

Jeffrey, Alan; Rathke, Julian

A fully abstract may testing semantics for concurrent objects. (English) Zbl 1078.68107
Theor. Comput. Sci. 338, No. 1-3, 17-63 (2005).

Object calculus is a minimal language for modelling object-based programming. Concurrent object calculus is a version of it dealing with concurrent objects. On this base the authors define a may testing preorder for concurrent object components. It is characterised by a trace semantics inspired by UML interaction diagrams and some of its properties are studied. It is shown that the trace semantics is fully abstract for may testing. This is the first result of this kind for a concurrent object language. In addition, the paper contains interesting directions for future research.

Reviewer: [Krassimir Atanassov \(Sofia\)](#)

MSC:

[68Q85](#) Models and methods for concurrent and distributed computing (process algebras, bisimulation, transition nets, etc.) Cited in 11 Documents
[68Q55](#) Semantics in the theory of computing

Keywords:

[Object calculus](#); [Full abstraction](#); [Concurrency](#)

Full Text: [DOI](#)

References:

- [1] Abramsky, S., The lazy lambda calculus, () · [Zbl 0779.03003](#)
- [2] Abramsky, S., Domain theory in logical form, Ann. pure appl. logic, 51, 1-77, (1991) · [Zbl 0737.03006](#)
- [3] Abramsky, S.; Jagadeesan, R.; Malacaria, P., Full abstraction for PCF, Inform. comput., 163, 409-470, (2000) · [Zbl 1006.68028](#)
- [4] Boreale, M.; de Nicola, R.; Pugliese, R., Trace and testing equivalences on asynchronous processes, Inform. comput., 172, 139-164, (2002) · [Zbl 1009.68079](#)
- [5] Brookes, S.D.; Hoare, C.A.R.; Roscoe, A.W., A theory of communicating sequential processes, J. assoc. comput. Mach., 31, 3, 560-599, (1984) · [Zbl 0628.68025](#)
- [6] Cardelli, L.; Abadi, M., A theory of objects, (1996), Springer Berlin · [Zbl 0876.68014](#)
- [7] Di Blasio, P.; Fisher, K., A calculus for concurrent objects, (), 655-670
- [8] Ferreira, W.; Hennessy, M.; Jeffrey, A.S.A., A theory of weak bisimulation for core CML, J. funct. programming, 8, 5, 447-491, (1998) · [Zbl 0916.68093](#)
- [9] Fiore, M.; Moggi, E.; Sangiorgi, D., A fully-abstract model for the π -calculus, (), 43-54 · [Zbl 1053.68066](#)
- [10] A.D. Gordon, P.D. Hankin, A concurrent object calculus: reduction and typing, in: Proc. Internat. Workshop on High-Level Concurrent Languages, Electronic Notes in Theoretical Computer Science, Vol. 16(3), Elsevier, Amsterdam, 1998. · [Zbl 0917.68064](#)
- [11] Gordon, A.D.; Rees, G.D., Bisimilarity for a first-order calculus of objects with subtyping, (), 386-395
- [12] Hennessy, M., Algebraic theory of processes, (1988), MIT Press Cambridge, MA · [Zbl 0744.68047](#)
- [13] Hennessy, M., A fully abstract denotational semantics for the π -calculus, Theoret. comput. sci., 278, 1, 53-89, (2002) · [Zbl 1014.68105](#)
- [14] M. Hennessy, J. Rathke, Typed behavioural equivalences for processes in the presence of subtyping, in: Proc. Comput.: Australasian Theory Symp. Electronic Notes in Theoretical Computer Science, Vol. 61, Elsevier, Amsterdam, 2002. · [Zbl 1268.68127](#)
- [15] K. Honda, M. Tokoro, On asynchronous communication semantics, in: Proc. ECOOP Workshop on Object-Based Concurrent Computing, Lecture Notes in Computer Science, Vol. 612, Springer, Berlin, 1992, pp. 21-51.
- [16] Hyland, J.M.E.; Ong, C.-H.L., On full abstraction for PCFI, II, and III, Inform. comput., 163, 285-408, (2000) · [Zbl 1006.68027](#)
- [17] Jeffrey, A.S.A.; Rathke, J., Towards a theory of weak bisimulation for local names, (), 56-66
- [18] Jeffrey, A.S.A.; Rathke, J., A theory of bisimulation for a fragment of concurrent ML with local names, (), 311-321
- [19] Lynch, N.A.; Tuttle, M.R., Hierarchical correctness proofs for distributed algorithms, (), 137-151

- [20] Milner, R., Fully abstract semantics of typed λ -calculi, *Theoret. comput. sci.*, 4, 1-22, (1977) · [Zbl 0386.03006](#)
- [21] Milner, R., *Communicating and mobile systems*, (1999), Cambridge University Press Cambridge · [Zbl 0942.68002](#)
- [22] Milner, R.; Parrow, J.; Walker, D., A calculus of mobile processes, *Inform. comput.*, 100, 1, 1-77, (1992) · [Zbl 0752.68037](#)
- [23] R. Milner, D. Sangiorgi, Barbed bisimulation, in: *Proc. Internat. Colloq. Automata, Languages and Programming*, Lecture Notes in Computer Science, Vol. 623, Springer, Berlin, 1992, pp. 685-695.
- [24] J.-H. Morris, *Lambda calculus models of programming languages*, Dissertation, MIT, 1968.
- [25] A.M. Pitts, I.D.B. Stark, Observable properties of higher order functions that dynamically create local names, or: what's new? in: *Proc. Internat. Symp. Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, Vol. 711, Springer, Berlin, 1993, pp. 122-141.
- [26] Plotkin, G., LCF considered as a programming language, *Theoret. comput. sci.*, 5, 223-256, (1977) · [Zbl 0369.68006](#)
- [27] Rumbaugh, G.B.J.; Jacobson, I., *The unified modeling language user guide*, (1999), Addison-Wesley Reading, MA
- [28] D.S. Scott, Domains for denotational semantics, in: M. Nielsen, E.M. Schmidt (Eds.), *Proc. ICALP 82*, Lecture Notes in Computer Science, Vol. 140, Springer, Berlin, 1982, pp. 577-613. · [Zbl 0495.68025](#)
- [29] P. Selinger, First-order axioms for asynchrony, in: *Proc. CONCUR*, Lecture Notes in Computer Science, Vol. 1243, Springer, Berlin, 1997, pp. 376-390.

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.