Nominal SOS

MohammadReza Mousavi[1,2], Michel A. Reniers[1], Murdoch J. Gabbay[3]

[1] Department of Computer Science,
Eindhoven University of Technology, The Netherlands

[2] School of Computer Science,
Reykjavík University, Iceland

[3] Computer Science Department,
Heriot-Watt University, United Kingdom

Concepts of names, variables, meta-variables, and name abstraction (binding) are the bread-and-butter of contemporary process algebras and calculi. For example, in the π-calculus [12], names are first-class citizens and the whole calculus is built around the notion of passing names among concurrent agents. These phenomena appear also in specifications involving the recursion operator, the infinite sum operator, and the time-integration operator (cf. [12], [14], and [2], respectively).

Structural Operational Semantics (SOS) [13] has been widely used to give operational semantics to formalisms and process calculi. Given the presence of names and binders in different formalisms and calculi, it is useful to accommodate these (often informal) notations inside the SOS framework. A feature of such a framework is that name abstraction may have process meta-variables in its scope. We propose a formal framework for names in SOS, based on the Nominal Techniques of Gabbay and Pitts [7, 15].

Similar work exists in [16] and [5]. The work of [16] is based on the $FO\lambda^{\Delta\nabla}$ by Miller and Tiu [10] and extends the tyft format of [9] with names and binders. The work of [5] is based on the categorical model of abstract syntax with names and binders as proposed by [4, 6] and extends the de Simone format of [3] by adding the support for names.

Apart from the fundamental differences stemming from the underlying treatment of names and binders, our framework extends that of [5] and [16] with negative premises, and that of [5] with arbitrary source terms for the premises.

Following the SOS tradition [1] we also introduce congruence rule formats for Nominal SOS. We extend the ntyft format of [8] with names and binders and then add more syntactic ingredients. We further study the addition of equational specification, or structural congruences as advocated by Milner [11, 12], to the operational framework with names and binders.

References

[1] Luca Aceto, Willem Jan (Wan) Fokkink, and Chris Verhoef. Structural operational semantics. In Jan A. Bergstra, Alban Ponse, and Scott A. Smolka, editors, *Handbook of Process Algebra, Chapter 3*, pages 197–292. Elsevier Science, Dordrecht, The Netherlands, 2001.

[2] J.C.M. (Jos) Baeten and Cornelis A. (Kees) Middelburg. *Process Algebra with Timing.* EATCS Monographs. Springer-Verlag, Berlin, Germany, 2002.

[3] Robert de Simone. Higher-level synchronizing devices in MEIJE-SCCS. *Theoretical Computer Science (TCS)*, 37:245–267, 1985.

[4] Marcelo P. Fiore, Gordon D. Plotkin, and Daniele Turi. Abstract syntax and variable binding. In *Proceedings of the 14th Annual IEEE Symposium on Logic in Computer Science (LICS'99)*, pages 193–202. IEEE Computer Society, Los Alamiots, CA, USA, 1999.

[5] Marcelo P. Fiore and Sam Staton. A congruence rule format for name-passing process calculi from mathematical structural operational semantics. In *Proceedings of the 21st Annual IEEE Symposium on Logic in Computer Science (LICS'06)*. IEEE Computer Society Press, Los Alamitos, CA, USA, 2006. To appear.

[6] Marcelo P. Fiore and Daniele Turi. Semantics of name and value passing. In *Proceedings of the 16th Annual IEEE Symposium on Logic in Computer Science (LICS'01)*, pages 93–104. IEEE Computer Society, Los Alamiots, CA, USA, 2001.

[7] Murdoch J. Gabbay and Andrew M. Pitts. A new approach to abstract syntax with variable binding. *Formal Aspects of Computing (FAC)*, 13(3–5):341–363, 2002.

[8] Jan Friso Groote. Transition system specifications with negative premises. *Theoretical Computer Science (TCS)*, 118(2):263–299, 1993.

[9] Jan Friso Groote and Frits W. Vaandrager. Structured operational semantics and bisimulation as a congruence. *Information and Computation (I&C)*, 100(2):202–260, October 1992.

[10] Dale Miller and Alwen Tiu. A proof theory for generic judgments. *ACM Transactions on Computational Logic*, 6(4):749–783, 2005.

[11] A.J.R.G. (Robin) Milner. Functions as processes. *Mathematical Structures in Computer Science*, 2:119–141, 1992. An earlier version of this paper appeared as Technical Report N.1154 of INRIA, Sophia-Antipolis, 1990.

[12] A.J.R.G. (Robin) Milner. The polyadic π-calculus: a tutorial. In Friedrich L. Bauer, Wilfried Brauer, and Helmut Schwichtenberg, editors, *Logic and Algebra of Specification*, pages 203–246. Springer-Verlag, 1993. An earlier version of this paper appeared as Technical Report ECS-LFCS-91-180 of University of Edinburgh, 1991.

[13] Gordon D. Plotkin. A structural approach to operational semantics. *Journal of Logic and Algebraic Progamming (JLAP)*, 60:17–139, 2004. This article first appeared as Technical Report DAIMI FN-19, of Aarhus University, 1981.

[14] Michel A. Reniers, Jan Friso Groote, Mark B. van der Zwaag, and Jos van Wamel. Completeness of timed μCRL. *Fundamenta Informaticae*, 50(3-4):361–402, 2002.

[15] Christian Urban, Andrew M. Pitts, and Murdoch J. Gabbay. Nominal unification. In Matthias Baaz, editor, *Proceedings of the 12th Annual Computer Science Logic Conference and 8th Kurt Godel Colloquium (CSL'03 & KGC)*, volume 2803 of *Lecture Notes in Computer Science*, pages 513–527. Springer-Verlag, Berlin, Germany, 2003.

[16] Axelle Ziegler, Dale Miller, and Catuscia Palamidessi. A congruence format for name-passing calculi. In *Proceedings of the 2nd Workshop on Structural Operational Semantics (SOS'05)*, Electronic Notes in Theoretical Computer Science, Lisbon, Portugal. Elsevier Science B.V., 2005. To appear.