# CoCo 2016 Participant: CSIˆho 0.2*

Julian Nagele

Department of Computer Science, University of Innsbruck, Austria
julian.nagele@uibk.ac.at

Higher-order rewriting combines standard, first-order rewriting with notions and concepts from the $\lambda$-calculus, resulting in rewriting systems with higher-order functions and bound variables. CSIˆho is a tool for automatically proving confluence of such higher-order systems, specifically pattern rewrite systems (PRSs) as introduced by Nipkow [3, 5]. The restriction to pattern left-hand sides is essential for obtaining decidability of unification and thus makes it possible to compute critical pairs. To this end CSIˆho implements a version of Nipkow's algorithm for higher-order pattern unification [6].

CSIˆho is built on top of CSI [9], a powerful confluence prover for first-order term rewrite systems. It is available from http://cl-informatik.uibk.ac.at/software/csi/ho/. Using CSI as foundation, CSIˆho inherits many of its attractions, in particular a strategy language, which allows for flexible configuration of the proof search. CSIˆho supports the following techniques:

2015 Knuth and Bendix' criterion, that is, for terminating PRSs we decide confluence by checking joinability of critical pairs [5]. For showing termination CSIˆho uses a basic higher-order recursive path ordering and static dependency pairs with dependency graph decomposition and the subterm criterion. For potentially non-terminating PRSs it supports weak orthogonality [8] and van Oostrom's result on development closed critical pairs [7].

2016 As a first divide-and-conquer criterion CSIˆho includes modularity of confluence for left-linear PRSs—note that confluence of PRSs is not modular in general [1]. To improve CSIˆho on terminating systems, external termination tools like WANDA [2] can now be used as a termination back-end. The final novelty this year is the simple technique of adding and removing redundant rules [4], adapted for PRSs.

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

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

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

[8] 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.

[9] H. Zankl, B. Felgenhauer, and A. Middeldorp. CSI – A confluence tool. In *Proc. 23rd CADE*, volume 6803 of *LNCS (LNAI)*, pages 499–505, 2011.

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