

MR2968322 03B40 03D15 68Q42

Dal Lago, Ugo (I-BOLO-NDM); Martini, Simone (I-BOLO-IS)

On constructor rewrite systems and the lambda-calculus. (English summary)

Log. Methods Comput. Sci. **8** (2012), no. 3, 3:12, 27 pp.

This paper relates the computational complexity of first-order and higher-order computational models. On one hand, orthogonal constructor term rewriting systems are considered with their naïve cost model, i.e. each rule application has unitary cost. On the other hand, the weak lambda-calculus (no reduction is allowed under the scope of a lambda-abstraction) is considered with its naïve cost model, i.e. each beta-reduction step has unitary cost. The lambda-calculus is studied in both call-by-value and call-by-name parameter passing fashion.

It is well known that each orthogonal constructor term rewriting system can be simulated by lambda-terms and beta-reduction, while the converse simulation is a bit more complex. The main result is that the above cost models are linearly related. Therefore the two models are both reasonable in a complexity perspective, since the lambda-calculus cost-model is polynomially related to the actual cost of reducing a lambda-term on a Turing machine.

Luca Paolini

References

1. Stephen Bellantoni and Stephen Cook. A new recursion-theoretic characterization of the polytime functions. *Computational Complexity*, 2:97–110, 1992. [MR1190824](#)
2. H. Barendregt, M. Eekelen, J. Glauert, J. Kennaway, M. Plasmeijer, and M. Sleep. Term graph rewriting. In J. de Bakker, A. Nijman, and P. Treleaven, editors, *Volume II: Parallel Languages on PARLE: Parallel Architectures and Languages Europe*, pages 141–158. Springer-Verlag, 1986. [MR0910306](#)
3. Ugo Dal Lago and Simone Martini. The weak lambda-calculus as a reasonable machine. *Theoretical Computer Science*, 398:32–50, 2008. [MR2419677](#)
4. Ugo Dal Lago and Simone Martini. On constructor rewrite systems and the lambda-calculus. In *Automata, Languages and Programming, 36th International Colloquium, Proceedings*, volume 5556 of *LNCS*, pages 163–174. Springer, 2009. [MR2544793](#)
5. Ugo Dal Lago and Simone Martini. Derivational complexity is an invariant cost model. In *Foundational and Practical Aspects of Resource Analysis, First International Workshop, Proceedings*, volume 6324 of *LNCS*, pages 88–101. Springer, 2010.
6. Jean-Yves Girard. Light linear logic. *Information and Computation*, 143(2):175–204, 1998. [MR1626681](#)
7. Yuri Gurevich. The sequential ASM thesis. In *Current trends in theoretical computer science*, pages 363–392. World Scientific, 2001. [MR1886055](#)
8. Simon Peyton Jones. *The Implementation of Functional Programming Languages*. Prentice Hall, 1987.
9. Daniel Leivant. Ramified recurrence and computational complexity I: word recurrence and polytime. In *Feasible Mathematics II*, pages 320–343. Birkhäuser, 1995. [MR1322281](#)
10. Jean-Yves Marion and Jean-Yves Moyen. Efficient first order functional program interpreter with time bound certifications. In *Logic for Programming and Automated*

- Reasoning, 7th International Conference, Proceedings*, volume 1955 of *LNCS*, pages 25–42. Springer, 2000. [MR1850691](#)
11. Michel Parigot. On the representation of data in lambda-calculus. In *Computer Science Logic, 3rd International Workshop, Proceedings*, volume 440 of *LNCS*, pages 309–321. Springer, 1990. [MR1079086](#)
 12. Gordon D. Plotkin. Call-by-name, call-by-value and the lambda-calculus. *Theoretical Computer Science*, 1(2):125–159, 1975. [MR0429501](#)
 13. Detlef Plump. Graph-reducible term rewriting systems. In *Graph-Grammars and Their Application to Computer Science*, volume 532 of *LNCS*, pages 622–636. Springer, 1990.
 14. Michel Parigot and Paul Rozière. Constant time reductions in lambda-calculus. In *Mathematical Foundations of Computer Science 1993, 18th International Symposium, Proceedings*, volume 711 of *LNCS*, pages 608–617. Springer, 1993. [MR1265094](#)
 15. D. Sands, J. Gustavsson, and A. Moran. Lambda calculi and linear speedups. In *The Essence of Computation: Complexity, Analysis, Transformation. Essays Dedicated to Neil D. Jones*, number 2566 in *LNCS*, pages 60–82. Springer, 2002.
 16. Zdzislaw Splawski and Pawel Urzyczyn. Type fixpoints: Iteration vs. recursion. In *Functional Programming, 4th International Conference, Proceedings*, pages 102–113. ACM, 1999.
 17. Peter van Emde Boas. Machine models and simulation. In *Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity (A)*, pages 1–66. MIT Press, 1990. [MR1127167](#)
 18. Christopher Wadsworth. Some unusual λ -calculus numeral systems. In J.P. Seldin and J.R. Hindley, editors, *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*. Academic Press, 1980. [MR0592804](#)

Note: This list reflects references listed in the original paper as accurately as possible with no attempt to correct errors.