# Automated verification of randomised distributed algorithms

Nathalie Bertrand

Inria - Irisa, Rennes – Team SUMO (http://www.irisa.fr/sumo/)

**Randomised distributed algorithms** Since the seminal work of Rabin [6], randomisation has proven to be a powerful tool to solve computationally hard problems. In particular, in the field of distributed computing, probabilities can yield more efficient solutions, or even permit to solve problems that are otherwise unsolvable. An example is the central problem of consensus for asynchronous message-passing systems, which admits no deterministic solution when as few as one process can crash [3], and for which Ben Or proposed a randomised solution [1]. The correctness of this randomised algorithm, assuming more than half of the processes are correct, can be formalised by *qualitative properties*, including probabilistic wait-free termination: "independently of the initial configuration, almost-surely all correct processes output a value".

**The need for formal verification** Despite the appearant simplicity of Ben Or's algorithm, only paper-and-pencil proofs of the properties it ensures appear in the literature. However, the combination of distributed aspects and probabilities makes human reasoning difficult, as observed by Lehmann and Rabin: "proofs of correctness for probabilistic distributed systems are extremely slippery" [5]. Formal verification techniques can be extremely useful in this context: they would avoid tedious and error-prone manual proofs.

**Parameterized verification** Distributed algorithms are designed to run on systems composed of arbitrarily many agents. The automated verification of such crowd systems [2] is challenging: its aim is to validate at once all instances of the model, independently of the (parameterised) number of agents. A promising approach is thus to develop *ad hoc* parameterized verification techniques to automatically verify the correctness of randomised distributed algorithms.

**Internship objectives** To model randomised distributed algorithms, we propose to build on the existing model of threshold automata, used to prove safety properties of fault-tolerant non-probabilistic distributed algorithms [4]. Parameters of the execution of the algorithm, namely the number of participants and the number of faulty processes, appear in guards of threshold automata. The first objective of the internship will be to design a model for randomised distributed algorithms, that extends threshold automata with multiple rounds, and obviously with random choices. A second objective will be to develop decision algorithms for simple qualitative properties. For example, the correctness of randomised distributed algorithms for consensus includes probabilistic wait-free termination: almost-surely all correct processes decide on a value. Finally,

to demonstrate the applicability of the approach, if time permits, the last objective will be to implement a light prototype tool to verify an example case-study, such as Ben Or's algorithm.

This internship comes within the scope of a broader research project on the verification and synthesis of probabilistic parameterised systems. It thus can lead to a PhD. The long-term goal is to tackle the verification of quantitative properties to assess the performances of randomised distributed algorihms. One would typically aim at proving automatically that: "the expected number of rounds before termination is logarithmic in the number of processes", or "the probability that all correct processes decide before 10 rounds is at least .85 when less than 20 processes are involved".

**Keywords:** parameterized verification, probabilistic models, randomised distributed algorithms

**Supervisor:** Nathalie Bertrand

Web: http://people.rennes.inria.fr/Nathalie.Bertrand

E-mail: nathalie.bertrand@inria.fr

# References

[1] M. Ben Or. Another advantage of free choice: Completely asynchronous agreement protocols. In *Proceedings of the Second Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing (PODC'83)*, pages 27–30. ACM, 1983.

[2] J. Esparza. Keeping a crowd safe: On the complexity of parameterized verification (invited talk). In *Proceedings of the 31st International Symposium on Theoretical Aspects of Computer Science (STACS'14)*, volume 25 of *LIPIcs*, pages 1–10. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2014.

[3] M. J. Fischer, N. A. Lynch, and M. Paterson. Impossibility of distributed consensus with one faulty process. *Journal of the ACM*, 32(2):374–382, 1985.

[4] I. Konnov, M. Lazic, H. Veith, and J. Widder. A short counterexample property for safety and liveness verification of fault-tolerant distributed algorithms. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'17)*, pages 719–734. ACM, 2017.

[5] D. J. Lehmann and M. O. Rabin. On the advantages of free choice: A symmetric and fully distributed solution to the dining philosophers problem. In *Proceedings of the 8th Annual ACM Symposium on Principles of Programming Languages (POPL'81)*, pages 133–138. ACM Press, 1981.

[6] M. O. Rabin. Probabilistic algorithms. In *Algorithms and Complexity: New directions and recent results*, pages 21–39. Academic Press, 1976.