Programming adaptive real-time systems

Funded Ph. D. thesis proposal, Lille, France

Supervision Giuseppe Lipari (Professor), Julien Forget (Associate Professor).

Validity The Ph.D. can begin any time in 2018. The duration is 3 years.

To apply `julien.forget@univ-lille1.fr` `giuseppe.lipari@univ-lille1.fr`

1 Scientific context

A *real-time system* controls a physical device in its environment, at a rate adapted to the device evolution. This requires not only to compute correct values, but also to compute values at the right time. For instance, in the control-command system of an aircraft, computations must be performed quickly enough to react to external perturbations (such as a gust of wind for instance). Real-time systems can be found in several industrial domains, such as automotive, aeronautics, nuclear plants or automated production lines. They are typically critical systems, meaning that an incorrect value or a deadline miss can have catastrophic consequences.

Real-time systems usually consist of several execution configurations, also called *modes*. For instance, the flight system of an aircraft has a takeoff mode, a cruise mode, a landing mode, and degraded modes to support temporary failures. A system reconfiguration, or *mode change*, modifies the system tasks, by adding or removing tasks, or by modifying their timing constraints. During a mode change, the system goes through a transitory state. Due to the critical nature of real-time systems, it is extremely important to ensure the correctness of the system for any execution mode, but also during transitory states. The adaptive nature of multi-mode systems requires to consider difficult problems: we must ensure that the system behaviour is well-defined during transitory states, that tasks communicate correctly, that deadlines are met, etc.

2 Thesis objective

The objective of this thesis is to define a *formal* framework for *simplifying* adaptive real-time systems development, while ensuring the system *safety*. First, this requires to define programming language constructs to specify execution modes and mode changes in a simple and unambiguous manner. Then, semantics-preserving compilation into multi-threaded C code must be tackled. Finally, static analyses must be defined, to ensure consistent task communications (clock calculus [2]), and to ensure that deadlines are met (scheduling analysis [4]).

Our proposal is to define a data-flow language that enables to specify real-time constraints and mode changes. More precisely, we want to extend the Prelude language [3] to support synchronous mode automata [1]. The thesis objective goes beyond this extension, an important part will concern the definition of sound language primitives for the dynamic modification of real-time task parameters.

There are mainly two related research areas in the literature. First, schedulability analysis [5], to ensure that deadlines are met during mode changes. Second, conception and compilation of programming languages for the design of multi-mode systems (for instance [1]). Unfortunately, the first area focuses only on temporal aspects, while the second area focuses only on functional aspects. This thesis will consider functional and temporal aspects together, from the conception of a high-level programming language to the final embedded code. This will provide a safe and simplified development framework for real-time multi-mode adaptive systems.

3 Research team: Émeraude

The Émeraude team has 6 permanent members. It is part of the Secure Embedded Adapative Systems group (SEAS) of the CRIStAL laboratory (Lille, France). The main research direction of the team concerns timing management at every level of a computer application (language, compilation, operating system, virtual machine).

The thesis is part of a longer term research project (5-10 years) on the conception of real-time systems that can adapt dynamically, to support highly varying environments and hardware architectures. The thesis is pivotal in this project, which started 3 years ago. It is financed by the CORTEVA project "A Correct-by-construction methodology for supporting execution time variability in real-time systems", an ANR (french research agency) project that starts in January 2018.

References

[1] Jean-Louis Colaço, Bruno Pagano, and Marc Pouzet. A conservative extension of synchronous data-flow with state machines. In *Proceedings of the 5th ACM international conference on Embedded software*, pages 173–182. ACM, 2005.

[2] Jean-Louis Colaço and Marc Pouzet. Clocks as first class abstract types. In *International Workshop on Embedded Software*, pages 134–155. Springer, 2003.

[3] Julien Forget, Frédéric Boniol, David Lesens, and Claire Pagetti. A real-time architecture design language for multi-rate embedded control systems. In *Proceedings of the 2010 ACM Symposium on Applied Computing*. ACM, 2010.

[4] Chung Laung Liu and James W Layland. Scheduling algorithms for multi-programming in a hard-real-time environment. *Journal of the ACM (JACM)*, 20(1):46–61, 1973.

[5] Jorge Real and Alfons Crespo. Mode change protocols for real-time systems: A survey and a new proposal. *Real-time systems*, 26(2):161–197, 2004.