

# Sujet 7

## Time parallel simulation and/or state-full model-checking with DWARF

Proposé par : Martin Quinson

### General Information

Advisor Martin Quinson  
Address LORIA, Campus Scientifique - BP 239, 54506 Vandœuvre-lès-Nancy  
Phone 03 83 59 20 98  
Email Martin.Quinson@loria.fr  
Office B 122

### Context

Even if Distributed Systems are nowadays routinely used, they remain extremely difficult to study and understand. The two major aspects to assess are the system *correction* (whether it works – lack of deadlocks or faults) and its *performance* (throughput, achieved flops, etc). The SimGrid framework allows to study both aspects by combining a simulator and a model-checker in the same tool suite.

For sake of performance, these tools are implemented in the C language. They are thus very scalable, making it possible to simulate hundred thousands of processes on a single machine, and to simulate months of experimental scenario within minutes. But it complicates the automatic exploration of the simulated processes state, compared to languages such as Java, ruby or python in which code introspection of running code is trivial.

This state exploration is however mandatory for several improvement of the SimGrid simulator. First, it would allow to detect cycles in the simulation, that is, situations where the simulated world is exactly in the same state than earlier. In most cases, the simulation can then be stopped since simulating again exactly the same behavior is of little interest.

This idea can be pushed further in the *time-parallel approach* [2, 3], which consists in splitting the time line in parts and simulating each parts in parallel. In practice, the idea is to seed several possible state of the system and simulate their evolutions in parallel. Then, when the state of one simulated system *A* become close enough to the initial state of *B*, the history already simulated on *B* can be concatenated to the one simulated on *A*.

Automatic state exploration would also allow several improvement of the model-checker. First, if the checker would know about the user variables of the user code, it really ease the expression of properties that the code must verify. It would also allow state-full model-checking, i.e. the detection of redundant verification based on the state of the system (and not only based on the observed transitions as done currently).

DWARF is a debugging file format used by many compilers and debuggers to support source level debugging. It is for example used by the gdb debugger to allow the exploration of the program variables during the debugging session.

### Description

The goal of this internship is to leverage the `libdwarf` library to automatically explore the state of the user processes in SimGrid to implement one or several of the previously discussed improvements : time parallel simulation, better property expression and/or state-full model-checking.

**Skills Required :** C programing under Linux mandatory ; System programming appreciated.

### References

- [1] H. Casanova, A. Legrand, and M. Quinson. SimGrid : a Generic Framework for Large-Scale Distributed Experiments. In *10th IEEE International Conference on Computer Modeling and Simulation*. IEEE Computer Society Press, March 2008.
- [2] G. Wang, L. Bölöni, D. Turgut and D. Marinescu. Time-parallel simulation of wireless ad hoc networks with compressed history. In *J. Parallel Distributed Computing*, 69 :2 (February 2009).
- [3] T. Kiesling and S. Pohl. Time-Parallel Simulation with Approximative State Matching. In *Proceedings of the 18th Workshop on Parallel and Distributed Simulation*, 2004.