Problem #22 (Solved !)

Originator: Nachum Dershowitz
Date: April 1991

Summary: Devise practical methods for proving termination of conditional rewriting systems.

Devise practical methods for proving termination of (standard) conditional rewriting systems. Part of the difficulty stems from the interdependence of normalization and termination.

Remark

Termination and decreasingness of CTRSs can be proved by transforming CTRSs into unconditional TRSs such that termination of the TRS is sufficient for decreasingness of the CTRS. Several variants of this transformation are studied in [BK86, DP86, GA01, GM87, Siv89, Mar96, Ohl01]. Termination of the TRSs resulting from this transformation can often be proved automatically using dependency pairs [AG00, GA01]. The transformation (together with the dependency pair approach) is implemented in the tools TALP [OCM00] and AProVE [GTSKF04]. Both tools use this transformation in order to show termination of logic programs, but AProVE can also prove termination and decreasingness of CTRSs in this way. A different approach for termination proofs of CTRSs with the general path order [DH95] is described in [Hoo96].

Bibliography

[AG00]   Thomas Arts and Jürgen Giesl. Termination of term rewriting using dependency pairs. *Theoretical Computer Science*, 236:133–178, 2000.

[BK86]   J. A. Bergstra and Jan Willem Klop. Conditional rewrite rules: Confluency and termination. *Journal of Computer and System Sciences*, 32(3):323–362, 1986.

[DH95]   Nachum Dershowitz and Charles Hoot. Natural termination. *Theoretical Computer Science*, 142:170–207, 1995.

[DP86]   Nachum Dershowitz and David Plaisted. Equational programming. *Machine Intelligence*, 11:21–56, 1986.

[GA01]   Jürgen Giesl and Thomas Arts. Verification of Erlang processes by dependency pairs. *Applicable Algebra in Engineering, Communication and Computing*, 12(1,2):39–72, 2001.

[GM87]   E. Giovanetti and C. Moiso. Notes on the elimination of conditions. In *Proc. CTRS '87*, LNCS 308, pages 91–97, 1987.

[GTSKF04] Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, and S. Falke. Automated termination proofs with AProVE. In Vincent van Oostrom, editor, *15th International Conference on Rewriting Techniques*, volume 3091 of *Lecture Notes in Computer Science*, pages 210–220, Aachen, Germany, June 2004. Springer-Verlag.

[Hoo96]   Charles Hoot. *Termination of Non-Simple Rewrite Systems*. PhD thesis, University of Illinois, 1996.

[Mar96]   M. Marchiori. Unravelings and ultra-properties. In *Proc. ALP '96*, LNCS 1139, pages 107–121, 1996.

[OCM00]   Enno Ohlebusch, Claus Claves, and Claude Marché. TALP: A tool for the termination analysis of logic programs. In Leo

Bachmair, editor, *Rewriting Techniques and Applications*, volume 1833 of *Lecture Notes in Computer Science*, pages 270–273, Norwich, UK, July 2000. Springer-Verlag.

[Ohl01]   Enno Ohlebusch. Termination of logic programs: Transformational methods revisited. *Applicable Algebra in Engineering, Communication and Computing*, 12(1,2):73–116, 2001.

[Siv89]   G. Sivakumar. *Proofs and Computations in Conditional Equational Theories*. PhD thesis, Department of Computer Science, University of Illinois, Urbana, IL, May 1989. Report R-89-1642.

January 22, 2014