CoCo 2019 Participant: CSI^ho 0.3.2

Julian Nagele

School of Electronic Engineering and Computer Science, Queen Mary University of London, UK
j.nagele@qmul.ac.uk

CSI^ho is a tool for automatically (dis)proving confluence of higher-order rewrite systems, specifically pattern rewrite systems (PRSs) as introduced by Nipkow [3,7]. CSI^ho focuses on patterns in order to ensure decidability of unification for computing critical pairs. To this end CSI^ho implements a version of Nipkow's algorithm for higher-order pattern unification [8]. CSI^ho is an extension of CSI, a confluence prover for first-order rewrite systems. The tool is available at

http://cl-informatik.uibk.ac.at/software/csi/ho

Below we briefly list the criteria implemented by CSI^ho—a more detailed description of both CSI^ho and CSI can be found in [5,6].

For terminating PRSs CSI^ho decides confluence by checking joinability of critical pairs [7]. As termination criteria CSI^ho implements a basic higher-order recursive path ordering and static dependency pairs with dependency graph decomposition and the subterm criterion. Alternatively, one can also use an external termination tool like WANDA [2] as an oracle. For potentially non-terminating systems CSI^ho supports weak orthogonality [10] and van Oostrom's result on development closed critical pairs [9]. As a divide-and-conquer technique CSI^ho implements modularity for left-linear PRSs—note that confluence of PRSs is not modular in general [1]. Moreover CSI^ho uses the simple technique of adding and removing redundant rules [4], adapted for PRSs.

No new features were added to CSI^ho since CoCo 2018. It ran unopposed in the HRS category of CoCo 2019.

References

[1] C. Appel, V. van Oostrom, and J. G. Simonsen. Higher-order (non-)modularity. In *Proc. 21st RTA*, volume 6 of *LIPIcs*, pages 17–32, 2010.

[2] Cynthia Kop. *Higher Order Termination*. PhD thesis, Vrije Universiteit, Amsterdam, 2012.

[3] R. Mayr and T. Nipkow. Higher-order rewrite systems and their confluence. *TCS*, 192(1):3–29, 1998.

[4] J. Nagele, B. Felgenhauer, and A. Middeldorp. Improving automatic confluence analysis of rewrite systems by redundant rules. In *Proc. 26th RTA*, volume 36 of *LIPIcs*, pages 257–268, 2015.

[5] Julian Nagele. *Mechanizing Confluence: Automated and Certified Analysis of First- and Higher-Order Rewrite Systems*. PhD thesis, University of Innsbruck, 2017.

[6] Julian Nagele, Bertram Felgenhauer, and Aart Middeldorp. CSI: New evidence — A progress report. In *Proc. 26th CADE*, volume 10395 of *LNCS (LNAI)*, pages 385–397, 2017.

[7] T. Nipkow. Higher-order critical pairs. In *Proc. 6th LICS*, pages 342–349, 1991.

[8] Tobias Nipkow. Functional unification of higher-order patterns. In *Proc. 8th LICS*, pages 64–74, 1993.

[9] V. van Oostrom. Developing developments. *TCS*, 175(1):159–181, 1997.

[10] V. van Oostrom and F. van Raamsdonk. Weak orthogonality implies confluence: The higher order case. In *Proc. 3rd LFCS*, volume 813 of *LNCS*, pages 379–392, 1994.