



## Habilitation Thesis Reviewer's Report

<b>Masaryk University</b>	Faculty of Informatics
<b>Faculty</b>	Informatics
<b>Procedure field</b>	Informatics
<b>Applicant</b>	Dr. rer.nat. RNDr. Mgr. Bc. Jan Křetínský, Ph.D.
<b>Applicant's home unit, institution</b>	Technical University of Munich, Germany
<b>Habilitation thesis</b>	<i>Modern Probabilistic Verification</i>
<b>Reviewer</b>	Prof. Orna Kupferman
<b>Reviewer's home unit, institution</b>	School of Computer Science and Engineering, Hebrew University, Israel

Please see attached pdf.

**Reviewer's questions for the habilitation thesis defence** (number of questions up to the reviewer)

Please see attached pdf.

### Conclusion

The habilitation thesis entitled “*Modern Probabilistic Verification*” by Jan Křetínský ~~does not fulfil~~ **fulfils** requirements expected of a habilitation thesis in the field of Informatics.

In Jerusalem on



## Jan Kretínský, Evaluation of Habilitation Thesis

The Habilitation thesis of Jan Kretínský is of a very high scientific level. It is based on an impressive collection of contributions in the area of verification of probabilistic systems, and in fact goes beyond the probabilistic setting.

Formal verification is the study of algorithms and tools for the development of correct hardware and software systems. The systems are non-terminating reactive systems that interact with their environment (e.g., operating systems, airplanes, internet protocols). Accordingly, their specifications describe on-going behaviors (e.g., every request is eventually granted), and should be satisfied in all environments with which the system may interact. The most prominent problem in formal verification is model checking ? the problem of deciding whether a formal model of a system satisfies a desired specification. Jan's thesis consider the setting where the systems are probabilistic, and so are the different measures for correctness. The thesis contains an exhaustive analysis of the different settings, verification techniques, and correctness measures:

- The setting: the probabilistic systems may be fully stochastic, combine stochastic and nondeterministic moves, as well as controllable and uncontrollable actions. Orthogonally, the systems may range from fully specified (a.k.a. white boxes) to systems that are observable only by their behaviors (a.k.a. black boxes).
- Verification techniques: typically (but not entirely) depending on the setting, the model-checking procedure may be based on state exploration or statistical methods. Again, orthogonally, many approaches for coping with large systems are needed.
- Correctness measures: in the non-probabilistic setting, we want all the computations of the system to be correct with respect to all possible environments. In the probabilistic one, there is room for several quantitative measures, referring to the probability of a computation to satisfy the (often quantitative) specification.

For the setting of white-box systems and statistical methods, Chapter 2 of the thesis describes methods for improving traditional verification techniques that are based on simulation and machine learning. Essentially, the techniques are based on tightening traditional algorithms by approximating the desired value from both above and below, and using the special structure of particular modalities of the specification formalism of Linear Temporal Logic (LTL). The contribution also includes practical considerations, like the data structures used by the different algorithms.

For the setting of black-box systems and traditional model-checking methods, Chapter 3 of the thesis describes a line of works whose highlight is the LICS 18 papers on translation of LTL formulas to deterministic automata. This is the chapter that is closest to my research, and I can witness that this line of work forms a beautiful and important contribution, which would have

a big impact. Let me elaborate about. The automata-theoretic approach to model-checking and synthesis uses the theory of automata as a unifying paradigm for system specification, verification, and synthesis. The approach considers the relationships between systems and their specifications as relationships between languages. Then, for example, questions about correctness of systems with respect to their specifications can be reduced to questions such as containment of the language of computations generated by the system in the language of computations satisfying the specification. The automata-theoretic approach separates the logical and the combinatorial aspects of reasoning about systems. The translation of specifications to automata handles the logic and shifts all the combinatorial difficulties to automata-theoretic problems, yielding clean and asymptotically optimal algorithms. Automata are the key to a large number of algorithms, heuristics, and tools. For probabilistic systems, one has to use deterministic automata. Known (that is, before Jan's work) translations of formulas to deterministic automata use intermediate nondeterministic automata and thus involve complicated determinization constructions.

Beyond avoiding a determinization construction, these translations suggested by Jan and his coauthors have two important advantages: unlike the traditional translations of LTL to deterministic automata, which yield monolithic automata, the novel translations avoid the intermediate nondeterministic automata and retain in their structure a direct and immediate connection to the original LTL formula. Indeed, the translations share the principle of describing each state by a collection of formulas, as happens in the classical tableaux construction for translation of LTL formulas into nondeterministic automata. This makes them particularly apt for semantic-based state reductions, e.g., for merging states corresponding to equivalent formulas; such reductions cannot be applied to Safra-based constructions, where this semantic structure gets lost. Moreover, the automata that the novel construction produce are boolean combinations of small automata, which makes them very promising for compositional and symbolic implementations. The translations have been implemented in the Rabinizer tool. Further, they have already been shown to be useful in two different areas: probabilistic model-checking and synthesis. First, a translation of LTL to so-called limit-deterministic Büchi automata has led to a new algorithm for quantitative probabilistic model-checking, implemented in the MoChiBa tool. MoChiBa outperforms PRISM, the reference tool for probabilistic model-checking, in numerous examples. Second, the translation from LTL to deterministic parity automata has been used to implement the Strix tool. The tool won in all categories for LTL-synthesis of the 2018 SYNTCOMP@CAV competition, which had not yet been achieved by any synthesis tool (the results of the competition are available at [www.syntcomp.org](http://www.syntcomp.org)).

Finally, richer correctness measures, where one examines the probabilistic systems with respect to quantitative properties, are studied in Chapter 4. This includes algorithms for model checking of specifications in probabilistic temporal logic, measuring the distance between probabilistic systems, and a study of the quantitative behavior of systems by payoff functions that refer to long-run average rewards and costs – all involve nice and new ideas and techniques.

The only less positive comment I have is that I find the papers describing the contributions

to be of a higher quality than the chapters of the thesis that introduce them (well, the papers are of a very high quality, so this is not a negative statement...). The organization of the chapters is excellent, but they assume a reader that is familiar with the setting and the technical issues: there is hardly any intuitive explanations and the high-level picture is not sufficiently emphasized.

### Questions for the defence

1. Explain what the problem is in using nondeterministic (rather than deterministic) automata in the context of reasoning about probabilistic systems. For games, researchers have suggested the use of Good-For-Games (GFG) nondeterministic automata. Are these automata good also for probabilistic reasoning?
2. What are your thoughts regarding an improvement of the translations in Chapter 3 so that they result in parity (rather than Rabin) automata?
3. The techniques described in the thesis are tailored for reasoning about probabilistic systems. Can we use the ideas in these techniques also for an improvement of reasoning about deterministic systems?