HOR 2014
7th International Workshop on
Higher-Order Rewriting

8.45–10.15: Session I – Calculi

Chair: Cynthia Kop

8.45: Vincent van Oostrom (Utrecht Univ.), Femke van Raamsdonk (VU Univ., Amsterdam)
The dynamic pattern calculus as a higher-order pattern rewriting system

We show that Jay and Kesners dynamic pattern calculus can be embedded into a higher-order pattern rewriting systems in the sense of Nipkow. Metatheoretical results, such as confluence and standardisation, are obtained for the dynamic pattern calculus as a consequence of this embedding. The embedding also opens a way to study the semantics of Jays programming language *bondi* based on pattern matching.

9.15: Beniamino Accattoli (Univ. di Bologna)
Distilling Abstract Machines
(joint work with Pablo Barenbaum and Damiano Mazza)

It is well-known that many environment-based abstract machines can be seen as strategies in lambda calculi with explicit substitutions (ES). Recently, graphical syntaxes and linear logic led to the linear substitution calculus (LSC), a new approach to ES that is halfway between big-step calculi and traditional calculi with ES. This paper studies the relationship between the LSC and environment-based abstract machines. While traditional calculi with ES simulate abstract machines, the LSC rather distills them: some transitions are simulated while others vanish, as they map to a notion of structural congruence. The distillation process unveils that abstract machines in fact implement weak linear head reduction, a notion of evaluation having a central role in the theory of linear logic. We show that such a pattern applies uniformly in call-by-name, call-by-value, and call-by-need, catching many machines in the literature (as the KAM, the CEK, the ZINC, *etc.*), we show that distillation preserves the time complexity of the executions, *i.e.*, the LSC is a complexity-preserving abstraction of abstract machines.

9.45: Kristoffer Rose (Two Sigma, New York)
Experience with Higher Order Rewriting from the Compiler Teaching Trenches

I have now twice used the "HACS" compiler generator tool in a New York Univ. graduate Compiler Construction class. HACS is based on a higher-order rewriting formalism, thus I have effectively been teaching students higher order rewriting techniques as the way to implement compilers. In this talk I report on how HACS matches specific rewriting notions to the techniques used by compiler writers, and where the main difficulties have been encountered in teaching these.

10.15–10.45: Coffee Break

10.45–12.00: Session II – Foundations

Chair: Beniamino Accatoli

10.45: Cynthia Kop (Univ. of Innsbruck)
The Higher-order Dependency Pair Framework

In recent years, two different dependency pair approaches have been introduced: the *dynamic* and *static* styles. The static style is based on a computability argument, and is limited to *plain function-passing* systems. The dynamic style has no limitations, but standard techniques to simplify sets of dependency pairs – such as the subterm criterion and usable rules – are not applicable. The basic *reduction pair* technique is also significantly weaker than in the static case. On the other hand, when the system is *left-linear* and *left-abstraction-free*, we can significantly improve the dynamic approach. In this talk, I will discuss how to combine the dynamic and static styles in a single dependency pair *framework*, extending the various notions from the first-order dependency pair framework.

11.15: Vincent van Oostrom (Utrecht Univ.)
Feebly not weakly

Some rewrite systems are not orthogonal, in that they do have critical peaks, but are very close to being orthogonal, in that for any given object there exists a partial function, called *orthogonalisation*, mapping the set of all redexes to an orthogonal subset and every multi-step to an equivalent one. Term rewrite systems having only trivial peaks, so-called weakly orthogonal systems with the $\lambda\beta\eta$-calculus as prime example, are known to admit such an orthogonalisation. Here we characterise the term rewrite systems that admit orthogonalisation as those whose critical peaks are *feeble*, in that at least two out of the three terms in such a peak must be identical (generalising weak orthogonality).

11.45: Kristoffer Rose (Two Sigma)
Report from the HOR 2014 Chair & Discussion

12.00–13.00: Joint HOR and WIR Invited Talk

Chairs: Kristoffer Rose and Hans Zantema

12.00: Damiano Mazza (Univ. Paris 13)
On Infinitary Affine Lambda-Calculi

We summarize recent work based on affine lambda-calculi which brings together some aspects of infinitary and higher-order rewriting. In particular, we discuss three points: the relationship with the (infinitary) lambda-calculus; the categorical perspective, which gives a way of building models of linear logic; and the applications to implicit computational complexity, in particular the possiblity of representing non-uniform polytime computation in a functional, higher-order setting.