Reasoning About Probabilistic Loops, Moment by Moment

Ezio Bartocci
TU Wien, Austria

Abstract

Probabilistic programs and stochastic models have become a central paradigm for describing systems operating under uncertainty, ranging from randomised algorithms and Bayesian inference to cyber-physical systems. Their formal analysis, however, remains highly challenging due to the interplay among probabilistic behaviour, nondeterminism, and potentially unbounded computations. In recent years, martingale-based reasoning and moment-based and recurrence-equation approaches have emerged as powerful techniques for the automated verification of probabilistic loops [4, 13].

We present a line of work [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] on the automated reasoning about probabilistic programs through moment-based analysis, recurrence solving, and asymptotic reasoning. A key observation is that, for the class of Prob-Solvable loops, it is always possible to characterise higher-order statistical moments via systems of linear recurrence equations admitting computable closed forms [4, 6]. This feature enables the systematic derivation of quantitative properties such as expected values, variances, and probabilistic termination guarantees.

We first introduce the class of probabilistic (potentially infinite) loops that we call *Prob-Solvable*. For every loop in this class we can compute, analytically and without sampling, the exact higher-order statistical moments as closed-form expressions of the number of iterations. Moreover, we develop a faithful encoding of several families of Bayesian networks (BNs) into Prob-Solvable loops. In particular, BNs can be represented as probabilistic loops with polynomial assignments over random variables, enabling automated reasoning about exact inference, filtering, sensitivity analysis, and sampling-based procedures through invariant generation and closed-form recurrence solving [5, 15]. The proposed framework supports discrete, Gaussian, conditional linear Gaussian, and dynamic BNs, extending probabilistic program analysis to a broad family of probabilistic graphical models.

Beyond Prob-Solvable loops, we also characterise a hierarchy of solvable and unsolvable probabilistic loop classes and extend moment-based analysis to loops with non-polynomial assignments [1, 2, 7, 8]. These works widen the applicability of symbolic techniques beyond the original polynomial setting.

The resulting algorithms are implemented in tools such as Mora [6] and Polar [12]. While Mora focuses on the automatic generation of moment-based invariants only for probabilistic loops with polynomial assignments, Polar provides a more general algebraic framework for exact symbolic analysis of probabilistic loops and related stochastic models [12]. We further investigate the inverse problem of synthesising probabilistic loops from prescribed moment sequences, thereby complementing analysis with program construction techniques [14].

We then turn to verifying probabilistic termination properties. In this setting, martingale-based proof rules provide sufficient conditions for establishing almost-sure termination (AST), positive almost-sure termination (PAST), as well as non-termination properties. The key challenge lies in automating these proof obligations. To overcome this, we introduce Amber, a fully automated framework for proving and refuting the probabilistic termination of polynomial loops [9, 10, 11]. Amber blends martingale reasoning with asymptotic bounds obtained from recurrence equations and handles symbolic constants as well as standard probability distributions.

A common thread running through these papers is the reduction of probabilistic reasoning to symbolic algebraic reasoning. By expressing expected values and higher-order moments of stochastic updates as systems of recurrence equations, automated techniques can be applied for solving recurrences, generating invariants, and performing asymptotic analysis. This combination of probability theory, formal methods, and symbolic computation yields exact or asymptotically tight properties of stochastic systems and offers a viable pathway to automate quantitative verification tasks that would otherwise be intractable.

37th International Conference on Concurrency Theory (CONCUR 2026).
Editors: Ana Sokolova and Patrick Totzke; Article No. 2; pp. 2:1–2:3
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2012 ACM Subject Classification Theory of computation → Probabilistic computation; Theory of computation → Automated reasoning; Theory of computation → Program analysis; Theory of computation → Logic and verification; Theory of computation → Program reasoning; Mathematics of computing → Discrete mathematics

Keywords and phrases Probabilistic programs, probabilistic loops, moment-based analysis, martingales, recurrence equations, invariant generation, probabilistic termination, Bayesian networks, symbolic computation, formal verification

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2026.2

Category Invited Talk

Funding *Ezio Bartocci*: This work has been partially funded by the Vienna Science and Technology Fund (WWTF) and by the State of Lower Austria [Grant ID: 10.47379/ICT25017] – grant ProMT.

Acknowledgements This invited talk is based on several joint works with my co-authors Daneshvar Amrollahi, Efstathia Bura, George Kenison, Andrey Kofnov, Laura Kovács, Marcel Moosbrugger, Julian Müllner, Miroslav Stanković.

References

1. Daneshvar Amrollahi, Ezio Bartocci, George Kenison, Laura Kovács, Marcel Moosbrugger, and Miroslav Stankovic. Solving invariant generation for unsolvable loops. In *Proc. of SAS 2022: the 29th International Symposium on Static Analysis*, Lecture Notes in Computer Science, pages 19–43. Springer, 2022. doi:10.1007/978-3-031-22308-2_3.
2. Daneshvar Amrollahi, Ezio Bartocci, George Kenison, Laura Kovács, Marcel Moosbrugger, and Miroslav Stankovic. (Un)Solvable loop analysis. *Formal Methods in System Design*, 65(1):163–194, 2025. doi:10.1007/s10703-024-00455-0.
3. Ezio Bartocci. The ProbInG project: Advancing automatic analysis of probabilistic loops. In *Proc. of ISoLA 2024: the 12th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation. REoCAS Colloquium in Honor of Rocco De Nicola*, Lecture Notes in Computer Science, pages 152–167. Springer, 2024. doi:10.1007/978-3-031-73709-1_10.
4. Ezio Bartocci, Laura Kovács, and Miroslav Stankovic. Automatic generation of moment-based invariants for prob-solvable loops. In *Proc. ATVA 2019: 17th International Symposium on Automated Technology for Verification and Analysis*, Lecture Notes in Computer Science, pages 255–276. Springer, 2019. doi:10.1007/978-3-030-31784-3_15.
5. Ezio Bartocci, Laura Kovács, and Miroslav Stankovic. Analysis of Bayesian networks via Prob-Solvable loops. In *Proc. of ICTAC 2020: the 17th International Colloquium on Theoretical Aspects of Computing*, Lecture Notes in Computer Science, pages 221–241. Springer, 2020. doi:10.1007/978-3-030-64276-1_12.
6. Ezio Bartocci, Laura Kovács, and Miroslav Stankovic. Mora – automatic generation of moment-based invariants. In *Proc. of TACAS 2020: the 26th International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, Lecture Notes in Computer Science, pages 492–498. Springer, 2020. doi:10.1007/978-3-030-45190-5_28.
7. Andrey Kofnov, Marcel Moosbrugger, Miroslav Stankovic, Ezio Bartocci, and Efstathia Bura. Moment-based invariants for probabilistic loops with non-polynomial assignments. In *Proc. of QEST 2022: 19th International Conference on Quantitative Evaluation of Systems*, Lecture Notes in Computer Science, pages 3–25. Springer, 2022. doi:10.1007/978-3-031-16336-4_1.
8. Andrey Kofnov, Marcel Moosbrugger, Miroslav Stankovic, Ezio Bartocci, and Efstathia Bura. Exact and approximate moment derivation for probabilistic loops with non-polynomial assignments. *ACM Transactions on Modeling and Computer Simulation*, 34(3):18:1–18:25, 2024. doi:10.1145/3641545.

9 Marcel Moosbrugger, Ezio Bartocci, Joost-Pieter Katoen, and Laura Kovács. Automated termination analysis of polynomial probabilistic programs. In *Proc. of ESOP 2021: the 30th European Symposium on Programming Languages and Systems*, Lecture Notes in Computer Science, pages 491–518. Springer, 2021. `doi:10.1007/978-3-030-72019-3_18`.

10 Marcel Moosbrugger, Ezio Bartocci, Joost-Pieter Katoen, and Laura Kovács. The probabilistic termination tool Amber. In *Proc. of FM 2021: the 24th International Symposium on Formal Methods*, Lecture Notes in Computer Science, pages 667–675. Springer, 2021. `doi:10.1007/978-3-030-90870-6_36`.

11 Marcel Moosbrugger, Ezio Bartocci, Joost-Pieter Katoen, and Laura Kovács. The probabilistic termination tool Amber. *Formal Methods in System Design*, 61(1):90–109, 2023. `doi:10.1007/s10703-023-00424-z`.

12 Marcel Moosbrugger, Julian Müllner, Ezio Bartocci, and Laura Kovács. Polar: An algebraic analyzer for (probabilistic) loops. In *Principles of Verification: Cycling the Probabilistic Landscape*, Lecture Notes in Computer Science, pages 179–200. Springer, 2024. `doi:10.1007/978-3-031-75783-9_8`.

13 Marcel Moosbrugger, Miroslav Stankovic, Ezio Bartocci, and Laura Kovács. This is the moment for probabilistic loops. *Proceedings of the ACM on Programming Languages*, 6(OOPSLA2):1497–1525, 2022. `doi:10.1145/3563341`.

14 Miroslav Stankovic and Ezio Bartocci. Probabilistic loop synthesis from sequences of moments. In *Proc. of QEST+FORMATS 2024: First International Joint Conference on Quantitative Evaluation of Systems and Formal Modeling and Analysis of Timed Systems*, Lecture Notes in Computer Science, pages 233–248. Springer, 2024. `doi:10.1007/978-3-031-68416-6_14`.

15 Miroslav Stankovic, Ezio Bartocci, and Laura Kovács. Moment-based analysis of Bayesian network properties. *Theoretical Computer Science*, 903:113–133, 2022. `doi:10.1016/j.tcs.2021.12.021`.