperative programs based on premonoidal categories. In that setting, he also modeled recursion using trace.

Acknowledgements

I want to express my heartfelt thanks to my supervisors; I can never thank them enough.

Rod Burstall, my first supervisor, always helped me to think constructively and positively, especially at difficult moments throughout my PhD study in Edinburgh. I will never forget a meeting with Rod when I had a bad hangover – there I got an essential inspiration in deciding my research direction.

Philippa Gardner, my second supervisor during the second year, always gave me enthusiastic encouragement, and I benefited from countless delightful (often overheated) discussions with her.

John Power has always been an important intellectual source and often a mentor for me during these three years, and he became my second supervisor after Philippa moved to Cambridge. Without his generous and much needed support, this thesis would probably never have been written in this form.

At a stimulating place like LFCS, even a brief chat often meant a lot to me. I am grateful to people who influenced me in various forms, especially to Andrew Barber, Ewen Denney, Marcelo Fiore, Alex Mifsud, Robin Milner, Gordon Plotkin and Alex

Simpson. In particular, Chapter 5 and Chapter 8 refer to joint work with Andrew, Philippa and Gordon.

I want to thank Martin Hyland for helpful discussions on traced monoidal categories as well as for his warm encouragement. Thanks are also due to Zena Ariola and Stefan Blom, for e-mail communications on cyclic lambda calculi and related topics.

And, above all, many, many thanks go to my cheerful, lovely, friends.

This work was partly supported by an Oversea Research Student award.

Contents