A Dependency Pair Framework for Relative Termination of Term Rewriting*

Jan-Christoph Kassing
RWTH Aachen University, Aachen, Germany

Grigory Vartanyan
RWTH Aachen University, Aachen, Germany

Jürgen Giesl
RWTH Aachen University, Aachen, Germany

Dependency pairs (DPs) [1,4,6] are one of the most powerful techniques for proving termination of term rewrite systems (TRSs), and they are used in almost all tools for termination analysis of TRSs.

A combination of two TRSs (a *main* TRS $\mathcal{R}$ and a *base* TRS $\mathcal{B}$) is *relatively terminating* if there is no rewrite sequence that uses infinitely many steps with rules from $\mathcal{R}$ (whereas rules from $\mathcal{B}$ may be used infinitely often). Relative termination of TRSs has been studied since decades [3], and approaches based on relative rewriting are used for many applications.

However, while techniques and tools for analyzing ordinary termination of TRSs are very powerful due to the use of DPs, a direct application of standard DPs to analyze relative termination is not possible. Therefore, most existing approaches for automated analysis of relative termination are quite restricted in power. Hence, one of the largest open problems regarding DPs is Problem #106 of the RTA List of Open Problems [2]: *Can we use the dependency pair method to prove relative termination?* A first major step towards an answer to this question was presented in [7] by giving criteria for $\mathcal{R}$ and $\mathcal{B}$ that allow the use of ordinary DPs for relative termination.

Recently, we adapted DPs to prove almost-sure termination of probabilistic TRSs, by using *annotated dependency pairs (ADPs)* [8,9]. In this adaption, one considers all *defined* function symbols in the right-hand side of a rule at once, whereas ordinary DPs consider them separately.

We show that considering the defined symbols on right-hand sides separately (as for classical DPs) does not suffice for relative termination. On the other hand, we do not need to consider all of them at once either (i.e., we do not have to use the notion of ADPs from [8,9]). Instead, we introduce a new definition of ADPs that is suitable for relative termination and develop a corresponding ADP framework for automated relative termination proofs of TRSs. So while [7] presented conditions under which the *ordinary classical* DP framework can be used to prove relative termination, we develop the first *specific* DP framework for relative termination. We implemented our new ADP framework in our tool AProVE [5] and evaluate it in comparison to state-of-the-art tools for relative termination of TRSs.

Related Version Extended abstract of our IJCAR '24 paper [10]

References

1 Thomas Arts and Jürgen Giesl. Termination of term rewriting using dependency pairs. *Theoretical Computer Science*, 236(1-2):133–178, 2000. doi:10.1016/S0304-3975(99)00207-8.

2 Nachum Dershowitz. The RTA list of open problems. URL: https://www.cs.tau.ac.il/~nachum/rtaloop/.

* funded by the DFG Research Training Group 2236 UnRAVeL

3 Alfons Geser. *Relative Termination*. PhD thesis, University of Passau, Germany, 1990. URL: https://www.uni-ulm.de/fileadmin/website_uni_ulm/iui/Ulmer_Informatik_Berichte/1991/UIB-1991-03.pdf.

4 Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, and Stephan Falke. Mechanizing and improving dependency pairs. *Journal of Automated Reasoning*, 37(3):155–203, 2006. doi:10.1007/s10817-006-9057-7.

5 Jürgen Giesl, Cornelius Aschermann, Marc Brockschmidt, Fabian Emmes, Florian Frohn, Carsten Fuhs, Jera Hensel, Carsten Otto, Martin Plücker, Peter Schneider-Kamp, Thomas Ströder, Stephanie Swiderski, and René Thiemann. Analyzing program termination and complexity automatically with AProVE. *Journal of Automated Reasoning*, 58(1):3–31, 2017. doi:10.1007/s10817-016-9388-y.

6 Nao Hirokawa and Aart Middeldorp. Automating the dependency pair method. *Information and Computation*, 199(1-2):172–199, 2005. doi:10.1016/j.ic.2004.10.004.

7 José Iborra, Naoki Nishida, Germán Vidal, and Akihisa Yamada. Relative termination via dependency pairs. *Journal of Automated Reasoning*, 58(3):391–411, 2017. doi:10.1007/S10817-016-9373-5.

8 Jan-Christoph Kassing, Stefan Dollase, and Jürgen Giesl. A complete dependency pair framework for almost-sure innermost termination of probabilistic term rewriting. In *Proc. FLOPS '24*, LNCS 14659, pages 62–80, 2024. doi:10.1007/978-981-97-2300-3_4.

9 Jan-Christoph Kassing and Jürgen Giesl. Annotated dependency pairs for full almost-sure termination of probabilistic term rewriting. In *Principles of Verification: Cycling the Probabilistic Landscape*, LNCS 15260, pages 339–366, 2024. doi:10.1007/978-3-031-75783-9_14.

10 Jan-Christoph Kassing, Grigory Vartanyan, and Jürgen Giesl. A dependency pair framework for relative termination of term rewriting. In *Proc. IJCAR '24*, LNCS 14740, pages 360–380, 2024. doi:10.1007/978-3-031-63501-4_19.