

Marriott, Kim; Odersky, Martin

A confluent calculus for concurrent constraint programming. (English) Zbl 0902.68034

Theor. Comput. Sci. 173, No. 1, 209-233 (1997).

Summary: Confluence is an important and desirable property as it allows the program to be understood by considering any desired scheduling rule, rather than having to consider all possible schedulings. Unfortunately, the usual operational semantics for concurrent constraint programs is not confluent as different process schedulings give rise to different sets of possible outcomes. We show that it is possible to give a natural confluent calculus for concurrent constraint programs, if the syntactic domain is extended by a blind choice operator and a special constant standing for a discarded branch. This has application to program analysis.

MSC:

68N99 Theory of software

Cited in 1 Document

Keywords:

confluence; concurrent constraint programs

Full Text: [DOI](#)

References:

- [1] Barendregt, H.P., The lambda calculus: its syntax and semantics, () · [Zbl 0467.03010](#)
- [2] Codish, M.; Falaschi, M.; Marriott, K., Suspension analyses for concurrent logic programs, ACM trans. programming languages systems, 16, 649-686, (1994)
- [3] Codish, M.; Falaschi, M.; Marriott, K.; Winsborough, W., Efficient analysis of concurrent constraint logic programs, (), 633-644 · [Zbl 1418.68052](#)
- [4] Codognet, C.; Codognet, P., Guarded constructive disjunction: angel or demon?, (), 345-361
- [5] Codognet, C.; Codognet, P.; Corsini, M., Abstract interpretation for concurrent logic languages, (), 215-232
- [6] Falaschi, M.; Gabbrielli, M.; Marriott, K.; Palamidessi, C., Compositional analysis for concurrent constraint programming, (), 210-221
- [7] Falaschi, M.; Gabbrielli, M.; Marriott, K.; Palamidessi, C., Confluence and concurrent constraint programming, (), 531-545
- [8] Henkin, L.; Monk, J.D.; Tarski, A., Cylindric algebras, () · [Zbl 0121.25402](#)
- [9] Huet, G., Confluent reductions: abstract properties and applications to term rewriting systems, J. ACM, 27, 797-821, (1980) · [Zbl 0458.68007](#)
- [10] lop, Jan Willem, Combinatory reduction systems, ()
- [11] Lloyd, J.W., Foundations of logic programming, (1987), Springer Berlin · [Zbl 0547.68005](#)
- [12] Maher, M., Logic semantics for a class of committed-choice programs, (), 858-876
- [13] Marriott, K.; Falaschi, M.; Gabbrielli, M.; Palamidessi, C., A simple semantics for logic programming languages with delay, (), 356-363
- [14] Montanari, U.; Rossi, F.; Saraswat, V., CC programs with both in- and non-determinism: A concurrent semantics, (), 151-161
- [15] Niehren, J., Funktionale berechnung in einem uniform nebenläufigen kalkül mit logischen variablen, () · [Zbl 0931.03050](#)
- [16] Niehren, J.; Smolka, G., A confluent relational calculus for higher-order programming with constraints, (), 89-104
- [17] Nyström, S.-O., λ -calculus and confluence — A fixpoint semantics for concurrent constraint programming, ()
- [18] Plotkin, G.D., Call-by-name, call-by-value, and the λ -calculus, Theoret. comput. sci., 1, 125-159, (1975) · [Zbl 0325.68006](#)
- [19] Saraswat, V.A.; Rinard, M., Concurrent constraint programming, (), 232-245
- [20] Saraswat, V.; Rinard, M.; Panangaden, P., The semantic foundations of concurrent constraint programming, (), 333-352
- [21] Zaffanella, E.; Levi, G.; Giacobazzi, R., Abstracting synchronization in concurrent constraint programming, () · [Zbl 0924.68045](#)

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.