Concurrency meets Probability: Theory and Practice

(Abstract)

Joost-Pieter Katoen

Software Modelling and Verification, RWTH Aachen University, Germany
Formal Methods and Tools, University of Twente, The Netherlands

Treating random phenomena in concurrency theory has a long tradition. Petri nets [18, 10] and process algebras [14] have been extended with probabilities. The same applies to behavioural semantics such as strong and weak (bi)simulation [1], and testing pre-orders [5]. Beautiful connections between probabilistic bisimulation [16] and Markov chain lumping [15] have been found. A plethora of probabilistic concurrency models has emerged [19]. Over the years, the focus shifted from covering discrete to treating continuous stochastic phenomena [12, 13].

We argue that both aspects can be elegantly combined with non-determinism, yielding the *Markov automata* model [8]. This model has nice theoretical characteristics. It is closed under parallel composition and hiding. Conservative extensions of (bi)simulation are congruences [8, 4]. It has a simple process algebraic counterpart [20]. On-the-fly partial-order reduction yields substantial state-space reductions [21]. Their quantitative analysis largely depends on (efficient) linear programming and scales well [11].

More importantly though: Markov automata serve an important *practical need*. They are the obvious choice for providing semantics to the Architecture Analysis & Design Language (AADL [9]), an industry standard for the automotive and aerospace domain. As experienced in several ESA projects, this holds in particular for the AADL annex dealing with error models [3]. They provide a compositional semantics to dynamic fault trees [6], a key model for reliability engineering [2]. Finally, they give a natural semantics to *every* generalised stochastic Petri net (GSPN [17]), a prominent model in performance analysis. This conservatively extends the existing GSPN semantics that is restricted to "well-defined" nets, i.e., nets without non-determinism [7]. Powerful software tools support this and incorporate efficient analysis and minimisation algorithms [11].

This substantiates our take-home message: *Markov automata bridge the gap between an elegant theory and practical engineering needs.*

Acknowledgement. This work is funded by the EU FP7-projects MoVeS, SENSATION and MEALS, the DFG-NWO bilateral project ROCKS, the NWO project SYRUP, the ESA project HASDEL, and the STW project ArRangeer.

References

1. C. Baier, J.-P. Katoen, H. Hermanns, and V. Wolf. Comparative branching-time semantics for Markov chains. *Inf. Comput.*, 200(2):149–214, 2005.
2. H. Boudali, P. Crouzen, and M. Stoelinga. A rigorous, compositional, and extensible framework for dynamic fault tree analysis. *IEEE Trans. Dependable Sec. Comput.*, 7(2):128–143, 2010.
3. M. Bozzano, A. Cimatti, J.-P. Katoen, V. Y. Nguyen, T. Noll, and M. Roveri. Safety, dependability and performance analysis of extended AADL models. *The Computer Journal*, 54(5):754–775, 2011.
4. Y. Deng and M. Hennessy. On the semantics of Markov automata. *Inf. Comput.*, 222:139–168, 2013.
5. Y. Deng, R. J. van Glabbeek, M. Hennessy, and C. Morgan. Characterising testing preorders for finite probabilistic processes. *Logical Methods in Computer Science*, 4(4), 2008.
6. J. Dugan and S. Bavuso. Dynamic fault-tree models for fault-tolerant computer systems. *IEEE Tr. on Reliability*, 41(3):363–377, 1992.
7. C. Eisentraut, H. Hermanns, J.-P. Katoen, and L. Zhang. A semantics for every GSPN. In *ICATPN*, volume 7927 of *LNCS*, pages 90–109. Springer, 2013.
8. C. Eisentraut, H. Hermanns, and L. Zhang. On probabilistic automata in continuous time. In *LICS*, pages 342–351. IEEE Computer Society, 2010.
9. P. H. Feiler and D. P. Gluch. *Model-Based Engineering with AADL - An Introduction to the SAE Architecture Analysis and Design Language*. SEI Series in Software Engineering. Addison-Wesley, 2012.
10. G. Florin and S. Natkin. Les reseaux de Petri stochastiques. *Technique et Science Informatiques*, 4(1):143–160, 1985.
11. D. Guck, H. Hatefi, H. Hermanns, J.-P. Katoen, and M. Timmer. Modelling, reduction and analysis of Markov automata. In *QEST*, LNCS. Springer, 2013.
12. H. Hermanns, U. Herzog, and J.-P. Katoen. Process algebra for performance evaluation. *Theor. Comput. Sci.*, 274(1-2):43–87, 2002.
13. J. Hillston. Process algebras for quantitative analysis. In *LICS*, pages 239–248. IEEE Computer Society, 2005.
14. B. Jonsson, W. Yi, and K. G. Larsen. Probabilistic extensions of process algebras. In J. Bergstra, A. Ponse, and S. Smolka, editors, *Handbook of Process Algebra*, chapter 11, pages 685–711. 2001.
15. J. Kemeny and J. Snell. *Finite Markov Chains*. D. Van Nostrand, 1960.
16. K. G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Inf. Comput.*, 94(1):1–28, 1991.
17. M. A. Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. John Wiley & Sons, 1995.
18. M. K. Molloy. Performance analysis using stochastic Petri nets. *IEEE Trans. Computers*, 31(9):913–917, 1982.
19. A. Sokolova and E. P. de Vink. Probabilistic automata: System types, parallel composition and comparison. In *Validation of Stochastic Systems*, volume 2925 of *LNCS*, pages 1–43. Springer, 2004.
20. M. Timmer, J.-P. Katoen, J. van de Pol, and M. Stoelinga. Efficient modelling and generation of Markov automata. In *CONCUR*, volume 7454 of *LNCS*, pages 364–379. Springer, 2012.
21. M. Timmer, M. Stoelinga, and J. van de Pol. Confluence reduction for Markov automata. In *FORMATS*, LNCS. Springer, 2013.