CoCo 2016 Participant: FORT 1.0*

Franziska Rapp and Aart Middeldorp

Department of Computer Science, University of Innsbruck, Austria
{franziska.rapp|aart.middeldorp}@uibk.ac.at

FORT is a decision and synthesis tool for the first-order theory of rewriting for finite left-linear right-ground rewrite systems. It implements the decision procedure for this theory, which uses tree automata techniques and goes back to Dauchet and Tison [1]. In this theory confluence-related properties on ground terms are easily expressible. The basic functionality of FORT is described in [2] and in [3] we report on an extension to deal with non-ground terms.

FORT 1.0 is a completely new implementation in Java, for which the JAR file can be downloaded from

http://cl-informatik.uibk.ac.at/software/FORT/

The tool participates in the demo categories GCR and UN at CoCo 2016. The former is about ground-confluence of *many-sorted* rewrite systems. Since the set of well-typed terms according to a many-sorted type discipline is accepted by a tree automaton, the modifications required in FORT were straightforward.

The most significant change in FORT 1.0 is the support for parallelism, using the multi-threading capabilities of Java. This greatly speeds up the synthesis of rewrite systems satisfying certain properties expressible in the first-order theory of rewriting. Furthermore, we exploit this functionality for the UN demo category. In this category tools report the strongest property among CR, NFP, UNC and UN that can be established, or the answer NO if UN can be disproved. For the given rewrite system FORT checks the four properties in parallel, reusing basic automata constructions that can be shared among the properties. Additionally, the first thread different from CR that finishes with a positive answer checks for WN. In case WN holds, all properties are equivalent and CR is reported. As soon as it has the required information, it reports the optimal result. In case this information is not present shortly before the time limit, it kills all remaining threads and reports the strongest result that was established. This strategy can be illustrated quite well on COPS #215.[1] Within 260 milliseconds FORT has established NFP, while the thread checking for confluence is still running. Hence, we do not yet know the exact answer. Shortly before the 60 seconds time limit FORT reports NFP. However, this is not the optimal answer, since this system is actually confluent. As can be seen on this and many other examples, confluence is often harder to verify than the other three properties.

References

[1] M. Dauchet and S. Tison. The theory of ground rewrite systems is decidable. In *Proc. 5th IEEE Symposium on Logic in Computer Science*, pages 242–248, 1990. doi: 10.1109/LICS.1990.113750.

[2] F. Rapp and A. Middeldorp. Automating the first-order theory of left-linear right-ground term rewrite systems. In *Proc. 1st International Conference on Formal Structures for Computation and Deduction*, volume 52 of *Leibniz International Proceedings in Informatics*, pages 36:1–36:12, 2016. doi: 10.4230/LIPIcs.FSCD.2016.36.

[3] F. Rapp and A. Middeldorp. Confluence properties on open terms in the first-order theory of rewriting. In *Proc. 5th International Workshop on Confluence*, 2016. This volume.

*Supported by FWF (Austrian Science Fund) project P27528.

[1] http://cops.uibk.ac.at/?q=215