Math. Struct. in Comp. Science (2007), *vol.* 17, *pp.* 361–362. © 2007 Cambridge University Press
doi:10.1017/S0960129507006081 Printed in the United Kingdom

Theory and applications of term graph rewriting: introduction

IAN MACKIE[†] and DETLEF PLUMP[‡]

[†]*King's College London, Department of Computer Science, Strand, London WC2R 2LS, U.K. and LIX, École Polytechnique, 91128 Palaiseau Cedex, France*
[‡]*Department of Computer Science, The University of York, Heslington, York YO10 5DD, U.K.*

Received 15th September 2006

Term graph rewriting is concerned with the representation of functional expressions as graphs and the evaluation of these expressions by rule-based graph transformation. The advantage of computing with graphs rather than terms is that common subexpressions can be shared, improving the efficiency of computations in space and time. Sharing is ubiquitous in implementations of programming languages: many functional, logic, object-oriented and concurrent calculi are implemented using term graphs.

Research in term and graph rewriting ranges from theoretical questions to practical implementation issues, and specific areas of interest include: the modelling of first- and higher-order term rewriting by (acyclic or cyclic) graph rewriting, the use of graphical frameworks such as interaction nets and sharing graphs to model strategies of evaluation (for instance, optimal reduction in the λ-calculus), rewrite calculi on cyclic higher-order term graphs for the semantics and analysis of functional programs, graph reduction implementations of programming languages, graphical calculi modelling concurrent and mobile computations, object-oriented systems, graphs as a model of biological or chemical abstract machines, and automated reasoning and symbolic computation systems working on shared structures.

The TERMGRAPH series of workshops covering these topics was started in 2002, with the aim of bringing researchers working in these different domains together. This special issue of Mathematical Structures in Computer Science collects a number of papers together to give a snapshot of the current research in the area, where, in particular, we encouraged authors of previous TERMGRAPH events to submit full versions of their papers. This volume is dedicated to the theory and applications of term graph rewriting, and came out of an open call for papers. Five articles were selected for this special issue, covering a range of topics – the following paragraphs give a brief introduction to each of them.

A rewriting calculus for cyclic higher-order term graphs by Paolo Baldan, Clara Bertolissi, Horatiu Cirstea and Claude Kirchner is concerned with a version of the ρ-calculus (also known as the rewriting calculus) with recursion equations. This work can be understood as a mixture of the usual ρ-calculus and the cyclic λ-calculus of Ariola and Klop. The main contributions of the paper are the definition of the cyclic ρ-calculus and a proof of the confluence of the calculus under certain constraints (the algebraic patterns are linear).

Graph rewriting for the π-calculus by Fabio Gadducci describes an encoding of π-calculus processes into graph rewriting, where reduction steps of the calculus are mimicked by graph rewriting rules. In addition to the basic π-calculus constructs, the encoding can also handle, with some additional rewrite rules, sums and recursion. This paper offers the use of standard graph rewriting mechanisms for modelling the reduction semantics of the π-calculus.

Natural deduction via graphs: formal definition and computation rules by Herman Geuvers and Iris Loeb revisits natural deduction by extending and generalising the formalisms introduced by Gentzen-Prawitz and Fitch to give *deduction graphs*. These are directed graphs with boxes, which make the sharing of subtrees explicit. The paper provides a thorough investigation of this formalism, and concludes with a comparison with linear logic proof nets.

A duality between proof systems for cyclic term graphs by Clemens Grabmayer studies different presentations of equality for first-order recursive types, and establishes a form of duality between two different approaches to prove the equivalence of certain kinds of finite syntactic representations of rational trees. From this, equality can be defined syntactically by induction or coinduction. Finally, the paper considers the representation of recursive types, and reproduces the same results for cyclic term graphs.

A denotational semantics for the symmetric interaction combinators by Damiano Mazza offers a denotational semantics for Lafont's interaction combinators. This system of rewriting has three combinators and six rewrite rules. The semantics is invariant under reduction as expected, but can also be used to validate a number of other transformations such as permutation and η-rules, and helps pave the way towards a semantics of general interaction nets.

Acknowledgements

We would like to thank the authors who contributed to this special issue, and the referees who worked diligently to evaluate and help improve the papers. Very special thanks are due to Giuseppe Longo, the editor-in-chief of *Mathematical Structures in Computer Science*, for his constant support, which made this issue possible.