Problem #50

Originator: Jean-Pierre Jouannaud
Date: December 1991

Summary: Investigate confluence and termination of combinations of typed lambda-calculi with term rewriting systems.

Combinations of typed λ-calculi with term-rewriting systems have been studied extensively in the past few years [Bar90][BTG89][DO90][Dou91]. The strongest termination result allows first-order rules as well as higher-order rules defined by a generalization of primitive recursion. Suppose all rules for functional constant F follow the schema:

$$F(\bar{l}[\bar{X}], \bar{Y}) \to v[F(\bar{r}_1[\bar{X}], \bar{Y}), ..., F(\bar{r}_m[\bar{X}], \bar{Y}), \bar{Y})]$$

where the (not necessarily disjoint) variables in $\bar{X}$ and $\bar{Y}$ are of arbitrary order, each of $\bar{l}, \bar{r}_1, ..., \bar{r}_m$ is in $\mathcal{T}(\mathcal{F}, \{\bar{X}\})$, $v[\bar{z}, \bar{Y}]$ is in $\mathcal{T}(\mathcal{F}, \{\bar{Y}, \bar{z}\})$, for new variables $\bar{z}$ of appropriate types, and $\bar{r}_1, \ldots, \bar{r}_m$ are each less than $\bar{l}$ in the multiset extension of the strict subterm ordering. If $\mathcal{T}(\mathcal{F}, \mathcal{X})$ is the term-algebra which includes only *previously* defined functional constants—forbidding the use of mutually recursive functional constants—termination is ensured [JO91]. Does termination also hold when there are mutually recursive definitions? Does this also hold when the subterm assumption is unfulfilled? (In [JO91] an alternative schema is proposed, with the subterm assumption weakened at the price of having only first-order variables in $\bar{X}$.) Questions of confluence of combinations of typed λ-calculi and higher-order systems also merit investigation. These results have been extended to combinations with more expressive type systems [BF93b][BF93a].

Remark

An extension to the Calculus of Constructions has been reported in [BFG94]. One can also allow the use of lexicographic and other "statuses" for the higher-order constants when comparing the subterms of F in left and right hand sides [Jouannaud and Okada, unpublished]. Finally, this can also be done when the rewrite rules follow from the induction schema in the initial algebra of the constructors [Wer94].

Important improvements of the previous works have been achieved in [Bla03] and [WC03].

Bibliography

[Bar90]   Franco Barbanera. Combining term rewriting and type assignment systems. *IJFCS*, 1:165–184, 1990.

[BF93a]   Franco Barbanera and Maribel Fernández. Combining first and higher order rewrite systems with type assignment systems. In *International Conference of Typed Lambda Calculi and Applications*, 1993.

[BF93b]   Franco Barbanera and Maribel Fernández. Modularity of termination and confluence in combinations of rewrite systems with λ_ω. In Andrzej Lingas, Rolf Karlsson, and Svante Carlsson, editors, *20th International Colloquium on Automata, Languages and Programming*, volume 700 of *Lecture Notes in Computer Science*, Lund, Sweden, July 1993. Springer-Verlag.

[BFG94]   Franco Barbanera, Maribel Fernández, and Herman Geuvers. Modularity of strong normalization and confluence in the λ-algebraic-cube. In Samson Abramsky, editor, *Ninth Symposium on Logic in Computer Science*, Paris, France, July 1994. IEEE.

[Bla03]   Frédéric Blanqui. Definitions by rewriting in the Calculus of Constructions, 2003. To appear in Mathematical Structures in Computer Science.

[BTG89]   Val Breazu-Tannen and Jean Gallier. Polymorphic rewriting conserves algebraic strong normalization. In *16th International Colloquium on Automata, Languages and Programming*, volume 372 of *Lecture Notes in Computer Science*, pages 137–150, Stresa, Italy, July 1989. Springer-Verlag.

[DO90]   Nachum Dershowitz and Mitsuhiro Okada. A rationale for conditional equational programming. *Theoretical Computer Science*, 75:111–138, 1990.

[Dou91]   Daniel Dougherty. Adding algebraic rewriting to the untyped lambda calculus (extended abstract). In Ronald. V. Book, ed-

itor, *4th International Conference on Rewriting Techniques and Applications*, volume 488 of *Lecture Notes in Computer Science*, pages 37–48, Como, Italy, April 1991. Springer-Verlag.

[JO91]   Jean-Pierre Jouannaud and Mitsuhiro Okada. Executable higher-order algebraic specification languages. In Gilles Kahn, editor, *Sixth Symposium on Logic in Computer Science*, pages 350–361, Amsterdam, The Netherlands, July 1991. IEEE.

[WC03]   Daria Walukiewicz-Chrzaszcz. Termination of rewriting in the Calculus of Constructions. *Journal of Functional Programming*, 13(2):339–414, 2003.

[Wer94]   Benjamin Werner. *Méta-théorie du Calcul des Constructions Inductives*. Thèse Univ. Paris VII, France, 1994.

January 22, 2014