Research internship (Master M2)

Title Verification of concurrent systems with data structures

Description

We are interested in verification techniques for realistic and powerful (physically) distributed programs that may access unbounded data structures such as queues or stacks. More precisely, each process is described by a finite state machine and, when performing a transition, a process may read from or write to some unbounded data structure. The architecture of such a system is defined by the set of processes, the set of data structures and their types (queues, stacks, bags, ...), and the relations `Reader` and `Writer` between data structures and processes.

Such systems are Turing powerful as soon as there is a process with a self-queue, or with two self-stacks, or there are two processes with queues between them, etc. Therefore, all interesting verification problems are undecidable in general.

To regain decidability, a very promising approach is to restrict the possible behaviors of systems by bounding some *parameter*. Our choice is to leave the data structures unbounded (otherwise we are in the well-studied setting of finite state systems) and to not compromise with their reliability (as, e.g., with lossy channels). Instead, we restrict to behaviors with bounded tree-width [3], or equivalently with bounded split-width [2]. This parameter subsumes many under-approximation techniques used so far (e.g., bounded context, bounded phase, bounded scope, ...) and gives decision procedures with good complexities.

The PhD thesis of Aiswarya Cyriac [1] will be the main reference for this internship. Our aim is to address some of the many problems left open there. For instance, the need to design good controllers for the various decidable restrictions. Ideally, we aim at controllers that are distributed, deterministic, non-blocking and that, when synchronized with a system, allow all and only those behaviors satisfying the considered restriction. Also, in the thesis, every data structure admits a single writer and a single reader. Is it possible to relax this hypothesis and to allow, for instance, several readers or several writers for some data structure?

References

[1] Aiswarya Cyriac. Verification of Communicating Recursive Programs via Split-width, *PhD thesis*, 2013. `http://www.lsv.ens-cachan.fr/~cyriac/cyriac-phd-thesis.pdf`

[2] Aiswarya Cyriac, Paul Gastin, and K. Narayan Kumar. MSO decidability of multi-pushdown systems via split-width. In *Proceedings of CONCUR'12*, LNCS 7454, pages 547–561. Springer, 2012.

[3] P. Madhusudan and Gennaro Parlato. The tree width of auxiliary storage. In *Proceedings of POPL'11*, pages 283–294. ACM, 2011.

Encadrants

Benedikt Bollig
`http://www.lsv.ens-cachan.fr/~bollig/`
Tél: 01 47 40 75 38
`bollig@lsv.ens-cachan.fr`

Paul Gastin
`http://www.lsv.ens-cachan.fr/~gastin/`
Tél: 01 47 40 75 60
`gastin@lsv.ens-cachan.fr`

Laboratoire Spécification et Vérification
École Normale Supérieure de Cachan
61, avenue du Président Wilson
94235 Cachan CEDEX