



Report on the Ph.D. thesis presented by Zdeněk Letko

“Analysis and Testing of Concurrent Programs”

Multi-core programming is one of the biggest challenges the software industry is facing. With the advent of multi- and many-core CPUs, we need to have the right programming paradigms, techniques and tools, to harness the new parallel processing capabilities of these architectures. Software for these architectures must strongly resort to concurrent programming, and developing correct and reliable concurrent programs is far from trivial.

This thesis specifically addresses the timely and important problem of correctness of concurrent programs. In particular, it gives valuable contributions towards finding concurrency errors in complex multi-threaded software systems written in object-oriented programming languages, especially in Java. It proposes an elegant combination of techniques used in software testing aiming at identifying concurrency errors, i.e., errors resulting from the execution of multiple concurrent threads in a single program. These techniques include noise injection, to interfere with the normal thread scheduling and increase the probability of occurrence of less common interleaving on repeated executions of the testing benchmarks; dynamic program analysis that allows a better understanding of the actual program behavior for a specific execution of a testing benchmark, but also allows to make inferences about possible erroneous behaviors that, although were not observed in a specific execution of the testing benchmark, can possibly occur in other executions of the same benchmark; and the use of stochastic optimization algorithms to control scheduling of the threads and to steer the execution to improve coverage and uncover less probable program states and potential errors.

The thesis, of 139 pages, is organized in 6 chapters and 2 appendixes. It makes the following major contributions (Chapters 3 through 5):

1. Proposal of methodology to combine information acquired from dynamic analysis to derive new coverage metrics. This methodology is based in the intuition that some events observed during the execution of a program are reliable hints for concurrency errors, and is used to derive several new coverage metrics, which are then evaluated against existing metrics used in saturation- and search-based testing. Empirical evaluation of the proposed metrics shows evidence that, in some situations, they improve considerably over the existing ones.
2. Proposal of a new noise injection heuristics for concurrency coverage, which considers the information gathered from previous execution of the benchmark to bias the noise-generator in the next execution. Two proposed heuristics were evaluated against other state-of-the-art noise injection heuristics and demonstrated that they are more effective in some cases.
3. Suggestion of using search techniques to improve the quality of noise-based testing and dynamic analysis, by finding suitable combinations of parameters of tests and noise heuristics, which was formalized as the *test and noise configuration search (TNCS) problem*. The strategy uses genetic algorithms and the thesis includes a proposal of a complex objective function suitable for data race detection, based on the GoldiLocks dynamic analysis algorithm.





These contributions are complemented in each of the above chapters with a related work and validation sections. Other chapters include an introduction (Chapter 1), a detailed description of background concepts and tools for testing concurrent programs (Chapter 2), and a conclusion (Chapter 6). The thesis also include two appendixes, one reporting on the extensive experimentation with parameters of Genetic Algorithms for optimizing the TNCS problem (Appendix A), and another describing SearchBestie, the infrastructure for testing concurrent programs used for the experimental evaluation of the ideas proposed in this thesis.

Related work is discussed in sufficient details and the bibliography is quite complete. Presentation is good, with a clear presentation of technical content.

I found the contributions of this thesis to be original and of quality. They cover an important and still much unexplored field of texting concurrent programs. The thesis has a good balance between the description of the related work, the proposal of new ideas and concepts, and the reporting of their experimental validation and analysis of the obtained results.

I would like to stress that the candidate has co-authored an impressive collection of publications in the context of his thesis. This alone is a good indication of the quality of the thesis!

Throughout the manuscript, the candidate demonstrates a deep understanding of the domain and a mastering of the spectrum of instruments necessary to conduct research, from theory to practice. In my opinion, the candidate shows in this thesis an excellent capability at focusing on scientific problems and bringing convincing solutions to non-trivial research questions.

In summary, I found this manuscript to be well written and suitable for publication. The contributions of this thesis are of practical interest to the software engineering community, and contribute to knowledge in the field. The methodology is appropriate and the accomplished objectives are of quality. This work is original and tends to demonstrate that the candidate has the capacity to carry out independent research.

I therefore recommend this thesis for acceptance without reservation.

João Lourenço
(Assistant Professor at FCT-UNL, Portugal)