

Report on the outcomes of a Short-Term Scientific Mission¹

Action number: CA20111

Grantee name: Dragana Milovancevic

Details of the STSM

Title:

Start and end date: 10/03/2023 to 27/03/2023

Description of the work carried out during the STSM

Description of the activities carried out during the STSM. Any deviations from the initial working plan shall also be described in this section.

The plan of this STSM was to explore termination checking techniques for functional programs, in order to allow sound and automated equivalence proofs of real-world examples. The activities carried out correspond to the three objectives of the STSM:

- We considered variants of TRSs as an intermediate representation for Scala. We revisited several existing frontends of AProVE, notably the ones for Haskell [1], Java [2] and C [3]. We decided to prioritize ITRSs (RTA'09 input format [4]) and integration with AProVE over Ctrl. We also decided to use the syntax with explicit conditions ([: syntax) and not continuations, for compatibility with other formats. Continuations are however mandatory for conditions with user-defined functions and we might use them systematically in the first version of the automated translation.
- We manually translated a sample of challenging real-world Scala programs to TRSs. We evaluated termination checking of AProVE (for various ITRSs) and Ctrl (for ITRSs and LCTRSs) on our translations. We observed that the results of both AProVE and Ctrl improve over the results of Stainless, and that AProVE outperforms Ctrl on those benchmarks.
- We identified the challenges and the next steps towards the automated translation between Scala and ITRSs. Some of the challenges we discussed are non-termination of auxiliary functions for irrelevant inputs, exceptions, mathematical vs. machine integers, disproving termination, and higher order arguments. We identified a simple first-order language with recursive functions as a minimal starting point for our automated translation, based on definitions from [5]. We came up with a preliminary plan on how to integrate the two tools (Stainless and AProVE).

¹ This report is submitted by the grantee to the Action MC for approval and for claiming payment of the awarded grant. The Grant Awarding Coordinator coordinates the evaluation of this report on behalf of the Action MC and instructs the GH for payment of the Grant.

References

1. Jürgen Giesl, Matthias Raffelsieper, Peter Schneider-Kamp, Stephan Swiderski, and René Thiemann. 2011. Automated Termination Proofs for Haskell by Term Rewriting. ACM Trans. Program. Lang. Syst.. 33. 7. 10.1145/1890028.1890030.
2. Carsten Otto, Marc Brockschmidt, Christian Essen, and Jürgen Giesl. 2010. Automated Termination Analysis of Java Bytecode by Term Rewriting. RTA 2010. 6. 259-276. 10.4230/LIPIcs.RTA.2010.259.
3. Cynthia Kop and Naoki Nishida. 2015. Converting C to LCTRSs. <https://www.trs.cm.is.nagoya-u.ac.jp/c2lctrs/formal.pdf>
4. Carsten Fuhs, Jürgen Giesl, Martin Plücker, Peter Schneider-Kamp, and Stephan Falke. RTA 2009. Proving Termination of Integer Term Rewriting. 32-47. 10.1007/978-3-642-02348-4_3.
5. Nicolas Charles Yves Voirol. 2019. Verified Functional Programming. <https://doi.org/10.5075/epfl-thesis-9479>

Description of the STSM main achievements and planned follow-up activities

Description and assessment of whether the STSM achieved its planned goals and expected outcomes, including specific contribution to Action objective and deliverables, or publications resulting from the STSM. Agreed plans for future follow-up collaborations shall also be described in this section.

The mission achieved its planned goals:

- We now have a translation sample.
- We were able to confirm our preliminary decisions and make further decisions based on results on our sample.
- We came up with a plan on how to continue with the automated translation and tool integration.

We identified the following next goals:

- Automated translation from Scala to ITRSs (.scala source -> stainless AST -> ITRS AST -> .itrs source)
- Integration of AProVE in Stainless, as a library (Stainless: translates .scala and sends .itrs to AProVE; AProVE: proves termination and sends YES/NO to Stainless)
- A formal proof (in the spirit of [1,2]) that termination proofs for translated ITRS programs imply termination of the original Scala programs.

We obtained EPFL's Doc.Mobility grant for a follow-up six-month collaboration project.

The efforts of this mission and particularly the planned follow-up activities regarding the integration of the two tools are very much in the spirit of WG3's objective on connecting tools.