

Supervisor's Opinion on the Ph.D. Thesis of

Jan Fiedor

The Ph.D. thesis of Jan Fiedor concentrates on *automated methods of finding errors in concurrent programs*. This area of research is rather active since concurrent programming is getting more and more frequent. At the same time, such programming is significantly more demanding than in the case of sequential programs, and it can easily lead to nasty errors that are difficult to discover but can have disastrous consequences in practice. The fact that methods for finding such errors are highly needed and that the currently existing methods and tools for finding such errors are—despite a lot of research effort going into the area—not yet satisfactory can be illustrated on the recent cooperation of Jan Fiedor on applying and optimising his results for the needs of Honeywell Technology Solutions. The main approaches that the thesis of Jan Fiedor is focusing on are *noise-based testing* and *dynamic analysis*, which are two common and to a large degree complementary approaches to increasing chances to spot rare concurrency errors during testing. The former approach influences the scheduling of threads by inserting various kinds of noise, while the latter is extrapolating the behaviour seen in testing runs, looking for symptoms of errors.

The research of Jan Fiedor was supervised by me and conducted within the VeriFIT research group at the Faculty of Information Technology of the Brno University of Technology. However, Jan has collaborated with a number of other researchers, including highly renowned foreign researchers such as Shmuel Ur from Israel or Joao Lourenco from Portugal (where Jan spent several working stays during his studies). This collaboration has led and will lead to multiple common publications. I would also like to stress that Jan has actively supported and supports several younger students working within VeriFIT on methods of dynamic analysis and noise-based testing.

The research conducted by Jan was an important part of multiple research projects including project P103/10/0306 “Static and Dynamic Verification of Programs with Advanced Features of Concurrency and Unboundedness” of the Czech Science Foundation, the Czech Ministry of Education project COST OC10009 “Dealing with Complex Data Structures and Concurrency within the Rich Model Toolkit” (and the associated European COST action IC0901 “Rich Model Toolkit”), the Czech-Israeli Kontakt II project of the Czech Ministry of Education LH13265 “Intelligent Testing and Analysis of Concurrent Software”, the Czech Ministry of Education project COST LD14001 “Automatic Analysis and Verification of Transactional Memories” (and the associated European COST action IC1001 “Euro-TM”), the European COST action IC1402 “Runtime Verification beyond Monitoring” (ARVI), the European Artemis JU project HoliDes, as well as several FIT BUT institutional projects (including the IT4I Centre of Excellence).

The main contributions of the research of Jan Fiedor presented in his thesis include:

- A combination of *dynamic analysis* supported with noise injection, *recording* of key points in the witnessed behaviours, their *replaying* in a model checker (using its state space exploration capabilities to navigate through the not recorded parts of the traces), and sub-sequent *bounded model checking* in the vicinity of the recorded trace to confirm existence of errors

about which dynamic analysis has warned. Experiments showed that this approach can reduce the number of false alarms raised by dynamic analysis and relax problems with state explosion in model checking.

- Proposal of techniques for *monitoring the execution of concurrent C/C++ programs on the binary level* that led to the *ANaConDA framework* for easy construction of dynamic analyses for C/C++ programs, supported with noise injection. The framework can be easily instantiated to support different concurrency libraries, allows one to analyse even code for which there is no source code (or where parts are implemented in assembly which is common, e.g., in low-level system code), and supports numerous advanced features of binary code (such as trampolines, merged finalization of functions, or self-modifying code). The excellent applicability of ANaConDa has proved in many experiments, often in collaboration with external partners, such as Honeywell.
- Proposal of various *novel noise injection techniques* (such as the asymmetric read-write noise) which have been experimentally proven to provide much better potential for finding bugs in concurrent programs than other kinds of noise.
- Proposal and experimental evaluation of several (both light-weight and heavy-weight) approaches for *monitoring programs using software transactional memories*. This result has also experimentally showed how difficult it is to monitor such programs when one is interested not only in correctness but also performance properties.
- Several new dynamic analysis for finding violation of *contracts for concurrency*, based both on lock-sets as well as the happens-before relation (and vector clocks). These dynamic analyses allow one to deal with fairly generic properties that are beyond the scope of common, specialised analyses targeting, e.g., race conditions or various specific kinds of atomicity violations. Using contracts, testers can easily express specific error patterns to be sought in behaviours of tested of programs, which can significantly ease their work.

Apart from the above results, Jan has published several further ones related to a uniform classification of concurrency errors, modechart verification of real-time systems, and he has even touched upon the area of using mathematical methods in economics. The results of Jan were published in six conference proceedings published in the *LNCS series of Springer-Verlag* (EUROCAST'11, RV'11, RV'12, MEMICS'14, and EUROCAST'15) and one proceedings published by *ACM Press* (PADTAD'12). Moreover, Jan is also a co-author of a paper published in the well-established journal of *Software Testing, Verification and Reliability* (with WoS 2015 impact factor 1.082). One more paper is currently being prepared for the renowned International Conference on Software Testing (ICST)—at the time of writing this report, the paper is almost ready.

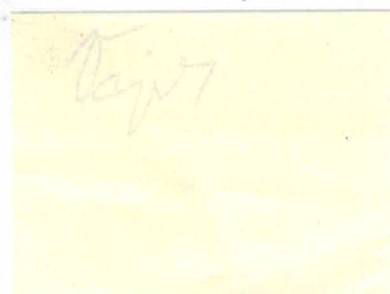
Despite all the papers have some co-author or co-authors, I can acknowledge that Jan contributed by key ideas as well as by a very sophisticated implementation and experiments to all of them, with his overall contribution being consistently higher than the plain numerical one given by the number of authors. Moreover, I would like to stress that the tool paper published at RV 2012 received the *Best Tool Paper Award*.

Finally, let me once again highlight the *ANaConDa framework* which Jan has devoted a lot of time to and which has achieved a degree of maturity that is well above common academic prototypes. This tool is very helpful for the VeriFIT group, for technology transfer from academia to industry (as witnessed by the already mentioned cooperation with Honeywell), and I believe that it will be further developed in the future. Indeed, the existence of ANaConDa makes it much easier for VeriFIT to participate on several H2020 ECSEL proposals that are currently being prepared for submission in September 2016.

Within his Ph.D. studies, Jan Fiedor has proved to have creative abilities, independence, and to be able to work hard. He has also proved to be capable of tight international cooperation, to work on collaborative research projects, to produce practically applicable results, and to support other students. In my opinion, the thesis of Jan Fiedor satisfies all requirements usually associated with Ph.D. theses in the area of computer science, and I therefore recommend it to be accepted.

Brno, August 30, 2016

Prof. Tomáš Vojnar

A yellow rectangular sticky note is placed over the printed name. It contains a handwritten signature in dark ink that appears to read "T. Vojnar".