
IN PARTNERSHIP WITH:
**Institut polytechnique de Grenoble**

Activity Report 2018

# Project-Team SPADES

## Sound Programming of Adaptive Dependable Embedded Systems

IN COLLABORATION WITH: Laboratoire d'Informatique de Grenoble (LIG)

RESEARCH CENTER
**Grenoble - Rhône-Alpes**

THEME
**Embedded and Real-time Systems**

# Table of contents

# Project-Team SPADES

*Creation of the Team: 2013 January 01, updated into Project-Team: 2015 July 01*

**Keywords:**

**Computer Science and Digital Science:**

A1.1.1. - Multicore, Manycore
A1.1.9. - Fault tolerant systems
A1.3. - Distributed Systems
A2.1.1. - Semantics of programming languages
A2.1.6. - Concurrent programming
A2.1.9. - Synchronous languages
A2.3. - Embedded and cyber-physical systems
A2.3.1. - Embedded systems
A2.3.2. - Cyber-physical systems
A2.3.3. - Real-time systems
A2.4.1. - Analysis
A2.4.3. - Proofs
A2.5.2. - Component-based Design
A7.3. - Calculability and computability

**Other Research Topics and Application Domains:**

B5.2.1. - Road vehicles
B6.3.3. - Network Management
B6.4. - Internet of things
B6.6. - Embedded systems

# 1. Team, Visitors, External Collaborators

**Research Scientists**

Gregor Goessler [Team leader, Inria, Researcher, HDR]
Pascal Fradet [Inria, Researcher, HDR]
Alain Girault [Inria, Senior Researcher, HDR]
Sophie Quinton [Inria, Researcher]
Jean-Bernard Stefani [Inria, Senior Researcher]

**Faculty Member**

Xavier Nicollin [Institut polytechnique de Grenoble, Associate Professor]

**Post-Doctoral Fellows**

Jia Jie Wang [Inria, from May 2018]
Nicolas Hili [IRT Saint-Exupery]

**PhD Students**

Xiaojie Guo [Univ. Grenoble Alpes]
Maxime Lesourd [Univ. Grenoble Alpes]
Stephan Plassart [Univ. Grenoble Alpes]
Christophe Prévot [Thales until Oct 2018, Inria from Nov 2018]
Arash Shafiei [Orange Labs]

Martin Vassor [Inria]

**Technical staff**

Souha Ben Rayana [Inria, from May 2018]

**Intern**

Louise Penz [Univ. Grenoble Alpes, from Jun 2018 to Jul 2018]

**Administrative Assistant**

Helen Pouchot-Rouge-Blanc [Inria]

**Visiting Scientist**

Ismail Assayad [Univ. Hassan II, Casablanca, Sep 2018]

# 2. Overall Objectives

## 2.1. Overall Objectives

The SPADES project-team aims at contributing to meet the challenge of designing and programming dependable embedded systems in an increasingly distributed and dynamic context. Specifically, by exploiting formal methods and techniques, SPADES aims to answer three key questions:

1. How to program open networked embedded systems as dynamic adaptive modular structures?
2. How to program reactive systems with real-time and resource constraints on multicore architectures?
3. How to program reliable, fault-tolerant embedded systems with different levels of criticality?

These questions above are not new, but answering them in the context of modern embedded systems, which are increasingly distributed, open and dynamic in nature [24], makes them more pressing and more difficult to address: the targeted system properties – dynamic modularity, time-predictability, energy efficiency, and fault-tolerance – are largely antagonistic (*e.g.*, having a highly dynamic software structure is at variance with ensuring that resource and behavioral constraints are met). Tackling these questions together is crucial to address this antagonism, and constitutes a key point of the SPADES research program.

A few remarks are in order:

- We consider these questions to be central in the construction of future embedded systems, dealing as they are with, roughly, software architecture and the provision of real-time and fault-tolerance guarantees. Building a safety-critical embedded system cannot avoid dealing with these three concerns.
- The three questions above are highly connected. For instance, composability along time, resource consumption and reliability dimensions are key to the success of a component-based approach to embedded systems construction.
- For us, "Programming" means any constructive process to build a running system. It can encompass traditional programming as well as high-level design or "model-based engineering" activities, provided that the latter are supported by effective compiling tools to produce a running system.
- We aim to provide semantically sound programming tools for embedded systems. This translates into an emphasis on formal methods and tools for the development of provably dependable systems.

# 3. Research Program

## 3.1. Introduction

The SPADES research program is organized around three main themes, *Design and Programming Models*, *Certified real-time programming*, and *Fault management and causal analysis*, that seek to answer the three key questions identified in Section 2.1. We plan to do so by developing and/or building on programming languages and techniques based on formal methods and formal semantics (hence the use of *"sound programming"* in the project-team title). In particular, we seek to support design where correctness is obtained by construction, relying on proven tools and verified constructs, with programming languages and programming abstractions designed with verification in mind.

## 3.2. Design and Programming Models

Work on this theme aims to develop models , languages and tools to support a "correct-by-construction" approach to the development of embedded systems.

On the programming side, we focus on the definition of domain specific programming models and languages supporting static analyses for the computation of precise resource bounds for program executions. We propose dataflow models supporting dynamicity while enjoying effective analyses. In particular, we study parametric extensions where properties such as liveness and boundedness remain statically analyzable.

On the design side, we focus on the definition of component-based models for software architectures combining distribution, dynamicity, real-time and fault-tolerant aspects. Component-based construction has long been advocated as a key approach to the "correct-by-construction" design of complex embedded systems [49]. Witness component-based toolsets such as PTOLEMY [38], BIP [30], or the modular architecture frameworks used, for instance, in the automotive industry (AUTOSAR) [22]. For building large, complex systems, a key feature of component-based construction is the ability to associate with components a set of *contracts*, which can be understood as rich behavioral types that can be composed and verified to guarantee a component assemblage will meet desired properties.

Formal models for component-based design are an active area of research. However, we are still missing a comprehensive formal model and its associated behavioral theory able to deal *at the same time* with different forms of composition, dynamic component structures, and quantitative constraints (such as timing, fault-tolerance, or energy consumption).

We plan to develop our component theory by progressing on two fronts: a semantical framework and domain-specific programming models. The work on the semantical framework should, in the longer term, provide abstract mathematical models for the more operational and linguistic analysis afforded by component calculi. Our work on component theory will find its application in the development of a COQ-based toolchain for the certified design and construction of dependable embedded systems, which constitutes our first main objective for this axis.

## 3.3. Certified Real-Time Programming

Programming real-time systems (*i.e.*, systems whose correct behavior depends on meeting timing constraints) requires appropriate languages (as exemplified by the family of synchronous languages [32]), but also the support of efficient scheduling policies, execution time and schedulability analyses to guarantee real-time constraints (*e.g.*, deadlines) while making the most effective use of available (processing, memory, or networking) resources. Schedulability analysis involves analyzing the worst-case behavior of real-time tasks under a given scheduling algorithm and is crucial to guarantee that time constraints are met in any possible execution of the system. Reactive programming and real-time scheduling and schedulability for multiprocessor systems are old subjects, but they are nowhere as mature as their uniprocessor counterparts, and still feature a number of open research questions [28], [36], in particular in relation with mixed criticality systems. The main goal in this theme is to address several of these open questions.

We intend to focus on two issues: multicriteria scheduling on multiprocessors, and schedulability analysis for real-time multiprocessor systems. Beyond real-time aspects, multiprocessor environments, and multicore ones in particular, are subject to several constraints *in conjunction*, typically involving real-time, reliability and energy-efficiency constraints, making the scheduling problem more complex for both the offline and the online cases. Schedulability analysis for multiprocessor systems, in particular for systems with mixed criticality tasks, is still very much an open research area.

Distributed reactive programming is rightly singled out as a major open issue in the recent, but heavily biased (it essentially ignores recent research in synchronous and dataflow programming), survey by Bainomugisha et al. [28]. For our part, we intend to focus on devising synchronous programming languages for distributed systems and precision-timed architectures.

## 3.4. Fault Management and Causal Analysis

Managing faults is a clear and present necessity in networked embedded systems. At the hardware level, modern multicore architectures are manufactured using inherently unreliable technologies [33], [43]. The evolution of embedded systems towards increasingly distributed architectures highlighted in the introductory section means that dealing with partial failures, as in Web-based distributed systems, becomes an important issue.

In this axis we intend to address the question of *how to cope with faults and failures in embedded systems?*. We will tackle this question by exploiting reversible programming models and by developing techniques for fault ascription and explanation in component-based systems.

A common theme in this axis is the use and exploitation of causality information. Causality, *i.e.*, the logical dependence of an effect on a cause, has long been studied in disciplines such as philosophy [53], natural sciences, law [54], and statistics [55], but it has only recently emerged as an important focus of research in computer science. The analysis of logical causality has applications in many areas of computer science. For instance, tracking and analyzing logical causality between events in the execution of a concurrent system is required to ensure reversibility [52], to allow the diagnosis of faults in a complex concurrent system [47], or to enforce accountability [51], that is, designing systems in such a way that it can be determined without ambiguity whether a required safety or security property has been violated, and why. More generally, the goal of fault-tolerance can be understood as being to prevent certain causal chains from occurring by designing systems such that each causal chain either has its premises outside of the fault model (*e.g.*, by introducing redundancy [45]), or is broken (*e.g.*, by limiting fault propagation [57]).

# 4. Application Domains

## 4.1. Industrial Applications

Our applications are in the embedded system area, typically: transportation, energy production, robotics, telecommunications, the Internet of things (IoT), systems on chip (SoC). In some areas, safety is critical, and motivates the investment in formal methods and techniques for design. But even in less critical contexts, like telecommunications and multimedia, these techniques can be beneficial in improving the efficiency and the quality of designs, as well as the cost of the programming and the validation processes.

Industrial acceptance of formal techniques, as well as their deployment, goes necessarily through their usability by specialists of the application domain, rather than of the formal techniques themselves. Hence, we are looking to propose domain-specific (but generic) realistic models, validated through experience (*e.g.*, control tasks systems), based on formal techniques with a high degree of automation (*e.g.*, synchronous models), and tailored for concrete functionalities (*e.g.*, code generation).

## 4.2. Industrial Design Tools

The commercially available design tools (such as UML with real-time extensions, MATLAB/ SIMULINK/ dSPACE [1]) and execution platforms (OS such as VXWORKS, QNX, real-time versions of LINUX ...) start now to provide, besides their core functionalities, design or verification methods. Some of them, founded on models of reactive systems, come close to tools with a formal basis, such as for example STATEMATE by iLOGIX.

Regarding the synchronous approach, commercial tools are available: SCADE [2] (based on LUSTRE), CONTROLBUILD and RT-BUILDER (based on SIGNAL) from GEENSYS [3] (part of DASSAULT SYSTEMES), specialized environments like CELLCONTROL for industrial automatism (by the Inria spin-off ATHYS– now part of DASSAULT SYSTEMES). One can observe that behind the variety of actors, there is a real consistency of the synchronous technology, which makes sure that the results of our work related to the synchronous approach are not restricted to some language due to compatibility issues.

[1] http://www.dspaceinc.com
[2] http://www.esterel-technologies.com
[3] http://www.geensoft.com

## 4.3. Current Industrial Cooperations

Regarding applications and case studies with industrial end-users of our techniques, we cooperate with Thales on schedulability analysis for evolving or underspecified real-time embedded systems, with Orange Labs on software architecture for cloud services and with Daimler on reduction of nondeterminism and analysis of deadline miss models for the design of automotive systems.

# 5. New Software and Platforms

## 5.1. pyCPA_TWCA

*Analysis tool for weakly-hard real-time systems*
KEYWORDS: Real time - Scheduling analyses
FUNCTIONAL DESCRIPTION: pyCPA_TWCA is a pyCPA plugin for Typical Worst-Case Analysis. pyCPA is an open-source Python implementation of Compositional Performance Analysis developed at TU Braunschweig, which allows in particular response-time analysis. pyCPA_TWCA is an extension of that tool that is co-developed by Sophie Quinton and Zain Hammadeh at TU Braunschweig. It allows in particular the computation of weakly-hard guarantees for real-time tasks, i.e. number of deadline misses out of a sequence of executions. So far, pyCPA_TWCA is restricted to uniprocessor systems of independent tasks. pyCPA_TWCA can handle the following scheduling policies: Fixed Priority Preemptive, Fixed Priority Non-Preemptive, Weighted Round-Robin, Earliest Deadline First.

- Contact: Sophie Quinton

## 5.2. CertiCAN

*Certifier of CAN bus analysis results*
KEYWORDS: Certification - CAN bus - Real time - Static analysis
FUNCTIONAL DESCRIPTION: CertiCAN is a tool, produced using the Coq proof assistant, allowing the formal certification of the correctness of CAN bus analysis results. Result certification is a process that is lightweight and flexible compared to tool certification, which makes it a practical choice for industrial purposes. The analysis underlying CertiCAN, which is based on a combined use of two well-known CAN analysis techniques, is computationally efficient. Experiments demonstrate that CertiCAN is able to certify the results of RTaW-Pegase, an industrial CAN analysis tool, even for large systems. Furthermore, CertiCAN can certify the results of any other RTA tool for the same analysis and system model (periodic tasks with offsets in transactions).

- Contact: Xiaojie Guo

# 6. New Results

## 6.1. Design and Programming Models

**Participants:** Pascal Fradet, Alain Girault, Gregor Goessler, Xavier Nicollin, Christophe Prévot, Sophie Quinton, Arash Shafiei, Jean-Bernard Stefani, Martin Vassor, Souha Ben Rayana.

### 6.1.1. A multiview contract theory for cyber-physical system design and verification

The design and verification of critical cyber-physical systems is based on a number of models (and corresponding analysis techniques and tools) representing different viewpoints such as function, timing, security and many more. Overall correctness is guaranteed by mostly informal, and therefore basic, arguments about the relationship between these viewpoint-specific models. More precisely, the assumptions that a viewpoint-specific analysis makes on the other viewpoints remain mostly implicit, and whenever explicit they are handled mostly manually. In [11], we argue that the current design process over-constrains the set of possible system designs and that there is a need for methods and tools to formally relate viewpoint-specific models and corresponding analysis results. We believe that a more flexible contract-based approach could lead to easier integration, to relaxed assumptions, and consequently to more cost efficient systems while preserving the current modelling approach and its tools.

The framework we have in mind would provide viewpoint specific contract patterns guaranteeing inter-viewpoint consistency in a flexible manner. At this point, most of the work remains to be done. On the application side, we need a more complete picture of existing inter-viewpoint models. We also need the theory required for the correctness proofs, but it should be based on the needs on the application side.

### 6.1.2. End-to-end worst-case latencies of task chains for flexibility analysis

In collaboration with Thales, we address the issue of change during design and after deployment in safety-critical embedded system applications. More precisely, we focus on timing aspects with the objective to anticipate, at design time, future software evolutions and identify potential schedulability bottlenecks. The work presented in this section is the PhD topic of Christophe Prévot, in the context of a collaboration with Thales TRT, and our algorithms are being implemented in the Thales tool chain, in order to be used in industry.

This year, we have completed our work on the analysis of end-to-end worst-case latencies of task chains [10] that was needed to extend our approach for quantifying the flexibility, with respect to timing, of real-time systems made of chains of tasks. In a nutshell, flexibility is the property of a given system to accommodate changes in the future, for instance the modification of some of the parameters of the system, or the addition of a new task in the case of a real-time system.

One major issue that hinders the use of performance analysis in industrial design processes is the pessimism inherent to any analysis technique that applies to realistic system models (*e.g.*, , systems with task chains). Indeed, such analyses may conservatively declare unschedulable systems that will in fact never miss any deadlines. The two main avenues for improving this are (i) computing tighter upper bounds on the worst-case latencies, and (ii) measuring the pessimism, which requires to compute also guaranteed lower bounds. A lower bound is guaranteed by providing an actual system execution exhibiting a behavior as close to the worst case as possible. As a first step, we focus in [10] on uniprocessor systems executing a set of sporadic or periodic hard real-time task chains. Each task has its own priority, and the chains are scheduled according to the fixed-priority preemptive scheduling policy. Computing the worst-case end-to-end latency of each chain is complex because of the intricate relationship between the task priorities. Compared to state of the art analyses, we propose here tighter upper bounds, as well as lower bounds on these worst-case latencies. Our experiments show the relevance of lower bounds on the worst-case behavior for the industrial design of real-time embedded systems.

Based on our end-to-end latency analysis for task chains, we have also proposed an extension of the concept of slack to task chains and shown how it can be used to perform flexibility analysis and sensitivity analysis. This solution is particularly relevant for industry as it provides means by which the system designer can anticipate the impact on timing of software evolutions, at design time as well as after deployment.

### 6.1.3. Location graphs

We have introduced the location graph model [58] as an expressive framework for the definition of component-based models able to deal with dynamic software configurations with sharing and encapsulation constraints. We have completed a first study of the location graph behavioral theory (under submission), initiated its

formalization in Coq, and an implementation of the location framework with an emphasis of the expression of different isolation and encapsulation constraints.

We are now studying conservative extensions to the location graph framework to support the compositional design of heterogeneous hybrid dynamical systems and their attendant notions of approximate simulations [60].

In collaboration with the Spirals team at Inria Lille – Nord Europe, we have applied the location framework for the definition of a pivot model for the description of software configurations in a cloud computing environment. We have shown how to interpret in our pivot model several configuration management models and languages including TOSCA, OCCI, Docker Compose, Aeolus, OpenStack HOT.

### *6.1.4. Dynamicity in dataflow models*

Recent dataflow programming environments support applications whose behavior is characterized by dynamic variations in resource requirements. The high expressive power of the underlying models (*e.g.*, Kahn Process Networks or the CAL actor language) makes it challenging to ensure predictable behavior. In particular, checking *liveness* (*i.e.*, no part of the system will deadlock) and *boundedness* (*i.e.*, the system can be executed in finite memory) is known to be hard or even undecidable for such models. This situation is troublesome for the design of high-quality embedded systems. In the past few years, we have proposed several parametric dataflow models of computation (MoCs) [40], [31], we have written a survey providing a comprehensive description of the existing parametric dataflow MoCs [34], and we have studied *symbolic* analyses of dataflow graphs [35]. More recently, we have proposed an original method to deal with lossy communication channels in dataflow graphs [39].

We are now studying models allowing dynamic reconfigurations of the *topology* of the dataflow graphs. In particular, many modern streaming applications have a strong need for reconfigurability, for instance to accommodate changes in the input data, the control objectives, or the environment.

We have proposed a new MoC called Reconfigurable Dataflow (RDF) [15]. RDF extends SDF with transformation rules that specify how the topology and actors of the graph may be reconfigured. Starting from an initial RDF graph and a set of transformation rules, an arbitrary number of new RDF graphs can be generated at runtime. The major quality of RDF is that it can be statically analyzed to guarantee that all possible graphs generated at runtime will be connected, consistent, and live. This is the research topic of Arash Shafiei's PhD, in collaboration with Orange Labs.

### *6.1.5. Monotonic prefix consistency in distributed systems*

We have studied the issue of data consistency in distributed systems. Specifically, we have considered a distributed system that replicates its data at multiple sites, which is prone to partitions, and which is assumed to be available (in the sense that queries are always eventually answered). In such a setting, strong consistency, where all replicas of the system apply synchronously every operation, is not possible to implement. However, many weaker consistency criteria that allow a greater number of behaviors than strong consistency, are implementable in available distributed systems. We have focused on determining the strongest consistency criterion that can be implemented in a convergent and available distributed system that tolerates partitions, and we have shown that no criterion stronger than Monotonic Prefix Consistency (MPC [61], [44]) can be implemented [18].

## 6.2. Certified Real-Time Programming

**Participants:** Pascal Fradet, Alain Girault, Gregor Goessler, Xavier Nicollin, Sophie Quinton, Xiaojie Guo, Maxime Lesourd.

### 6.2.1. Time predictable programming languages and architectures

Time predictability (PRET) is a topic that emerged in 2007 as a solution to the ever increasing unpredictability of today's embedded processors, which results from features such as multi-level caches or deep pipelines [37]. For many real-time systems, it is mandatory to compute a strict bound on the program's execution time. Yet, in general, computing a tight bound is extremely difficult [64]. The rationale of PRET is to simplify both the programming language and the execution platform to allow more precise execution times to be easily computed [27].

We have extended the PRET-C compiler [25] in order to make it energy aware. To achieve this, we use dynamic voltage and frequency scaling (DFVS) and we insert DVFS control points in the control flow graph of the PRET-C program. Several difficulties arise: (i) the control flow graph is concurrent, (ii) the resulting optimization problem is a time and energy multi-criteria problem, and (iii) since we consider PRET-C programs, we actually address the Worst-Case Execution Time (WCET) and the Worst-Case Energy Consumption (WCEC). Thanks to a novel ILP formulation and to a bicriteria heuristic, we are able to address the two objectives jointly and to compute, for each PRET-C program, the Pareto front of the non-dominated solutions in the 2D space (WCET,WCEC) [63]. We have recently improved this result to reduce the complexity of the algorithm and to produce the *optimal* Pareto front. This is the topic of Jia Jie Wang's postdoc.

Moreover, within the CAPHCA project, we have proposed a new approach for predictable inter-core communication between tasks allocated on different cores. Our approach is based on the execution of synchronous programs written in the FOREC programming language on deterministic architectures called PREcision Timed. The originality resides in the time-triggered model of computation and communication that allows for a very precise control over the thread execution. Synchronisation is done via configurable Time Division Multiple Access (TDMA) arbitrations (either physical or conceptual) where the optimal size and offset of the time slots are computed to reduce the inter-core synchronization costs. Results show that our model guarantees time-predictable inter-core communication, the absence of concurrent accesses (without relying on hardware mechanisms), and allows for optimized execution throughput. This is the topic of Nicolas Hili's postdoc.

### 6.2.2. Schedulability of weakly-hard real-time systems

We focus on the problem of computing tight deadline miss models for real-time systems, which bound the number of potential deadline misses in a given sequence of activations of a task. In practical applications, such guarantees are often sufficient because many systems are in fact not hard real-time [4]. A weakly-hard real-time guarantee specifies an upper bound on the maximum number m of deadline misses of a task in a sequence of k consecutive executions. Based on our previous work on Typical Worst-Case Analysis [4], [8], we have introduced in [13] the first verification method which is able to provide weakly-hard real-time guarantees for tasks and task chains in systems with multiple resources under partitioned scheduling with fixed priorities. All existing weakly-hard real-time verification techniques are restricted today to systems with a single resource. Our verification method is applied in the context of switched networks with traffic streams between nodes, and we demonstrate its practical applicability on an automotive case study.

### 6.2.3. Synthesis of switching controllers using approximately bisimilar multiscale abstractions

The use of discrete abstractions for continuous dynamics has become standard in hybrid systems design (see *e.g.*, [60] and the references therein). The main advantage of this approach is that it offers the possibility to leverage controller synthesis techniques developed in the areas of supervisory control of discrete-event systems [56]. The first attempts to compute discrete abstractions for hybrid systems were based on traditional systems behavioral relationships such as simulation or bisimulation, initially proposed for discrete systems most notably in the area of formal methods. These notions require inclusion or equivalence of observed behaviors which is often too restrictive when dealing with systems observed over metric spaces. For such systems, a more natural abstraction requirement is to ask for closeness of observed behaviors. This leads to the notions of approximate simulation and bisimulation introduced in [42]. These approaches are based on sampling of time and space where the sampling parameters must satisfy some relation in order to obtain abstractions of a prescribed precision. In particular, the smaller the time sampling parameter, the finer the

lattice used for approximating the state-space; this may result in abstractions with a very large number of states when the sampling period is small. However, there are a number of applications where sampling has to be fast; though this is generally necessary only on a small part of the state-space.

We are currently investigating an approach using mode sequences as symbolic states for our abstractions. By using mode sequences of variable length we are able to adapt the granularity of our abstraction to the dynamics of the system, so as to automatically trade off precision against controllability of the abstract states.

### *6.2.4. A Markov Decision Process approach for energy minimization policies*

In the context of independent real-time sporadic jobs running on a single-core processor equipped with Dynamic Voltage and Frequency Scaling (DVFS), we have proposed a Markov Decision Process approach (MDP) to compute the scheduling policy that dynamically chooses the voltage and frequency level of the processor such that each job meets its deadline and the total energy consumption is minimized. We distinguish two cases: the finite case (there is a fixed time horizon) and the infinite case. In the finite case, several *offline* solutions exist, which all use the complete knowledge of all the jobs that will arrive within the time horizon [65], *i.e.*, their size and deadlines. But clearly this is unrealistic in the embedded context where the characteristics of the jobs are not known in advance. Then, an optimal offline policy called Optimal Available (OA) has been proposed in [65]. Our goal was to improve this result by taking into account the *statistical characteristics* of the upcoming jobs. When such information is available (for instance by profiling the jobs based on execution traces), we have proposed several speed policies that optimize the *expected* energy consumption. We have shown that this general constrained optimization problem can be modeled as an unconstrained MDP by choosing a proper state space that also encodes the constraints of the problem. In particular, this implies that the optimal speed at each time can be computed using a *dynamic programming* algorithm (under a finite horizon), and that the optimal speed at any time $t$ will be a deterministic function of the current state at time $t$ [41]. Under an infinite horizon, we use a *Value Iteration* algorithm.

This work led us to compare several existing speed policies with respect to their feasibility. Indeed, the policies (OA) [65], (AVR) [65], and (BKP) [29] all assume that the maximal speed $S_{max}$ available on the processor is infinite, which is an unrealistic assumption. For these three policies and for our (MDP) policy, we have established necessary and sufficient conditions on $S_{max}$ guaranteeing that no job will ever miss its deadline.

This is the topic of Stephan Plassart's PhD, funded by the CASERM Persyval project.

### *6.2.5. Formal proofs for schedulability analysis of real-time systems*

We have started to lay the foundations for computer-assisted formal verification of real-time systems analyses. Specifically, we contribute to Prosa [23], a Coq library of reusable concepts and proofs for real-time systems analysis. A key scientific challenge is to achieve a modular structure of proofs, *e.g.*, for response time analysis. Our goal is to use this library for:

1. a better understanding of the role played by some assumptions in existing proofs;
2. a formal verification and comparison of different analysis techniques; and
3. the certification of results of existing (*e.g.*, industrial) analysis tools.

Our first major result [16] is a task model that generalizes the digraph model [59] and its corresponding analysis for fixed-priority scheduling with limited preemption. The motivation for this work, which is not yet fully proven in Coq, is to obtain a formally verified schedulability analysis for a very expressive task model. In the context of computer assisted verification, it permits to factorize the correctness proofs of a large number of analyses. The digraph task model seems a good candidate due to its powerful expressivity. Alas, its ability to capture dependencies between arrival and execution times of jobs of different tasks is very limited. Our extended model can capture dependencies between jobs of the same task as well as jobs of different tasks. We provide a correctness proof of the analysis that is written in a way amenable to its formalization in the Coq proof assistant. Despite being much more general, the Response Time Analysis (RTA) for our model is not significantly more complex than the original one. Also, it underlines similarities between existing analyses, in particular the analysis for the digraph model and Tindell's offset model [62].

A second major result is CertiCAN, a tool produced using Coq for the formal certification of CAN analysis results. Result certification is a process that is light-weight and flexible compared to tool certification, which makes it a practical choice for industrial purposes. The analysis underlying CertiCAN is based on a combined use of two well-known CAN analysis techniques [62] that makes it computationally efficient. Experiments demonstrate that CertiCAN is able to certify the results of RTaW-Pegase, an industrial CAN analysis tool, even for large systems. This result paves the way for a broader acceptance of formal tools for the certification of real-time systems analysis results. Beyond CertiCAN, we believe that this work is significant in that it demonstrates the advantage of result certification over tool certification for the RTA of CAN buses. In addition, the underlying technique can be reused for any other system model for which there exist RTAs with different levels of precision. This work will be presented at RTAS 2019.

In parallel, we have completed and published in [17] a Coq formalization of Typical Worst-Case Analysis (TWCA) [4], [8], an analysis technique for weakly-hard real-time systems. Our generic analysis is based on an abstract model that characterizes the exact properties needed to make TWCA applicable to any system model. Our results are formalized and checked using the Coq proof assistant along with the Prosa schedulability analysis library. This work opens up new research directions for TWCA by providing a formal framework for the trade-off that must be found between time efficiency and precision of the analysis. Hopefully, our generic proof will make it easier to extend TWCA to more complex models in the future. In addition, our experience with formalizing real-time systems analyses shows that it is not only a way to increase confidence in the results of the analyses; it also helps understanding their key intermediate steps, the exact assumptions required, and how they can be generalized.

### *6.2.6. Logical execution time*

In collaboration with TU Braunschweig and Daimler, we have worked on the application of the Logical Execution Time (LET) paradigm [50], according to which data are read and written at predefined time instants, to the automotive industry. The LET paradigm was considered until recently by the automotive industry as not efficient enough in terms of buffer space and timing performance. The shift to embedded multicore processors has represented a game changer: The design and verification of multicore systems is a challenging area of research that is still very much in progress. Predictability clearly is a crucial issue which cannot be tackled without changes in the design process. Several OEMs and suppliers have come to the conclusion that LET might be a key enabler and a standardization effort is already under way in the automotive community to integrate LET into AUTOSAR. We have organized a Dagstuhl seminar [9] to discuss and sketch solutions to the problems raised by the use of LET in multicore systems. A white paper on the topic is under preparation.

So far, LET has been applied only at the ECU (Electronic Control Unit) level by the automotive industry. Recent developments in electric powertrains and autonomous vehicle functions raise parallel programming from the multicore level to the vehicle level where the standard LET approach cannot apply directly. We have proposed System Level LET [21], an extension of LET with relaxed synchronization requirements which allows separating network design from ECU design and makes LET applicable to automotive distributed systems.

### *6.2.7. Scheduling under multiple constraints and Pareto optimization*

We have continued our work on multi-criteria scheduling, in two directions. First, in the context of dynamic applications that are launched and terminated on an embedded homogeneous multi-core chip, under execution time and energy consumption constraints, we have proposed a two layer adaptive scheduling method [26]. In the first layer, each application (represented as a DAG of tasks) is scheduled statically on subsets of cores: 2 cores, 3 cores, 4 cores, and so on. For each size of these sets (2, 3, 4, ...), there may be only one topology or several topologies. For instance, for 2 or 3 cores there is only one topology (a “line”), while for 4 cores there are three distinct topologies (“line”, “square”, and “T shape”). Moreover, for each topology, we generate statically several schedules, each one subject to a different total energy consumption constraint, and consequently with a different Worst-Case Reaction Time (WCRT). Coping with the energy consumption constraints is achieved thanks to Dynamic Frequency and Voltage Scaling (DVFS). In the second layer, we use these pre-generated static schedules to reconfigure dynamically the applications running on the multi-core each

time a new application is launched or an existing one is stopped. The goal of the second layer is to perform a dynamic global optimization of the configuration, such that each running application meets a pre-defined quality-of-service constraint (translated into an upper bound on its WCRT) and such that the total energy consumption be minimized. For this, we *(i)* allocate a sufficient number of cores to each active application, *(ii)* allocate the unassigned cores to the applications yielding the largest gain in energy, and *(iii)* choose for each application the best topology for its subset of cores (*i.e.*, better than the by default "line" topology). This is a joint work with Ismail Assayad (U. Casablanca, Morocco) who visited the team in 2018.

Second, we have proposed the first of its kind multi-criteria scheduling heuristics for a DAG of tasks onto an homogeneous multi-core chip. Given an application modeled as a Directed Acyclic Graph (DAG) of tasks and a multicore architecture, we produce a set of non-dominated (in the Pareto sense) static schedules of this DAG onto this multicore. The criteria we address are the execution time, reliability, power consumption, and peak temperature. These criteria exhibit complex antagonistic relations, which make the problem challenging. For instance, improving the reliability requires adding some redundancy in the schedule, which penalizes the execution time. To produce Pareto fronts in this 4-dimension space, we transform three of the four criteria into constraints (the reliability, the power consumption, and the peak temperature), and we minimize the fourth one (the execution time of the schedule) under these three constraints. By varying the thresholds used for the three constraints, we are able to produce a Pareto front of non-dominated solutions. Each Pareto optimum is a static schedule of the DAG onto the multicore. We propose two algorithms to compute static schedules. The first is a ready list scheduling heuristic called ERPOT (Execution time, Reliability, POwer consumption and Temperature). ERPOT actively replicates the tasks to increase the reliability, uses Dynamic Voltage and Frequency Scaling to decrease the power consumption, and inserts cooling times to control the peak temperature. The second algorithm uses an Integer Linear Programming (ILP) program to compute an optimal schedule. However, because our multi-criteria scheduling problem is NP-complete, the ILP algorithm is limited to very small problem instances. Comparisons showed that the schedules produced by ERPOT are on average only 10% worse than the optimal schedules computed by the ILP program, and that ERPOT outperforms the PowerPerf-PET heuristic from the literature on average by 33%. This is a joint work with Athena Abdi and Hamid Zarandi from Amirkabir University in Tehran, Iran.

## 6.3. Fault Management and Causal Analysis

**Participants:** Pascal Fradet, Alain Girault, Gregor Goessler, Jean-Bernard Stefani, Martin Vassor.

### *6.3.1. Fault Ascription in Concurrent Systems*

The failure of one component may entail a cascade of failures in other components; several components may also fail independently. In such cases, elucidating the exact scenario that led to the failure is a complex and tedious task that requires significant expertise.

The notion of causality *(did an event $e$ cause an event $e'$?)* has been studied in many disciplines, including philosophy, logic, statistics, and law. The definitions of causality studied in these disciplines usually amount to variants of the counterfactual test "$e$ is a cause of $e'$ if both $e$ and $e'$ have occurred, and in a world that is as close as possible to the actual world but where $e$ does not occur, $e'$ does not occur either". In computer science, almost all definitions of logical causality — including the landmark definition of [48] and its derivatives — rely on a causal model that. However, this model may not be known, for instance in presence of black-box components. For such systems, we have been developing a framework for blaming that helps us establish the causal relationship between component failures and system failures, given an observed system execution trace. The analysis is based on a formalization of counterfactual reasoning [6].

We are currently working on a revised version of our general semantic framework for fault ascription in [46] that satisfies a set of formally stated requirements — such as its behavior under several notions of abstraction and refinement —, and on its instantiation to acyclic models of computation, in order to compare our approach with the standard definition of *actual causality* proposed by Halpern and Pearl.

#### 6.3.2. Fault Management in Virtualized Networks

From a more applied point of view we are investigating, in the context of Sihem Cherrared's PhD thesis, approaches for fault explanation and localization in virtualized networks. In essence, Network Function Virtualization (NFV), widely adopted by the industry and the standardization bodies, is about running network functions as software workloads on commodity hardware to optimize deployment costs and simplify the life-cycle management of network functions. However, it introduces new fault management challenges including dynamic topology and multi-tenant fault isolation that we discuss in [14]. As a first step to tackle those challenges, we have extended the classical fault management process to the virtualized functions by introducing LUMEN: a Global Fault Management Framework. Our approach aims at providing the availability and reliability of the virtualized 5G end-to-end service chain. LUMEN includes the canonical steps of the fault management process and proposes a monitoring solution for all types of Network virtualization Environments. Our framework is based on open source solutions and could easily be integrated with other existing autonomic management models.

# 7. Bilateral Contracts and Grants with Industry

## 7.1. Bilateral Contracts with Industry

- Inria and Orange Labs have established in 2015 a joint virtual research laboratory, called I/O LAB. We have been heavily involved in the creation of the laboratory and are actively involved in its operation (Jean-Bernard Stefani is one of the two co-directors of the lab). I/O LAB focuses on the network virtualization and cloudification. As part of the work of I/O LAB, we have cooperated with Orange Lab, as part of a cooperative research contract funded by Orange, on defining architectural principles and frameworks for network cloud infrastructures encompassing control and management of computing, storage and network resources.
- With Daimler (subcontracting via iUTBS): We have proposed, in collaboration with TU Braunschweig, an extension of the LET paradigm [50], called *System-level LET, to accommodate the specific needs of the design process in the automotive industry, in which the network structure must be made explicit in the LET program.*

## 7.2. Bilateral Grants with Industry

With Thales: Early performance assessment for evolving and variable cyber-physical systems. This CIFRE grant funds the PhD of Christophe Prévot.

With Orange: Programming IoT and sofware defined radio with dynamic dataflow models of computation. This CIFRE grant funds the PhD of Arash Shafiei.

# 8. Partnerships and Cooperations

## 8.1. Regional Initiatives

#### 8.1.1. CASERM (PERSYVAL-Lab project)

**Participants:** Pascal Fradet, Alain Girault, Gregor Goessler, Xiaojie Guo, Maxime Lesourd, Xavier Nicollin, Stephan Plassart, Sophie Quinton, Jean-Bernard Stefani, Martin Vassor.

Despite recent advances, there exists currently no integrated formal methods and tools for the design and analysis of reconfigurable multi-view embedded systems. This is the goal of the CASERM project.

The CASERM project represents a significant effort towards a COQ-based design method for reconfigurable multi-view embedded systems, in order to formalize the structure and behavior of systems and to prove their main properties. The use of a proof assistant to support such a framework is motivated by the fact that the targeted systems are both extremely complex and critical. The challenges addressed are threefold:

1. to model software architectures for embedded systems taking into account their dynamicity and multiple constraints (functional as well as non functional);
2. to propose novel scheduling techniques for dynamically reconfiguring embedded systems; and
3. to advance the state of the art in automated proving for such systems.

The objectives of CASERM that address these challenges are organized in three tasks. They consist respectively in designing an architecture description framework based on a process calculus, in proposing online optimization methods for dynamic reconfiguration systems (this is the topic of Stephan Plassart's PhD), and in developing a formal framework for real-time analysis in the COQ proof assistant (this is the topic of Xiaojie Guo's and Maxime Lesourd's PhD). A fourth task focuses on common case studies for the evaluation of the obtained results.

The CASERM consortium gathers researchers from the LIG and VERIMAG laboratories who are reknowned specialists in these fields. The project started in November 2016 and will last three years.

## 8.2. National Initiatives

### *8.2.1. ANR*

*8.2.1.1. RT-Proofs*

**Participants:** Pascal Fradet, Xiaojie Guo, Maxime Lesourd, Sophie Quinton.

RT-Proofs is an ANR/DFG project between Inria, MPI-SWS, Onera, TU Braunschweig and Verimag, running from 2018 until 2020.

The overall objective of the RT-Proofs project is to lay the foundations for computer-assisted formal verification of timing analysis results. More precisely, the goal is to provide:

1. a strong formal basis for schedulability, blocking, and response-time analysis supported by the Coq proof assistant, that is as generic, robust, and modular as possible;
2. correctness proofs for new and well-established generalized response-time analysis results, and a better, precise understanding of the role played by key assumptions and formal connections between competing analysis techniques;
3. an approach for the generation of proof certificates so that analysis results – in contrast to analysis tools – can be certified.

*8.2.1.2. DCore*

**Participants:** Gregor Goessler, Jean-Bernard Stefani.

DCORE is an ANR project between Inria project teams ANTIQUE, FOCUS and SPADES, and the IRIF lab, running from 2019 to 2023.

The overall objective of the project is to develop a semantically well-founded, novel form of concurrent debugging, which we call *causal debugging*, that aims to alleviate the deficiencies of current debugging techniques for large concurrent software systems. The causal debugging technology developed by DCORE will comprise and integrate two main novel engines:

1. *a reversible execution engine* that allows programmers to backtrack and replay a concurrent or distributed program execution, in a way that is both precise and efficient (only the exact threads involved by a return to a target anterior or posterior program state are impacted);
2. a *causal analysis engine* that allows programmers to analyze concurrent executions, by asking questions of the form "what caused the violation of this program property?", and that allows for the precise and efficient investigation of past and potential program executions.

### 8.2.2. Institute of Technology (IRT)

#### 8.2.2.1. CAPHCA

**Participants:** Alain Girault, Nicolas Hili.

CAPHCA is a project within the Antoine de Saint Exupéry IRT. The general objective of the project is to provide methods and tools to achieve performance and determinism on modern, high-performance, multi-core and FPGA-enabled SOCs. Our specific contribution lies withing work pakacges dedicated to the design of novel PRET architectures and programming languages (see Section 6.2.1).

## 8.3. European Initiatives

### 8.3.1. Collaborations in European Programs, Except FP7 & H2020

Program: Celtic-Plus

Project acronym: SENDATE

Project title: Secure Networking for a Data center cloud in Europe

Duration: April 2016 - March 2019

Coordinator: Nokia France

Other partners: Nokia, Orange, IMT, Inria

Abstract: The SENDATE project aims to develop a clean-slate architecture for converged telecommunications networks and distributed data centers supporting 5G cellular networks and the needs from the Industrial Internet and the Internet of Things. It aims to provide scientific and technical solutions for intra and inter data centrers security, control, management and orchestration, placement and management of virtual network functions, as well as high-speed transport networks for data centers access and interconnection.

### 8.3.2. Collaborations with Major European Organizations

We have a strong collaboration with the Technische Universität Braunschweig in Germany. In particular, Sophie Quinton is involved in the CCC project (http://ccc-project.org/) to provide methods and mechanisms for the verification of software updates after deployment in safety-critical systems, and in the TypicalCPA project which aims at computing deadline miss models for distributed systems.

We also have a recent collaboration with the MPI-SWS in Kaiserslautern (Germany) on formal proofs for real-time systems. This collaboration will be concretized by an ANR-PRCI project called RT-PROOFS starting in 2018, which involves MPI-SWS, TU Braunschweig, Inria, and Onera.

## 8.4. International Research Visitors

### 8.4.1. Visits of International Scientists

- Ismail Assayad (from U. Casablanca, Morocco) visited the team for one month in September 2018, to work on a two layer adaptive scheduling method.

# 9. Dissemination

## 9.1. Promoting Scientific Activities

### 9.1.1. Scientific Events Organisation

#### 9.1.1.1. General Chair, Scientific Chair

- Alain Girault is member of the steering committee of the International Federated Conference on Distributed Computing Techniques (DISCOTEC) and of the ACM International Conference on Embedded Software (EMSOFT).
- Gregor Gössler is member of the steering committee of the International Workshop on Causal Reasoning for Embedded and Safety-critical Systems Technologies (CREST).
- Jean-Bernard Stefani is the current chair of the steering committee of the IFIP FORTE international conference series, a member of the steering committee of the IFIP DISCOTEC conference series, and the current chair of the IFIP Working Group 6.1.

*9.1.1.2. Member of the Organizing Committees*

- Sophie Quinton was the co-organizer of a Dagstuhl seminar entitled "The Logical Execution Time Paradigm: New Perspectives for Multicore Systems". https://www.dagstuhl.de/18092

### *9.1.2. Scientific Events Selection*

*9.1.2.1. Member of the Conference Program Committees*

- Alain Girault served in the program committees of the Symposium on Industrial Embedded Systems (SIES'18), the Forum on specification and Design Languages (FDL'18), and the Conference on Applications of Concurrency to System Design (ACSD'18).
- Gregor Gössler served in the program committees of the 18th International Workshop on Automated Verification of Critical Systems (AVOCS 2018) and the 3rd international Workshop on Formal Reasoning about Causation, Responsibility, and Explanations in Science and Technology (CREST 2018).
- Sophie Quinton served in the program committees of the 30th Euromicro Conference on Real-Time Systems (ECRTS'18), the 9th International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS'18), the 39th IEEE Real-Time Systems Symposium (RTSS'18) and the 26th International Conference on Real-Time Networks and Systems (RTNS'18).

*9.1.2.2. Reviewer*

- Alain Girault reviewed papers for the ECRTS'18 conference.

### *9.1.3. Journal*

*9.1.3.1. Member of the Editorial Boards*

- Alain Girault is a member of the editorial board of the Journal on Embedded Systems.

*9.1.3.2. Reviewer - Reviewing Activities*

- Alain Girault reviewed articles for J. of Transportation Technologies (JTT) and IEEE Trans. Dependable and Secure Computing (TDSC).
- Gregor Gössler reviewed articles for IEEE Transactions on Automatic Control (TAC) and ACM Transactions on Embedded Computing Systems (TECS).
- Sophie Quinton reviewed an article for ACM Trans. on Embedded Computing Systems (TECS).

### *9.1.4. Research Administration*

- Pascal Fradet is head of the committee for doctoral studies ("Responsable du comité des études doctorales") of the Inria Grenoble – Rhône-Alpes research center and local correspondent for the young researchers Inria mission ("Mission jeunes chercheurs").
- Alain Girault is vice-chair of the Inria Evaluation Committee.
- Xavier Nicollin is member of the committee for computing resources users ("Comité des Utilisateurs des Moyens Informatiques") of the Inria Grenoble – Rhône-Alpes research center.
- Jean-Bernard Stefani is head of science (délégué scientifique) of the Inria Grenoble – Rhône-Alpes research center and a member of the Inria Evaluation Committee.

## 9.2. Teaching - Supervision - Juries

### *9.2.1. Teaching*

Licence : Pascal Fradet, Théorie des Langages 1 & 2, 36 HeqTD, niveau L3, Grenoble INP (Ensimag), France

Licence : Pascal Fradet, Modèles de Calcul : $\lambda$-calcul, 12 HeqTD, niveau L3, Univ. Grenoble Alpes, France

Master : Pascal Fradet, Langages et Traducteurs, 16 HeqTD, niveau M1, Polytech Grenoble, Univ. Grenoble Alpes, France

Master : Xavier Nicollin, Sémantique et Analyse des Programmes, 45 HeqTD, niveau M1, Grenoble INP (Ensimag), France

Licence : Xavier Nicollin, Théorie des Langages 2, 36 HeqTD, niveau L3, Grenoble INP (Ensimag), France

Licence : Xavier Nicollin, Bases de la Programmation Impérative, 81 HeqTD (2017-2018), niveau L3, Grenoble INP (Ensimag), France

Licence : Sophie Quinton, Théorie des Langages 2, 18 HeqTD, niveau L3, Grenoble INP (Ensimag), France

Master : Sophie Quinton, Performance and Quantitative Properties, 6h, MOSIG, Univ. Grenoble Alpes, France

Master: Jean-Bernard Stefani, Formal Aspects of Component Software, 9h, MOSIG, Univ. Grenoble Alpes, France.

### *9.2.2. Supervision*

- PhD in progress: Sihem Cherrared, "Fault Management in Multi-Tenant Programmable Networks", Univ. Rennes 1, since October 2016, co-advised by Eric Fabre and Gregor Gössler.
- PhD in progress: Christophe Prévot, "Early Performance assessment for evolving and variable Cyber-Physical Systems", Univ. Grenoble Alpes, since November 2015, co-advised by Alain Girault and Sophie Quinton.
- PhD in progress: Stephan Plassart, "On-line optimization in dynamic real-time systems", Univ. Grenoble Alpes, since September 2016, co-advised by Bruno Gaujal and Alain Girault.
- PhD in progress: Xiaojie Guo, "Formal Proofs for the Analysis of Real-Time Systems in Coq", Univ. Grenoble Alpes, since December 2016, co-advised by Pascal Fradet, Jean-François Monin, and Sophie Quinton.
- PhD in progress: Maxime Lesourd, "Generic Proofs for the Analysis of Real-Time Systems in Coq", Univ. Grenoble Alpes, since September 2017, co-advised by Pascal Fradet, Jean-François Monin, and Sophie Quinton.
- PhD in progress: Arash Shafiei, "Programming IoT and sofware defined radio with dynamic dataflow models of computation", Univ. Grenoble Alpes, since September 2017, co-advised by Pascal Fradet, Alain Girault, and Xavier Nicollin.
- PhD in progress: Martin Vassor, "Analysis and types for safe dynamic software reconfigurations", Univ. Grenoble Alpes, since November 2017, co-advised by Pascal Fradet and Jean-Bernard Stefani.
- M2 SIF in progress: T. Mari, "From diagnosis to causal analysis", U. Rennes, since November 2018, co-supervised by Gregor Gössler and Louise Travé-Massuyès (Laas).
- PFE: Clément Arvis, "Génération automatique de musique", Grenoble INP/Ensimag, September 2018, supervized by Sophie Quinton.

### *9.2.3. Juries*

- Alain Girault was referee for the PhD thesis of Colin Vidal, Université Côte d'Azur, and for the PhD thesis of Julien Hascoet, INSA Rennes. He was also vice-president of the Inria Senior Researcher jury (DR2) and of the Inria Junior Researcher national jury (CRCN).
- Gregor Gössler was examiner for the PhD jury of Vincent Wang (U. Pennsylvania).
- Jean-Bernard Stefani was examiner for the Habilitation (HDR) jury of Thomas Ledoux (U. Nantes).
- Sophie Quinton was member of the CRCN jury in Rennes.

## 9.3. Popularization

### *9.3.1. Interventions*

Sophie Quinton gave a keynote at the MathC2+ event organized by Inria, entitled "Faire des preuves par ordinateur : Pourquoi et comment ?" (Computer-assisted proofs: Why and how?).

# 10. Bibliography

## Major publications by the team in recent years

[1] S. ANDALAM, P. ROOP, A. GIRAULT, C. TRAULSEN. *A Predictable Framework for Safety-Critical Embedded Systems*, in "IEEE Trans. on Computers", July 2014, vol. 63, n° 7, pp. 1600–1612

[2] A. BOUAKAZ, P. FRADET, A. GIRAULT. *A Survey of Parametric Dataflow Models of Computation*, in "ACM Trans. Design Autom. Electr. Syst.", 2017, vol. 22, n° 2, pp. 38:1–38:25, https://doi.org/10.1145/2999539

[3] S. DJOKO DJOKO, R. DOUENCE, P. FRADET. *Aspects preserving properties*, in "Science of Computer Programming", 2012, vol. 77, n° 3, pp. 393-422

[4] G. FREHSE, A. HAMANN, S. QUINTON, M. WÖHRLE. *Formal Analysis of Timing Effects on Closed-loop Properties of Control Software*, in "35th IEEE Real-Time Systems Symposium 2014 (RTSS)", Rome, Italy, December 2014, https://hal.inria.fr/hal-01097622

[5] A. GIRARD, G. GÖSSLER, S. MOUELHI. *Safety Controller Synthesis for Incrementally Stable Switched Systems Using Multiscale Symbolic Models*, in "IEEE Transactions on Automatic Control", 2016, vol. 61, n° 6, pp. 1537-1549 [*DOI :* 10.1109/TAC.2015.2478131], https://hal.archives-ouvertes.fr/hal-01197426

[6] G. GÖSSLER, D. LE MÉTAYER. *A general framework for blaming in component-based systems*, in "Science of Computer Programming", 2015, vol. 113, Part 3 [*DOI :* 10.1016/J.SCICO.2015.06.010], https://hal.inria.fr/hal-01211484

[7] I. LANESE, C. A. MEZZINA, J.-B. STEFANI. *Reversibility in the higher-order $\pi$-calculus*, in "Theoretical Computer Science", 2016, vol. 625, pp. 25-84 [*DOI :* 10.1016/J.TCS.2016.02.019], https://hal.inria.fr/hal-01303090

[8] S. QUINTON, M. HANKE, R. ERNST. *Formal analysis of sporadic overload in real-time systems*, in "2012 Design, Automation & Test in Europe Conference & Exhibition, DATE 2012, Dresden, Germany, March, 2012", 2012, pp. 515–520, http://dx.doi.org/10.1109/DATE.2012.6176523

# Publications of the year

## Articles in International Peer-Reviewed Journals

[9] R. ERNST, S. KUNTZ, S. QUINTON, M. SIMONS. *The Logical Execution Time Paradigm: New Perspectives for Multicore Systems (Dagstuhl Seminar 18092)*, in "Dagstuhl Reports", 2018, vol. 8, pp. 122 - 149 [*DOI :* 10.4230/DAGREP.8.2.122], https://hal.inria.fr/hal-01956964

[10] A. GIRAULT, C. PRÉVOT, S. QUINTON, R. HENIA, N. SORDON. *Improving and Estimating the Precision of Bounds on the Worst-Case Latency of Task Chains*, in "IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems", August 2018, vol. 37, n° 11, pp. 2578-2589 [*DOI :* 10.1109/TCAD.2018.2861016], https://hal.inria.fr/hal-01956931

## Invited Conferences

[11] S. GRAF, S. QUINTON, A. GIRAULT, G. GÖSSLER. *Building Correct Cyber-Physical Systems: Why we need a Multiview Contract Theory*, in "FMICS 2018 - 23rd International Conference on Formal Methods for Industrial Critical Systems", Dublin, Ireland, LNCS, Springer, September 2018, vol. 11119, pp. 19-31 [*DOI :* 10.1007/978-3-030-00244-2_2], https://hal.inria.fr/hal-01891146

[12] S. QUINTON. *Evaluation and Comparison of Real-Time Systems Analysis Methods and Tools*, in "FMICS 2018 - 23rd International Conference on Formal Methods for Industrial Critical Systems", Maynooth, Ireland, LNCS, Springer, September 2018, vol. 11119, pp. 284-290 [*DOI :* 10.1007/978-3-030-00244-2_19], https://hal.inria.fr/hal-01903730

## International Conferences with Proceedings

[13] L. AHRENDTS, S. QUINTON, T. BOROSKE, R. ERNST. *Verifying Weakly-Hard Real-Time Properties of Traffic Streams in Switched Networks*, in "ECRTS 2018 - 30th Euromicro Conference on Real-Time Systems", Barcelona, Spain, July 2018, pp. 1-22 [*DOI :* 10.4230/LIPICS.ECRTS.2018.15], https://hal.inria.fr/hal-01903759

[14] S. CHERRARED, S. IMADALI, E. FABRE, G. GÖSSLER. *LUMEN: A Global Fault Management Framework For Network Virtualization Environments*, in "ICIN 2018 - 21st Conference on Innovation in Clouds, Internet and Networks and Workshops", Paris, France, IEEE, February 2018, pp. 1-8 [*DOI :* 10.1109/ICIN.2018.8401622], https://hal.inria.fr/hal-01851610

[15] P. FRADET, A. GIRAULT, R. KRISHNASWAMY, X. NICOLLIN, A. SHAFIEI. *RDF: Reconfigurable Dataflow*, in "2019 Design, Automation & Test in Europe Conference & Exhibition, DATE 2019", Florence, Italy, March 2019, https://hal.inria.fr/hal-01960788

[16] P. FRADET, X. GUO, J.-F. MONIN, S. QUINTON. *A Generalized Digraph Model for Expressing Dependencies*, in "RTNS '18 - 26th International Conference on Real-Time Networks and Systems", Chasseneuil-du-Poitou, France, October 2018, pp. 1-11 [*DOI :* 10.1145/3273905.3273918], https://hal.inria.fr/hal-01878100

[17] P. FRADET, M. LESOURD, J.-F. MONIN, S. QUINTON. *A Generic Coq Proof of Typical Worst-Case Analysis*, in "RTSS 2018 - 39th IEEE Real-Time Systems Symposium", Nashville, United States, December 2018, pp. 1-12, https://hal.inria.fr/hal-01903752

[18] A. GIRAULT, G. GÖSSLER, R. GUERRAOUI, J. HAMZA, D.-A. SEREDINSCHI. *Monotonic Prefix Consistency in Distributed Systems*, in "FORTE 2018 - 38th International Conference on Formal Techniques for Distributed Objects, Components, and Systems", Madrid, Spain, C. BAIER, L. CAIRES (editors), Formal Techniques for Distributed Objects, Components, and Systems, Springer International Publishing, June 2018, vol. LNCS-10854, pp. 41-57 [*DOI :* 10.1007/978-3-319-92612-4_3], https://hal.inria.fr/hal-01824817

[19] M. VASSOR, J.-B. STEFANI. *Checkpoint/rollback vs causally-consistent reversibility*, in "RC 2018 - 10th International Conference on Reversible Computation", Leicester, United Kingdom, Lecture Notes in Computer Science, Springer, September 2018, vol. 11106, pp. 286-303 [*DOI :* 10.1007/978-3-319-99498-7_20], https://hal.inria.fr/hal-01953756

### Research Reports

[20] A. ABDI, A. GIRAULT, H. ZARANDI. *ERPOT: A quad-criteria scheduling heuristic to optimize the execution time, failure rate, power consumption and temperature in multicores*, Inria ; 38, July 2018, n° RR-9196, pp. 1-38, https://hal.inria.fr/hal-01848087

[21] R. ERNST, L. AHRENDTS, K.-B. GEMLAU, S. QUINTON, H. VON HASSELN, J. HENNIG. *System Level LET with Application to Automotive Design*, TU Braunschweig, 2018, pp. 1-11, https://hal.inria.fr/hal-01962330

## References in notes

[22] *Automotive Open System Architecture*, 2003, http://www.autosar.org

[23] *A Library for formally proven schedulability analysis*, http://prosa.mpi-sws.org/

[24] ARTEMIS JOINT UNDERTAKING. *ARTEMIS Strategic Research Agenda*, 2011

[25] S. ANDALAM, P. ROOP, A. GIRAULT. *Predictable Multithreading of Embedded Applications Using PRET-C*, in "International Conference on Formal Methods and Models for Codesign, MEMOCODE'10", Grenoble, France, IEEE, July 2010, pp. 159–168

[26] I. ASSAYAD, A. GIRAULT. *Adaptive Mapping for Multiple Applications on Parallel Architectures*, in "Third International Symposium on Ubiquitous Networking, UNET'17", Casablanca, Morocco, May 2017, https://hal.inria.fr/hal-01672463

[27] P. AXER, R. ERNST, H. FALK, A. GIRAULT, D. GRUND, N. GUAN, B. JONSSON, P. MARWEDEL, J. REINEKE, C. ROCHANGE, M. SEBATIAN, R. VON HANXLEDEN, R. WILHELM, W. YI. *Building Timing Predictable Embedded Systems*, in "ACM Trans. Embedd. Comput. Syst.", 2014, To appear

[28] E. BAINOMUGISHA, A. CARRETON, T. VAN CUTSEM, S. MOSTINCKX, W. DE MEUTER. *A Survey on Reactive Programming*, in "ACM Computing Surveys", 2013, vol. 45, n° 4

[29] N. BANSAL, T. KIMBREL, K. PRUHS. *Speed Scaling to Manage Energy and Temperature*, in "Journal of the ACM", 2007, vol. 54, n° 1

[30] A. BASU, S. BENSALEM, M. BOZGA, J. COMBAZ, M. JABER, T.-H. NGUYEN, J. SIFAKIS. *Rigorous Component-Based System Design Using the BIP Framework*, in "IEEE Software", 2011, vol. 28, n° 3

[31] V. BEBELIS, P. FRADET, A. GIRAULT, B. LAVIGUEUR. *BPDF: A Statically Analyzable Dataflow Model with Integer and Boolean Parameters*, in "International Conference on Embedded Software, EMSOFT'13", Montreal, Canada, ACM, September 2013

[32] A. BENVENISTE, P. CASPI, S. A. EDWARDS, N. HALBWACHS, P. LE GUERNIC, R. DE SIMONE. *The synchronous languages 12 years later*, in "Proceedings of the IEEE", 2003, vol. 91, n$^{o}$ 1

[33] S. BORKAR. *Designing Reliable Systems from Unreliable Components: The Challenges of Transistor Variability and Degradation*, in "IEEE Micro", 2005, vol. 25, n$^{o}$ 6

[34] A. BOUAKAZ, P. FRADET, A. GIRAULT. *A Survey of Parametric Dataflow Models of Computation*, in "ACM Transactions on Design Automation of Electronic Systems (TODAES)", January 2017, https://hal.inria.fr/hal-01417126

[35] A. BOUAKAZ, P. FRADET, A. GIRAULT. *Symbolic Analyses of Dataflow Graphs*, in "ACM Transactions on Design Automation of Electronic Systems (TODAES)", January 2017, https://hal.inria.fr/hal-01417146

[36] R. DAVIS, A. BURNS. *A Survey of Hard Real-Time Scheduling for Multiprocessor Systems*, in "ACM Computing Surveys", 2011, vol. 43, n$^{o}$ 4

[37] S. A. EDWARDS, E. A. LEE. *The Case for the Precision Timed (PRET) Machine*, in "44th Design Automation Conference (DAC)", IEEE, 2007

[38] J. EKER, J. W. JANNECK, E. A. LEE, J. LIU, X. LIU, J. LUDVIG, S. NEUENDORFFER, S. SACHS, Y. XIONG. *Taming heterogeneity - the Ptolemy approach*, in "Proceedings of the IEEE", 2003, vol. 91, n$^{o}$ 1

[39] P. FRADET, A. GIRAULT, L. JAMSHIDIAN, X. NICOLLIN, A. SHAFIEI. *Lossy channels in a dataflow model of computation*, in "Principles of Modeling, Festschrift in Honor of Edward A. Lee", Berkeley, United States, Lecture Notes in Computer Science, Springer, October 2017, https://hal.inria.fr/hal-01666568

[40] P. FRADET, A. GIRAULT, P. POLPAVKO. *SPDF: A schedulable parametric data-flow MoC*, in "Design, Automation and Test in Europe, DATE'12", IEEE, 2012

[41] B. GAUJAL, A. GIRAULT, S. PLASSART. *Dynamic Speed Scaling Minimizing Expected Energy Consumption for Real-Time Tasks*, UGA - Université Grenoble Alpes ; Inria Grenoble Rhône-Alpes ; Université de Grenoble, October 2017, n$^{o}$ RR-9101, pp. 1-35, https://hal.inria.fr/hal-01615835

[42] A. GIRARD, G. PAPPAS. *Approximation metrics for discrete and continuous systems*, in "IEEE Trans. on Automatic Control", 2007, vol. 52, n$^{o}$ 5, pp. 782–798

[43] D. GIZOPOULOS, M. PSARAKIS, S. V. ADVE, P. RAMACHANDRAN, S. K. S. HARI, D. SORIN, A. MEIXNER, A. BISWAS, X. VERA. *Architectures for Online Error Detection and Recovery in Multicore Processors*, in "Design Automation and Test in Europe (DATE)", 2011

[44] R. GUERRAOUI, M. PAVLOVIC, D.-A. SEREDINSCHI. *Trade-offs in replicated systems*, in "IEEE Data Engineering Bulletin", 2016, vol. 39, pp. 14–26

[45] F. C. GÄRTNER. *Fundamentals of Fault-Tolerant Distributed Computing in Asynchronous Environments*, in "ACM Computing Surveys", 1999, vol. 31, n° 1

[46] G. GÖSSLER, J.-B. STEFANI. *Fault Ascription in Concurrent Systems*, in "Proc. Trustworthy Global Computing - 10th International Symposium, TGC 2015", P. GANTY, M. LORETI (editors), LNCS, Springer, 2016, vol. 9533

[47] S. HAAR, E. FABRE. *Diagnosis with Petri Net Unfoldings*, in "Control of Discrete-Event Systems", Lecture Notes in Control and Information Sciences, Springer, 2013, vol. 433, chap. 15

[48] J. HALPERN, J. PEARL. *Causes and Explanations: A Structural-Model Approach. Part I: Causes*, in "British Journal for the Philosophy of Science", 2005, vol. 56, n° 4, pp. 843-887

[49] T. HENZINGER, J. SIFAKIS. *The Embedded Systems Design Challenge*, in "Formal Methods 2006", Lecture Notes in Computer Science, Springer, 2006, vol. 4085

[50] C. M. KIRSCH, A. SOKOLOVA. *The Logical Execution Time Paradigm*, in "Advances in Real-Time Systems (to Georg Färber on the occasion of his appointment as Professor Emeritus at TU München after leading the Lehrstuhl für Realzeit-Computersysteme for 34 illustrious years)", 2012, pp. 103–120

[51] R. KÜSTERS, T. TRUDERUNG, A. VOGT. *Accountability: definition and relationship to verifiability*, in "ACM Conference on Computer and Communications Security", 2010, pp. 526-535

[52] I. LANESE, C. A. MEZZINA, J.-B. STEFANI. *Reversing Higher-Order Pi*, in "21th International Conference on Concurrency Theory (CONCUR)", Lecture Notes in Computer Science, Springer, 2010, vol. 6269

[53] P. MENZIES. *Counterfactual Theories of Causation*, in "Stanford Encyclopedia of Philosophy", E. ZALTA (editor), Stanford University, 2009, http://plato.stanford.edu/entries/causation-counterfactual

[54] M. MOORE. *Causation and Responsibility*, Oxford, 1999

[55] J. PEARL. *Causal inference in statistics: An overview*, in "Statistics Surveys", 2009, vol. 3, pp. 96-146

[56] P. RAMADGE, W. WONHAM. *Supervisory Control of a Class of Discrete Event Processes*, in "SIAM Journal on control and optimization", January 1987, vol. 25, n° 1, pp. 206–230

[57] J. RUSHBY. *Partitioning for Safety and Security: Requirements, Mechanisms, and Assurance*, NASA Langley Research Center, 1999, n° CR-1999-209347

[58] J.-B. STEFANI. *Components as Location Graphs*, in "11th International Symposium on Formal Aspects of Component Software", Bertinoro, Italy, Lecture Notes in Computer Science, September 2014, vol. 8997, https://hal.inria.fr/hal-01094208

[59] M. STIGGE, P. EKBERG, N. GUAN, W. YI. *The Digraph Real-Time Task Model*, in "17th IEEE Real-Time and Embedded Technology and Applications Symposium, RTAS 2011, Chicago, Illinois, USA, 11-14 April 2011", 2011, pp. 71–80, https://doi.org/10.1109/RTAS.2011.15

[60] P. TABUADA. *Verification and Control of Hybrid Systems - A Symbolic Approach*, Springer, 2009

[61] D. TERRY. *Replicated Data Consistency Explained Through Baseball*, Microsoft Research, October 2011, n$^{o}$ MSR-TR-2011-137

[62] K. TINDELL. *Using offset information to analyse static priority pre-emptively scheduled task sets*, Technical report YCS 182, University of York, Department of Computer Science, 1992, https://books.google.fr/books?id=qARQHAAACAAJ

[63] J. WANG, P. S. ROOP, A. GIRAULT. *Energy and timing aware synchronous programming*, in "International Conference on Embedded Software, EMSOFT'16", Pittsburgh, United States, ACM, October 2016, 10 p. [*DOI :* 10.1145/2968478.2968500], https://hal.inria.fr/hal-01412100

[64] R. WILHELM, J. ENGBLOM, A. ERMEDAHL, N. HOLSTI, S. THESING, D. B. WHALLEY, G. BERNAT, C. FERDINAND, R. HECKMANN, T. MITRA, F. MUELLER, I. PUAUT, P. P. PUSCHNER, J. STASCHULAT, P. STENSTRÖM. *The Determination of Worst-Case Execution Times — Overview of the Methods and Survey of Tools*, in "ACM Trans. Embedd. Comput. Syst.", April 2008, vol. 7, n$^{o}$ 3

[65] F. YAO, A. DEMERS, S. SHENKER. *A scheduling model for reduced CPU energy*, in "Proceedings of lEEE Annual Foundations of Computer Science", 1995, pp. 374–382