1 Summary of the Most Important Scientific Results

The project resulted in 17 conference papers and 3 journal articles all of which were accepted to international peer-reviewed venues (http://www.ahmedrezine.com/ceniit-12-04/publications/). The project made possible the extension of automatic verification techniques to:

New classes of concurrent programs. Concurrent programs exhibit an intractable number of possible interleavings among their concurrent sequential parts. The project allowed for defining and comparing new abstraction based model-checking techniques that could establish or refute safety properties (e.g., assertion violations, runtime errors, deadlocks) for programs creating an arbitrary but finite number of concurrent threads whose correctness might depend on invariants relating values of shared variables to numbers of threads in given states, on synchronization barriers or on complex constructs such as the Habanero Java phasers [1, 2, 3, 4].

Side-channel attacks. Computer hardware makes use of complex micro-architectural constructions (e.g., caches, speculative execution) to optimize program executions. New security attacks leverage such constructions to retrieve sensitive information (e.g., secret crypto keys, other processes’ virtual memories). The project made possible starting a framework [5] to formally assess the vulnerability of existing mainstream software to cache-based attacks. This work has been invited to a special issue to the ACM Transactions on Embedded Computing (TECS). This line of work resulted in a Singapore-based project where the CENIIT project leader co-investigates, with Prof. Sudipta Chattopadhyay from SUTD-Singapore, a project on the “Verification and Validation of Side-channel Freedom project”. The Singapore-based project is supported by a highly competitive three-years grant from the Singapore Ministry of Education (MOE).

Highly concurrent data structures. Concurrent libraries are prone to errors because complex fine grained locking or non-blocking schemes are adopted. The project allowed for the participation in the development of automatic verification techniques that handle arbitrary numbers of concurrent threads, without limiting the heap size or the data domains and in the absence of garbage collectors (best paper award [6] at TACAS’13 and its journal version [7]).

Weak memory models. These relaxations on the orders at which instructions appear to different threads result in extremely intricate behaviors. The problem appears for highly specialized code meant to be intensively used by concurrent software. The project allowed for the participation in several contributions that resulted in new approaches for the automatic placement of memory fences and for the removal of undesired additional behaviors [8, 9, 10].
Hybrid transactional memories. Hybrid transactional memories leverage on the versioning capabilities of underlying cache coherence protocols to use the existing parallelism while giving the programmer the illusion of atomicity. The project allowed for adapting and developing approaches to establish strict serializability, liveness and coherence of the underlying protocols [11, 12].

String manipulating programs. Many programs and scripts manipulate strings to deal with usernames, internet links, commands, keywords etc. The project allowed for key contributions in defining decision procedures for a logic that combines word equations over string variables denoting words of arbitrary lengths, together with constraints on the length of words, and on the regular languages to which words belong. This work was implemented and allowed for the verification and debugging of programs that were beyond the capabilities of existing decision procedures [13, 14, 15].

2 Summary of degrees and promotions

The PhD thesis of Yunyun Zhu (defended spring 2018, Uppsala) on “Caches, Transactions and Memories: Models, Coherence and Consistency” was partially supervised within the project. The PhD thesis of Zeinab Ganjei (defense planned autumn 2019, Linköping) on the “Algorithmic Verification of Concurrent Programs” is also supervised within the project. The project leader was promoted to assistant professor on February 1st 2016 and a docent lecture is planned for October 1st 2018. The project leader has been appointed vice-chair (proprefekt) since January 2018 of the Computer and Information Science Department (IDA). He has also been director of undergraduate education (2015-2017) of the Software and Systems (Sns) division of the Computer and Information Science Department (IDA).

3 Relevant thesis works and teaching

The project leader was involved in the supervision and the examination of several relevant Bachelor and Masters projects. Among them, [16] explored using SMT solvers to aid test case generation for constrained feature models, [17] extracted analyzable models from multi-threaded programs, [18] studied the effects of mutation testing on safety critical software, [19] studied how to improve MCDC adequate test sets for safety critical software to be RORG adequate, and [20] studied the applicability of learning based techniques leveraging model-checking tools to improve testing of safety critical software.

The project leader teaches/taught a software verification course (advanced level), a satisfiability modulo theory (SMT) and optimization course (graduate level), and an introduction to automatic verification course (graduate level). He also participates in the teaching of software security (advanced level), Software testing (advanced level), and advanced software engineering (advanced level).

4 Funded persons

The project funded 30% of the project leader’s costs (salary, premises, overhead) in addition to the participation to conferences and workshops.

5 Industrial Connections made possible by the project

The project allowed for collaborations with Saab via several student projects [18, 19] and the participation in the Saab led Vinnova NFFP6 project PILOT - Platform Independent Level of Testing. There has also been collaborations with Ericsson via student projects [16].
6 New research group

The project allowed the applicant to supervise two PhD students, one in Uppsala and one in Linköping, to conduct world-class research, to collaborate with national and international researchers and to introduce automatic verification techniques in graduate and undergraduate courses at LiU. In collaboration with the research activities of the ESLAB group at IDA, the CENIIT project will have enabled fostering the beginning of a formal verification group at LiU.

References


