

CoCo 2026 Participant: `nonreach` + CeTA-Prover

Florian Meßner¹ and René Thiemann²

¹ Independent Researcher, Austria

² University of Innsbruck, Austria

The tool `nonreach` [1, 2] is an automated, efficient tool to check infeasibility with respect to oriented conditional term rewrite systems (CTRSs). The Haskell source code can be obtained from a public *git* repository hosted on *codeberg*, and during the installation process it will automatically download CeTA-Prover [3].

<https://codeberg.org/fmessner/nonreach>

Given a CTRS (or a TRS) and one or more infeasibility problems, `nonreach` uses a combination of *decomposition*, based on narrowing (with some heuristics) and proving root-nonreachability [4], and *fast checks*, based on *tcap* and the *inductive symbol transition graph* [4].

These methods are applied alternately until `nonreach` either obtains infeasibility (by simplifying the tree to False), finds a satisfying substitution, or reaches a user-defined threshold of iterations (and concludes MAYBE).

Whereas the proofs of infeasibility of the previous version of `nonreach` are quite verbose and can also be machine checked, this was not the case for feasibility. In particular, providing just the satisfying substitution σ such that $s\sigma \rightarrow^* t\sigma$ for some reachability condition $s \approx t$ is not enough information for the CPF certificates that are expected by CeTA [5].

To this end the new version of `nonreach`, version 1.4, now invokes CeTA-Prover internally to reconstruct all the details of the rewrite sequence $s\sigma \rightarrow^* t\sigma$. These details are then displayed in human-readable form (by `nonreach`) or they are used to generate machine checkable certificates (by CeTA-Prover).

In total, our new combination of `nonreach` + CeTA-Prover is the first automatic tool that generates *certifiable feasibility proofs*. Moreover, there are minor extensions on the `nonreach` side, e.g., by adding native support for the ARI format and for CPF 3 certificates, i.e., without any external translation tools from COPS or from CPF 2.

References

- [1] Florian Meßner and Christian Sternagel. `nonreach` – A Tool for Nonreachability Analysis. *Proc. of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, volume 11427 of *LNCS*, pages 337–343. Springer, 2019. URL: [doi:10.1007/978-3-030-17462-0_19](https://doi.org/10.1007/978-3-030-17462-0_19).
- [2] Florian Meßner. Automated Conditional Nonreachability Master Thesis, University of Innsbruck, 2020. <https://diglib.uibk.ac.at/ulbtirolhs/content/titleinfo/5489915>.
- [3] René Thiemann. CoCo 2026 Participant: CeTA-Prover. *Proc. of the 15th International Workshop on Confluence, July 24, 2026*. This volume.
- [4] Christian Sternagel and Akihisa Yamada. Reachability analysis for termination and confluence of rewriting. *Proc. of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, volume 11427 of *LNCS*, pages 262–278. Springer, 2019. [doi:10.1007/978-3-030-17462-0_15](https://doi.org/10.1007/978-3-030-17462-0_15).
- [5] René Thiemann and Christian Sternagel. Certification of termination proofs using CeTA. *Proc. of the International Conference on Theorem Proving in Higher Order Logics*, volume 5674 of *LNCS*, pages 452–468. Springer, 2009. [doi:10.1007/978-3-642-03359-9_31](https://doi.org/10.1007/978-3-642-03359-9_31).