The Bisimulation Proof Method: Enhancements and Open Problems

Davide Sangiorgi

Università di Bologna, Italy
<http://www.cs.unibo.it/~sangio/>

Bisimulation (and, more generally, co-induction) can be regarded as one of the most important contributions of Concurrency Theory to Computer Science. Nowadays, bisimulation and the co-inductive techniques developed from the idea of bisimulation are widely used, not only in Concurrency, but, more broadly, in Computer Science, in a number of areas: functional languages, object-oriented languages, type theory, data types, domains, databases, compiler optimisations, program analysis, verification tools, etc.. For instance, in type theory bisimulation and co-inductive techniques have been used: to prove soundness of type systems; to define the meaning of equality between (recursive) types and then to axiomatise and prove such equalities; to define co-inductive types and manipulate infinite proofs in theorem provers. Also, the development of Final Semantics, an area of Mathematics based on co-algebras and category theory and that gives us a rich and deep perspective on the meaning of co-induction and its duality with induction, has been largely motivated by the interest in bisimulation.

In my talk I will discuss the bisimulation proof method – an instance of the co-induction proof method – that is at the heart of the success of bisimulation. I will discuss a number of enhancements of the method and some open problems.

The objective of enhancements is to relieve the work involved with the bisimulation proof method. Thus proving a bisimulation result becomes simpler. Such enhancements can sometimes be extremely important. They seem to be even *essential* in calculi for mobility such as the π -calculus [Mil99, SW01], and in higher-order languages (that is, languages where substitutions can involve the replacement of variables with arbitrary terms of the language) such as Higher-Order π -calculus [San92], Ambients [CG98], or even sequential languages such as the λ -calculus.

References

- [CG98] L. Cardelli and A.D. Gordon. Mobile ambients. In Nivat. M., editor, *Proc. FoSSaCS '98*, volume 1378 of *Lecture Notes in Computer Science*, pages 140–155. Springer Verlag, 1998.
- [Hir99] D. Hirschhoff. *Mise en oeuvre de preuves de bisimulation*. PhD thesis, PhD Thesis, Ecole Nationale des Ponts et Chausses, 1999.
- [HPS05] Daniel Hirschhoff, Damien Pous, and Davide Sangiorgi. A correct abstract machine for safe ambients. In *COORDINATION*, pages 17–32, 2005.

- [Mil89] R. Milner. *Communication and Concurrency*. Prentice Hall, 1989.
- [Mil99] R. Milner. *Communicating and Mobile Systems: the π -Calculus*. Cambridge University Press, 1999.
- [MZN05] M. Merro and F. Zappa Nardelli. Behavioural theory for mobile ambients. *Journal of the ACM*. To appear, 2005.
- [Pou05] Damien Pous. Up-to techniques for weak bisimulation. In *ICALP*, pages 730–741, 2005.
- [San98] D. Sangiorgi. On the bisimulation proof method. *Journal of Mathematical Structures in Computer Science*, 8:447–479, 1998.
- [SM92] D. Sangiorgi and R. Milner. The problem of “Weak Bisimulation up to”. In W.R. Cleveland, editor, *Proc. CONCUR '92*, volume 630 of *Lecture Notes in Computer Science*, pages 32–46. Springer Verlag, 1992.
- [San92] D. Sangiorgi. *Expressing Mobility in Process Algebras: First-Order and Higher-Order Paradigms*. PhD thesis CST-99-93, Department of Computer Science, University of Edinburgh, 1992.
- [SW01] D. Sangiorgi and D. Walker. *The π -calculus: a Theory of Mobile Processes*. Cambridge University Press, 2001.