A First Decision Procedure for Almost-Sure Termination of Probabilistic Term Rewriting

Jan-Christoph Kassing
RWTH Aachen University, Germany

Arion Scheid
TU Wien, Vienna, Austria

Henri Nagel
RWTH Aachen University, Germany

Jürgen Giesl
RWTH Aachen University, Germany

Abstract

While termination of ordinary programs has been studied for decades, the analysis of *probabilistic programs* has become increasingly important. In the probabilistic setting, requiring all executions to be finite is often too restrictive. Instead, one studies *almost-sure termination* (AST) where every computation has to terminate with probability 1. Thus, infinite executions may still exist, but the probability of such an infinite execution is 0.

In this talk, we consider term rewriting, a well-studied functional programming model based on pattern matching [2]. More precisely, we consider probabilistic term rewrite systems (PTRSs) [1, 4], where the choice of the applied rule remains nondeterministic, but the result after applying a rule is determined probabilistically, similar to the semantics of a Markov decision process.

In addition to developing automatic techniques for analyzing AST, one should investigate their limitations. There are multiple ways to assess the "difficulty" of a decision problem in computer science. For example, one may ask under which assumptions the problem becomes decidable. Such assumptions can be syntactic restrictions, e.g., considering only a specific class of PTRSs. Another way to assess the difficulty of an undecidable problem is to relate it to other undecidable problems. So one asks which undecidable problems would need to be decidable in order to decide AST for arbitrary PTRSs. This induces a hierarchy of undecidable problems, where problem A is "harder" than problem B if A remains undecidable even when equipped with an oracle for B. For example, universal termination remains undecidable even if the halting problem (termination on a given input) were decidable, and is therefore "harder".

Since similar decision problems for imperative probabilistic programs have already been placed in such hierarchies [3], we follow the first approach and restrict the structure of PTRSs so that AST becomes decidable. In the non-probabilistic setting, there is a well-known subclass of TRSs where termination is decidable: *right-ground TRSs* [5]. A TRS is *right-ground* if all of its right-hand sides are ground terms, i.e., contain no variables. For example, the rule $\mathsf{leq}(x,x) \to \mathsf{true}$ is right-ground, but $\mathsf{leq}(\mathsf{s}(x),\mathsf{s}(y)) \to \mathsf{leq}(x,y)$ is not. If right-hand sides contain no variables, then one cannot pass arbitrary information through recursive calls.

We show why the decision procedure for termination of right-ground TRSs does not generalize to the probabilistic setting. Consequently, we further restrict the subclass so that it can be related to a stochastic system with a known decision procedure for AST: *stochastic context-free grammars*. Stochastic context-free grammars (SCFGs) [6] generalize context-free grammars by replacing nondeterministic choice with probabilistic choice.

In this talk, we recapitulate SCFGs from [6] and how to decide AST for them. Then, we show how to automatically transform every PTRS $\mathcal{P}$ from a certain subclass into an SCFG $\mathcal{G}$ such that $\mathcal{P}$ is AST if and only if $\mathcal{G}$ is AST. Here, we require the PTRS to be: (RG) right-ground (as in the non-probabilistic decision procedure), (NO) non-overlapping (to remove additional nondeterminism in the rule selection), and (TR) tail-recursive (there are no nested defined symbols, as in context-free grammars). This yields the first decision procedure for AST of PTRSs that are RG, NO, and TR.

2012 ACM Subject Classification Theory of computation → Logic and verification

Keywords and phrases Termination, Probabilistic Programs, Term Rewriting, Decision Procedures

References

1. Martin Avanzini, Ugo Dal Lago, and Akihisa Yamada. On Probabilistic Term Rewriting. *Sci. Comput. Program.*, 185, 2020. doi:10.1016/j.scico.2019.102338.
2. Franz Baader and Tobias Nipkow. *Term Rewriting and All That.* 1998. doi:10.1017/CBO9781139172752.
3. Benjamin Lucien Kaminski, Joost-Pieter Katoen, and Christoph Matheja. On the Hardness of Analyzing Probabilistic Programs. *Acta Informatica*, 56:255–285, 2019. doi:10.1007/s00236-018-0321-1.
4. Olivier Bournez and Florent Garnier. Proving Positive Almost-Sure Termination. In *Proc. RTA '05*, LNCS 3467, pages 323–337. Springer, 2005. doi:10.1007/978-3-540-32033-3_24.
5. Nachum Dershowitz. Termination of Linear Rewriting Systems. In *Proc. ICALP '81*, LNCS 115, pages 448–458, 1981. doi:10.1007/3-540-10843-2_36.
6. Kousha Etessami and Mihalis Yannakakis. Recursive Markov Chains, Stochastic Grammars, and Monotone Systems of Nonlinear Equations. *J. ACM*, 56:1:1–1:66, 2009. doi:10.1145/1462153.1462154.