RESEARCH CENTRE

Grenoble - Rhône-Alpes

IN PARTNERSHIP WITH:

Institut polytechnique de Grenoble

2020 ACTIVITY REPORT

Project-Team SPADES

Sound Programming of Adaptive Dependable Embedded Systems

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

DOMAIN

Algorithmics, Programming, Software and Architecture

THEME

Embedded and Real-time Systems

Contents

Project-Team SPADES

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

Keywords

Computer sciences and digital sciences

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

Other research topics and application domains

B6.3.3. – Network Management

B6.4. – Internet of things

B6.6. – Embedded systems

1 Team members, visitors, external collaborators

Research Scientists

- Gregor Goessler [Team leader, Inria, Senior Researcher, HDR]
- Martin Bodin [Inria, Researcher, from Oct 2020]
- Pascal Fradet [Inria, Researcher, HDR]
- Alain Girault [Inria, Senior Researcher, HDR]
- Sophie Quinton [Inria, Researcher]
- Jean-Bernard Stefani [Inria, Senior Researcher]
- Paolo Torrini [Inria, Advanced Research Position, from Jul 2020]

Faculty Member

- Xavier Nicollin [Institut polytechnique de Grenoble, Associate Professor]

Post-Doctoral Fellow

- Jia Jie Wang [Inria, until May 2020]

PhD Students

- Xiaojie Guo [Inria, until April 2020]
- Maxime Lesourd [Univ Grenoble Alpes]
- Thomas Mari [Institut polytechnique de Grenoble]
- Stephan Plassart [Inria, until Jun 2020]
- Aina Rasoldier [Inria, from Oct 2020]
- Arash Shafiei [Orange Labs, until Sep 2020]
- Martin Vassor [Inria]

Technical Staff

- Roger Pissard-Gibollet [Inria]

Interns and Apprentices

- Lucca Hoogenbosch [Ministère de l'Education Nationale, Intern, Feb 2020]
- Aina Rasoldier [Inria, Intern, from Feb 2020 until Jul 2020]
- Clement Stefani [Ministère de l'Education Nationale, Intern, Feb 2020]

Administrative Assistants

- Julia Di Toro [Inria, from Oct 2020]
- Helen Pouchot-Rouge-Blanc [Inria, until Sep 2020]

External Collaborator

- Aina Rasoldier [INSA Lyon, from Jul 2020 until Sep 2020]

2 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 distributed embedded systems as dynamic adaptive modular structures?
2. How to program reactive systems with real-time and resource constraints?
3. How to program fault-tolerant and explainable embedded systems?

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 [23], 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

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. 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.

3.1 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 [42]. Witness component-based toolsets such as PTOLEMY [33], BIP [27], or the modular architecture frameworks used, for instance, in the automotive industry (AUTOSAR) [24]. 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.

3.2 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 [28]), 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 [25, 32], 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. [25]. For our part, we intend to focus on devising synchronous programming languages for distributed systems and precision-timed architectures.

3.3 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 [29, 39]. 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 [47], natural sciences, law [48], and statistics [51], 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 [45], to allow the diagnosis of faults in a complex concurrent system [41], or to enforce accountability [44], 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 [37]), or is broken (*e.g.*, by limiting fault propagation [52]).

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 Current Industrial Cooperations

Regarding applications and case studies with industrial end-users of our techniques, we cooperate with Orange Labs on software architecture for cloud services.

5 Social and environmental responsibility

5.1 Footprint of research activities

This is the first time the team has considered assessing the footprint of its activities. However, 2020 has been quite an unusual year (!) in terms of research activities: almost no travel, remote working... In addition, there are currently no Inria guidelines regarding how GHG emissions (or more generally environmental impacts) should be estimated and attributed (e.g. heating of buildings, eating at work). As a result, we are not able at this point to provide a representative estimate of the footprint of our research activities. That said, we support the idea of providing such an estimate in the future and contribute here a few suggestions and questions.

Based on similar assessments in other institutions, one can expect the footprint to be mainly due to the following activities:

1. travels (for conferences, project meetings etc.)
2. daily commutes (to and from Inria as well as to and from the campus)
3. purchase of computers, screens, etc. (how should we account for this?)
4. heating, and more generally building costs
5. all supporting activities (digital infrastructures such as web servers, mail servers, data storage... but also footprint of the activities of nonscientific staff)

Note that SPADES' research activities do not require data intensive computations.

We argue that the last two points should be assessed at the research center level. Besides, we would appreciate supporting tools for the (semi-)automatic extraction and analysis of data for travels, as well as specific guidelines for daily commutes and purchases.

5.2 Impact of research results

Research into the connection between ICT (Information and Communication Technologies) and the environmental crisis has started in 2020 within the SPADES team. Sophie Quinton co-supervised the internship of Aina Rasoldier on the assessment of environmental impacts of ICT. Aina is now a PhD candidate in the team, working on the potential and risks of local digital solutions for tackling the environmental crisis. This is the first visible action of a larger move of the team to establish new research directions in the broad topic of environmental impacts.

6 New software and platforms

6.1 New software

6.1.1 pyCPA_TWCA

Name: 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

6.1.2 CertiCAN

Name: 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 light-weight 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).

Contacts: Pascal Fradet, Xiaojie Guo, Jean-François Monin, Sophie Quinton

7 New results

7.1 Design and Programming Models

Participants Pascal Fradet, Alain Girault, Xavier Nicollin, Arash Shafiei, Jean-Bernard Stefani, Martin Vassor.

7.1.1 Hypercells

The Hypercell framework, presented in [53], allows the definition of different component models for dynamic software architectures featuring both sharing and encapsulation. Its behavioral theory is still in its initial stages but features the definition of a form of contextual bisimilarity. This year has seen the further development of the framework with new results on the modeling of encapsulation policies and their characterization by means of contextual bisimilarity, and the development of a first implentation of the framework as a Rust programming language library.

In collaboration with the Spirals team at Inria Lille – Nord Europe, and Orange, we have used hypercells as a pivot model for developing interpretations, formally defined with the Alloy specification language, of various languages and formalisms for the description of software configurations for cloud computing environments. Configuration languages considered include the TOSCA and OCCI standards, as well as the Open Stack Heat Orchestration Template (HOT), Docker Compose, and the Aeolus component model for cloud deployment. This work, developed as part of a bilateral contract with Orange, allowed the development of a verification tool for the correctness of HOT configurations, helped uncover several flaws in the HOT specifications, and in the ETSI NFV standard. The work is reported in [20].

7.1.2 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) [34, 50], we have written a survey providing a comprehensive description of the existing parametric dataflow MoCs [30], and we have studied *symbolic* analyses of dataflow graphs [31]. More recently, we have proposed an original method to deal with lossy communication channels in dataflow graphs [35].

We are nowadays studying models allowing *dynamic reconfigurations* of the *topology* of the dataflow graphs. This is required by many modern streaming applications that 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) [36]. 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. Transformations can be seen as graph rewriting rules that match some

sub-part of the dataflow graph and replace it by another one. Transformations can be applied an arbitrary number of times during execution and therefore can produce an arbitrary number of new graphs. The major feature and advantage of RDF is that it can be *statically analyzed* to guarantee that all possible graphs generated at runtime will be connected, consistent, and live. To the best of our knowledge, RDF is the only dataflow MoC allowing an arbitrary number of topological reconfigurations while remaining statically analyzable. The RDF MoC has been implemented by Arash Shafiei within a software tool that allows the designer to write an initial RDF graph and its transformation rules. The static analyses for connectivity, consistency, and liveness have been implemented too. And a canny edge detector case study shows that dynamic reconfigurations to increase the parallelism level, when the incoming video stream becomes more computationally intensive, can be performed seamlessly. Finally, we have proposed in 2020 a new latency analysis for RDF that allows us to bound the latendy variation incurred by applying a given transformation rule whatever the RDF graph it is applied to.

This is the research topic of Arash Shafiei's PhD, in collaboration with Orange Labs.

7.2 Certified Real-Time Programming

Participants Pascal Fradet, Alain Girault, Xavier Nicollin, Sophie Quinton, Xiaojie Guo, Maxime Lesourd.

7.2.1 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 minimize the energy consumption while guaranteeing that each job meets its deadline. The idea is to leverage on the *statistical information* on the jobs' characteristics available at design time: release time, worst-case execution time (WCET), and relative deadline. This is the topic of Stephan Plassart's PhD [18], funded by the CASERM Persyval project. We have considered several cases depending on the amount of information available at design time:

Offline case: In the offline case, all the information is known and we have proposed the first linear complexity offline scheduling algorithm that minimizes the total energy consumption [9, 19]: our complexity is $\mathcal{O}(n)$, where n is the number of jobs to be scheduled, while the previously best known algorithms were in $\mathcal{O}(n^2)$ and $\mathcal{O}(n \log n)$ [46].

Clairvoyant case: In the clairvoyant case, the characteristics of the jobs are only known statistically, and each job's WCET and relative deadline are only known at release time. We want to compute the *optimal* online scheduling speed policy that minimizes the *expected* energy consumption while guaranteeing that each job meets its deadline. 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 the finite horizon case we use a dynamic programming algorithm, while in the infinite horizon case we use a value iteration algorithm [10].

Non-clairvoyant case: In the non-clairvoyant case, the actual execution time (AET) of a job is only known only when this job completes its execution. This AET is of course assumed to be less than the WCET, which is known at the job's release time. Again, by building an MDP for the system with a well chosen state, we compute the *optimal* online scheduling speed policy that minimizes the *expected* energy consumption [16].

Learning case: In the learning case, the only information known for the jobs are a bound on the jobs' WCETs and a bound on their deadlines. We have proposed two *reinforcement learning* algorithms, one that learns the optimal value of the expected energy (Q-learning), and another one that learns the probability transition matrix of the system, from which we derive the optimal online speed policy.

This work led us to compare several existing speed policies with respect to their feasibility. Indeed, the policies (OA) [54], (AVR) [54], and (BKP) [26] 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 [11].

7.2.2 Formal proofs for schedulability analysis of real-time systems

We contribute to Prosa [21], 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.

We advanced our work on formal proofs for schedulability analysis of real-time systems mainly in two directions in 2020. First, several optimizations have increased dramatically the timing efficiency of our CertiCAN tool, the first formally proven tool able to certify the results of commercial CAN analysis tools. This work is included in the PhD thesis of Xiaojie Guo (not yet published), who defended in December 2020. Second, Paolo Torrini, who was recruited in July 2020, has opened a new line of work on correct by construction implementation of schedulers.

7.2.3 Scheduling under multiple constraints and Pareto optimization

We have considered the bi-criteria minimization problem in the (worst-case execution time – WCET, worst-case energy consumption – WCEC) space for real-time programs. To the best of our knowledge, this is the first contribution of this kind in the literature.

A real-time program is abstracted as a Timed Control Flow Graph (TCFG), where each basic block is labeled with the number of clock cycles required to execute it on the chosen processor at the nominal frequency. This timing information can be obtained, for instance, with a WCET analysis tool. The target processor is equipped with dynamic voltage and frequency scaling (DVFS) and offers several (frequency f, voltage V) operating points, as is the case with most processors today. The goal is to compute a set of assignments from the set of basic blocks of the TCFG to the set of available (f, V) pairs, such that each such assignment is a *non-dominated* points in the (WCET, WCEC) plane, non-dominated in the *Pareto* sense.

From the TCFG we extract the longest execution path, and then we compute the WCET and the WCEC for this path at a chosen (f, V) pair. By construction, all the other execution paths are shorter, so this WCET and this WCEC hold for the whole program. This ensures that each single-frequency assignment is a non-dominated point, and therefore belongs to the Pareto front. Then, we have studied two frequencies assignments, still for the longest execution path. When the frequency switching costs in time and in energy are assumed to be negligible, we have proved that each two frequencies (say with f_i and f_k) assignment is a point located in the segment between the single frequency assignment at f_i and the single frequency assignment at f_k. We have also proposed a linear time heuristic to assign a (f, V) pair to all the other blocks (*i.e.*, those not belonging to the longest path) such that all the other execution paths of the TCFG have a shorter WCET and a lesser WCEC. We have also established conditions under which the resulting assignment corresponds to a non-dominated point. Finally, we have generalized these results to the case where the frequency switching costs are not negligible. Surprisingly, this case reduces the size of the search space from exponential (m^n, where n is the number of blocks and m is the number of frequencies) to polynomial because all the assignments involving more than three different frequencies will be dominated by an assignment involing either one or two frequencies. This key result allows us to generate the *exact* Pareto front. This was the topic of Jia Jie Wang's postdoc.

All our algorithms have been evaluated on a set of hard real time benchmark programs. This shows that they perform extremely well. Our DVFS assignment algorithm can also be used as a back-end for the compiler of the PRET-C programming language [49, 22, 8] in order to make it energy aware, thanks to the ability of this compiler to generate TCFGs. Future work will invlove developping similar algorithms but in the much more difficult case of the parallel programming language ForeC [38].

7.3 Fault Management and Causal Analysis

Participants Gregor Goessler, Jean-Bernard Stefani, Sihem Cherrared, Thomas Mari.

7.3.1 Fault Ascription in Concurrent Systems

Fault ascription is a precise form of fault diagnosis that relies on counterfactual analysis for pinpointing the causes of system failures. Research on counterfactual causality has been marked, until today, by a succession of definitions of causation that are informally validated against human intuition on mostly simple examples. This approach suffers from its dependence on the tiny number and incompleteness of examples in the literature, and from the lack of objective correctness criteria [40].

We have introduced in [14] a general framework for fault ascription, which consists in identifying, in a concurrent system, the events or components whose faulty behavior has caused the failure of said system. Our framework uses configuration structures as a general semantical model to handle truly concurrent executions, partial and distributed observations in a uniform way. In contrast with most of the current literature on counterfactual analysis which relies heavily on a set of toy examples, we have defined a set of expected formal properties for counterfactual builders, i.e. operators that build counterfactual executions. We have then shown that causality analyses that satisfy our requirements meet a set of elementary soundness and completeness properties. Finally we have presented a concrete causality analysis meeting all our requirements, and we have shown it to be monotonic under two forms of refinement.

7.3.2 Causal Explanations for Embdded Systems

Model-Based Diagnosis of discrete event systems (DES) usually aims at detecting failures and isolating faulty event occurrences based on a behavioural model of the system and an observable execution log. The strength of a diagnostic process is to determine *what* happened that is consistent with the observations. In order to go a step further and explain *why* the observed outcome occurred, we borrow techniques from causal analysis. We are currently exploring techniques that are able to extract, from an execution trace, the causally relevant part for a property violation. As part of the SEC project we are investigating how such techniques can be extended to classes of real-time systems.

7.3.3 Fault Management in Virtualized Networks

From a more applied point of view we have been investigating 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. In her PhD thesis [17], Sihem Cherrared has proposed a model-based root cause analysis framework for virtualized networks. In order to overcome the lack of accurate previous knowledge, the framework features a self-modeling algorithm that models the dependencies within and between layers of virtual networks, including auto-recovery and elasticity aspects. Model-based diagnosis is performed using constraint solving on the previous and acquired knowledge.

8 Bilateral contracts and grants with industry

8.1 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 was one of the two co-directors of the lab, till Feb. 2020). 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.

8.2 Grants with Industry

With Orange:

- Fault Management in Multi-Tenant Programmable Networks. This CIFRE grant funded the PhD of Sihem Cherrared.
- Dynamic dataflow models of computation. This CIFRE grant funds the PhD of Arash Shafiei.

9 Partnerships and cooperations

9.1 National initiatives

9.1.1 ANR

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 2022.

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.

The results obtained in 2020 in connection with the RT-proofs project are described in Section 7.2.2.

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.

9.1.2 Institute of Technology (IRT)

CAPHCA

Participants Alain Girault, Nicolas Hili.

CAPHCA is a project within the Antoine de Saint Exupéry IRT in Toulouse. The general objective of the project is to provide methods and tools to achieve both performance and determinism on modern, high-performance, multi-core and FPGA-enabled SOCs. Our specific contribution lies withing work packages dedicated to the design of novel PRET architectures and programming languages. This contract has yielded two publications so far [43, 38].

9.2 Regional initiatives

SEC

Participants Gregor Goessler, Thomas Mari.

SEC (*Safe and Explainable Cyber-physical systems*, 2019–22) is a joint project by SPADES and VERIMAG, funded by the "Initiative of Excellence" of Grenoble University. It funds Thomas Mari's PhD thesis.

10 Dissemination

10.1 Promoting scientific activities

10.1.1 Scientific events: organisation

General chair, scientific chair

Alain Girault belongs to the Steering Committee of the international conferences EMSOFT and DISCOTEC.

10.1.2 Scientific events: selection

Chair of conference program committees

Alain Girault was TPC co-chair of the *Forum on Specification & Design Languages (FDL'20)*.

Member of the conference program committees

Gregor Gössler was a TPC member of the *International Conference on Embedded Software (EMSOFT'20)* and *International Conference on Formal Methods and Models for System Design (MEMOCODE'20)*.

Sophie Quinton was a TPC member of the *Euromicro Conference on Real-Time Systems (ECRTS'20)* and the *Real-Time Systems Symposium (RTSS'20)*.

Reviewer

Alain Girault has reviewed articles for the international conferences EMSOFT'20 and RTSS'20.

10.1.3 Journal

Member of the editorial boards

Alain Girault is guest editor for a special issue of ACM Trans. on Embedded Computing Systems. He is associate editor for Real-Time Systems Journal and for Eurasip Journal on Embedded Systems.

Reviewer - reviewing activities

Alain Girault has reviewed several articles for ACM Trans. on Embedded Computing Systems.

Gregor Gössler has reviewed an article for the Journal of Computer Security.

10.1.4 Leadership within the scientific community

Alain Girault is a member of the EMSIG board and he manages its mailing list.

Sophie Quinton is a member of the ACM SIGBED Executive Committee and Associate Editor of the SIGBED Blog.

Sophie Quinton co-chairs a working group of the GDR CIS associated with the Center for Internet and Society (http://cis.cnrs.fr/) focused on environmental issues.

10.1.5 Scientific expertise

Gregor Gössler has reviewed a project for the French funding agency ANR.

10.1.6 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 Deputy Scientific Director in charge of the domain "Algorithmics, Programming, Software and Architecture".

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.

Sophie Quinton is in charge of organizing discussions and actions regarding the environmental and societal impact of our research at Inria Grenoble Rhône-Alpes.

Jean-Bernard Stefani is Head of Science of the Inria Grenoble Rhône-Alpes research center.

10.2 Teaching - Supervision - Juries

10.2.1 Teaching

- Licence : Pascal Fradet, Théorie des Langages 1 & 2, 36 HeqTD, niveau L3, Grenoble INP (Ensimag), France
- Licence : Pascal Fradet, Modèles de Calcul : λ-calcul, 12 HeqTD, niveau L3, Univ. Grenoble Alpes, France
- Master : Xavier Nicollin, Analyse de Code pour la Sûreté et la Sécurité, 45 HeqTD, niveau M1, Grenoble INP (Ensimag), France
- Licence : Xavier Nicollin, Théorie des Langages 1, 48 HeqTD, niveau L3. Grenoble INP (Ensimag), France

- Licence : Xavier Nicollin, Théorie des Langages 2, 37,5 HeqTD, niveau L3, Grenoble INP (Ensimag), France
- Licence : Xavier Nicollin, Bases de la Programmation Impérative, 30 HeqTD, niveau L3, Grenoble INP (Ensimag), France
- Master : Sophie Quinton, Performance and Quantitative Properties, 8 HeqTD, MOSIG, Univ. Grenoble Alpes, France
- Master: Jean-Bernard Stefani, Formal Aspects of Component Software, 9h, MOSIG, Univ. Grenoble Alpes, France.

10.2.2 Supervision

- PhD: Sihem Cherrared, "Fault Management in Multi-Tenant Programmable Networks"; Univ. Rennes 1; defended on June 26, 2020; co-advised by Eric Fabre and Gregor Gössler.
- PhD in progress: Thomas Mari, "Construction of Safe Explainable Cyber-physical systems"; Grenoble INP; since October 2019; co-advised by Gregor Gössler and Thao Dang.
- PhD defended in June 2020: Stephan Plassart, "On-line optimization in dynamic real-time systems"; Univ. Grenoble Alpes; co-advised by Bruno Gaujal and Alain Girault.
- PhD defended on December 18 2020: 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, "RDF: A reconfigurable dataflow MoC supporting dynamic topological transformations and static analyzability"; 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.

10.2.3 Juries

- Alain Girault was president of the PhD jury of Alexandre Honorat (INSA-Rennes).
- Gregor Gössler was reviewer for the PhD thesis of Valentin Bouziat (Toulouse University).

10.3 Popularization

10.3.1 Education

Sophie Quinton is part of the scientific committee of the upcoming "COP2 étudiante" (https://cop2etudiante.org/).

10.3.2 Interventions

Sophie Quinton was invited to the panel of Forum 5i on the topic: "Innovation and carbon footprint: which synergies ?".

11 Scientific production

11.1 Major publications

[1] A. Bouakaz, P. Fradet and A. Girault. 'A Survey of Parametric Dataflow Models of Computation'. In: *ACM Trans. Design Autom. Electr. Syst.* 22.2 (2017), 38:1–38:25. DOI: 10.1145/2999539.

[2] S. D. Djoko, R. Douence and P. Fradet. 'Aspects preserving properties'. In: *Science of Computer Programming* 77.3 (2012), pp. 393–422.

[3] P. Fradet, X. Guo, J.-F. Monin and S. Quinton. 'CertiCAN: A Tool for the Coq Certification of CAN Analysis Results'. In: *RTAS 2019 - 25th IEEE Real-Time and Embedded Technology and Applications Symposium*. Montreal, Canada: IEEE, Apr. 2019, pp. 1–10. DOI: 10.1109/RTAS.2019.00023. URL: https://hal.archives-ouvertes.fr/hal-02119024.

[4] G. Frehse, A. Hamann, S. Quinton and 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, Dec. 2014. URL: https://hal.inria.fr/hal-01097622.

[5] A. Girard, G. Gössler and S. Mouelhi. 'Safety Controller Synthesis for Incrementally Stable Switched Systems Using Multiscale Symbolic Models'. In: *IEEE Transactions on Automatic Control* 61.6 (2016), pp. 1537–1549. DOI: 10.1109/TAC.2015.2478131. URL: https://hal.archives-ouvertes.fr/hal-01197426.

[6] G. Gössler and D. Le Métayer. 'A general framework for blaming in component-based systems'. In: *Science of Computer Programming* 113, Part 3 (2015). DOI: 10.1016/j.scico.2015.06.010. URL: https://hal.inria.fr/hal-01211484.

[7] I. Lanese, C. A. Mezzina and J.-B. Stefani. 'Reversibility in the higher-order π-calculus'. In: *Theoretical Computer Science* 625 (2016), pp. 25–84. DOI: 10.1016/j.tcs.2016.02.019. URL: https://hal.inria.fr/hal-01303090.

[8] S. Andalam, P. S. Roop, A. Girault and C. Traulsen. 'A Predictable Framework for Safety-Critical Embedded Systems'. In: *TC* 63.7 (July 2014), pp. 1600–1612.

11.2 Publications of the year

International journals

[9] B. Gaujal, A. Girault and S. Plassart. 'A Pseudo-Linear Time Algorithm for the Optimal Discrete Speed Minimizing Energy Consumption'. In: *Discrete Event Dynamic Systems* (2020). URL: https://hal.archives-ouvertes.fr/hal-03030416.

[10] B. Gaujal, A. Girault and S. Plassart. 'Dynamic Speed Scaling Minimizing Expected Energy Consumption for Real-Time Tasks'. In: *Journal of Scheduling* (4th July 2020), pp. 1–25. DOI: 10.1007/s10951-020-00660-9. URL: https://hal.inria.fr/hal-02888573.

[11] B. Gaujal, A. Girault and S. Plassart. 'Feasibility of on-line speed policies in real-time systems'. In: *Real-Time Systems* (27th Apr. 2020). DOI: 10.1007/s11241-020-09347-y. URL: https://hal.inria.fr/hal-02557148.

[12] K.-B. Gemlau, L. Köhler, R. Ernst and S. Quinton. 'System-level Logical Execution Time: Augmenting the Logical Execution Time Paradigm for Distributed Real-Time Automotive Software'. In: *ACM Transactions on Cyber-Physical Systems* 5.2 (28th Jan. 2021), pp. 1–27. DOI: 10.1145/3381847. URL: https://hal.inria.fr/hal-03125851.

[13] A. Girard and G. Gössler. 'Safety Synthesis for Incrementally Stable Switched Systems using Discretization-Free Multi-Resolution Abstractions'. In: *Acta Informatica* 57 (2020), pp. 245–269. DOI: 10.1007/s00236-019-00341-x. URL: https://hal.archives-ouvertes.fr/hal-02286661.

[14] G. Gössler and J.-B. Stefani. 'Causality analysis and fault ascription in component-based systems'. In: *Theoretical Computer Science* 837 (2020), pp. 158–180. DOI: 10.1016/j.tcs.2020.06.010. URL: https://hal.inria.fr/hal-02927216.

[15] Z. A. H. Hammadeh, S. Quinton and R. Ernst. 'Weakly-hard Real-time Guarantees for Earliest Deadline First Scheduling of Independent Tasks'. In: *ACM Transactions on Embedded Computing Systems (TECS)* 18.6 (22nd Jan. 2020), pp. 1–25. DOI: 10.1145/3356865. URL: https://hal.inria.fr/hal-02459836.

Conferences without proceedings

[16] B. Gaujal, A. Girault and S. Plassart. 'Discrete and Continuous Optimal Control for Energy Minimization in Real-Time Systems'. In: EBCCSP 2020 - 6th International Conference on Event-Based Control, Communication, and Signal Processing. Krakow, Poland, 23rd Sept. 2020, pp. 1–8. URL: https://hal.archives-ouvertes.fr/hal-03020330.

Doctoral dissertations and habilitation theses

[17] S. Cherrared. 'Fault management of programmable multi-tenant networks'. Université Rennes 1, 26th June 2020. URL: https://tel.archives-ouvertes.fr/tel-03047092.

[18] S. Plassart. 'Online optimization in dynamic real-time systems'. Université Grenoble Alpes [2020-....], 16th June 2020. URL: https://tel.archives-ouvertes.fr/tel-02990646.

Reports & preprints

[19] B. Gaujal, A. Girault and S. Plassart. *A Linear Time Algorithm Computing the Optimal Speeds Minimizing Energy Under Real-Time Constraints*. Inria Grenoble Rhône-Alpes, 10th Apr. 2020. URL: https://hal.inria.fr/hal-02540230.

[20] P. Merle, S. Ben Rayana, L. Seinturier, R. Pissard-Gibollet, J.-B. Stefani and A. Ndeye Sylla. *Towards a formal reference computational model for cloud configuration management*. INRIA, 7th Jan. 2020. URL: https://hal.inria.fr/hal-02940938.

11.3 Cited publications

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

[22] S. Andalam, P. S. Roop, A. Girault and C. Traulsen. 'A Predictable Framework for Safety-Critical Embedded Systems'. In: *IEEE Transactions on Computers* (July 2014), p. 13. URL: https://hal.inria.fr/hal-01095468.

[23] ARTEMIS Joint Undertaking. *ARTEMIS Strategic Research Agenda*. 2011.

[24] *Automotive Open System Architecture*. 2003. URL: http://www.autosar.org.

[25] E. Bainomugisha, A. Carreton, T. Van Cutsem, S. Mostinckx and W. De Meuter. 'A Survey on Reactive Programming'. In: *ACM Computing Surveys* 45.4 (2013).

[26] N. Bansal, T. Kimbrel and K. Pruhs. 'Speed Scaling to Manage Energy and Temperature'. In: *Journal of the ACM* 54.1 (2007).

[27] A. Basu, S. Bensalem, M. Bozga, J. Combaz, M. Jaber, T.-H. Nguyen and J. Sifakis. 'Rigorous Component-Based System Design Using the BIP Framework'. In: *IEEE Software* 28.3 (2011).

[28] A. Benveniste, P. Caspi, S. A. Edwards, N. Halbwachs, P. Le Guernic and R. de Simone. 'The synchronous languages 12 years later'. In: *Proceedings of the IEEE* 91.1 (2003).

[29] S. Borkar. 'Designing Reliable Systems from Unreliable Components: The Challenges of Transistor Variability and Degradation'. In: *IEEE Micro* 25.6 (2005).

[30] A. Bouakaz, P. Fradet and A. Girault. 'A Survey of Parametric Dataflow Models of Computation'. In: *ACM Transactions on Design Automation of Electronic Systems (TODAES)* (Jan. 2017). URL: https://hal.inria.fr/hal-01417126.

[31] A. Bouakaz, P. Fradet and A. Girault. 'Symbolic Analyses of Dataflow Graphs'. In: *ACM Transactions on Design Automation of Electronic Systems (TODAES)* (Jan. 2017). URL: https://hal.inria.fr/hal-01417146.

[32] R. Davis and A. Burns. 'A Survey of Hard Real-Time Scheduling for Multiprocessor Systems'. In: *ACM Computing Surveys* 43.4 (2011).

[33] J. Eker, J. W. Janneck, E. A. Lee, J. Liu, X. Liu, J. Ludvig, S. Neuendorffer, S. Sachs and Y. Xiong. 'Taming heterogeneity - the Ptolemy approach'. In: *Proceedings of the IEEE* 91.1 (2003).

[34] P. Fradet, A. Girault and P. Polpavko. 'SPDF: A schedulable parametric data-flow MoC'. In: *Design, Automation and Test in Europe, DATE'12*. IEEE, 2012.

[35] P. Fradet, A. Girault, L. Jamshidian, X. Nicollin and 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, Oct. 2017. URL: https://hal.inria.fr/hal-01666568.

[36] P. Fradet, A. Girault, R. Krishnaswamy, X. Nicollin and A. Shafiei. 'RDF: Reconfigurable Dataflow'. In: *DATE 2019 - Design, Automation & Test in Europe Conference & Exhibition*. Florence, Italy, Mar. 2019, pp. 1709–1714. DOI: 10.23919/DATE.2019.8714987. URL: https://hal.inria.fr/hal-01960788.

[37] F. C. Gärtner. 'Fundamentals of Fault-Tolerant Distributed Computing in Asynchronous Environments'. In: *ACM Computing Surveys* 31.1 (1999).

[38] A. Girault, N. Hili, É. Jenn and E. Yip. 'A Multi-Rate Precision Timed Programming Language for Multi-Cores'. In: *FDL 2019 - Forum for Specification and Design Languages*. Southampton, United Kingdom: IEEE, Sept. 2019, pp. 1–8. DOI: 10.1109/FDL.2019.8876950. URL: https://hal.inria.fr/hal-02399998.

[39] D. Gizopoulos, M. Psarakis, S. V. Adve, P. Ramachandran, S. K. S. Hari, D. Sorin, A. Meixner, A. Biswas and X. Vera. 'Architectures for Online Error Detection and Recovery in Multicore Processors'. In: *Design Automation and Test in Europe (DATE)*. 2011.

[40] C. Glymour, D. Danks, B. Glymour, F. Eberhardt, J. Ramsey, R. Scheines, P. Spirtes, C. M. Teng and J. Zhang. 'Actual causation: a stone soup essay'. In: *Synthese* 175.2 (2010), pp. 169–192.

[41] S. Haar and E. Fabre. 'Diagnosis with Petri Net Unfoldings'. In: *Control of Discrete-Event Systems*. Vol. 433. Lecture Notes in Control and Information Sciences. Springer, 2013. Chap. 15.

[42] T. Henzinger and J. Sifakis. 'The Embedded Systems Design Challenge'. In: *Formal Methods 2006*. Vol. 4085. Lecture Notes in Computer Science. Springer, 2006.

[43] N. Hili, A. Girault and E. Jenn. 'Worst-Case Reaction Time Optimization on Deterministic Multi-Core Architectures with Synchronous Languages'. In: *RTCSA2019 2019 - 25th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. Hangzhou, China: IEEE, Aug. 2019, pp. 1–11. DOI: 10.1109/RTCSA.2019.8864570. URL: https://hal.inria.fr/hal-02400009.

[44] R. Küsters, T. Truderung and A. Vogt. 'Accountability: definition and relationship to verifiability'. In: *ACM Conference on Computer and Communications Security*. 2010, pp. 526–535.

[45] I. Lanese, C. A. Mezzina and J.-B. Stefani. 'Reversing Higher-Order Pi'. In: *21th International Conference on Concurrency Theory (CONCUR)*. Vol. 6269. Lecture Notes in Computer Science. Springer, 2010.

[46] M. Li, F. Yao and H. Yuan. 'An $O(n^2)$ Algorithm for Computing Optimal Continuous Voltage Schedules'. In: *Annual Conference on Theory and Applications of Models of Computation, TAMC'17*. Vol. 10185. LNCS. Bern, Switzerland, Apr. 2017, pp. 389–400.

[47] P. Menzies. 'Counterfactual Theories of Causation'. In: *Stanford Encyclopedia of Philosophy*. Ed. by E. Zalta. Stanford University, 2009. URL: http://plato.stanford.edu/entries/causation-counterfactual.

[48] M. Moore. *Causation and Responsibility*. Oxford, 1999.

[49] S. Andalam, P. S. Roop and 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.

[50] V. Bebelis, P. Fradet, A. Girault and B. Lavigueur. 'BPDF: A Statically Analyzable Dataflow Model with Integer and Boolean Parameters'. In: *International Conference on Embedded Software, EMSOFT'13*. Montreal, Canada: ACM, Sept. 2013.

[51] J. Pearl. 'Causal inference in statistics: An overview'. In: *Statistics Surveys* 3 (2009), pp. 96–146.

[52] J. Rushby. *Partitioning for Safety and Security: Requirements, Mechanisms, and Assurance*. Tech. rep. CR-1999-209347. NASA Langley Research Center, 1999.

[53] J.-B. Stefani and M. Vassor. 'Encapsulation and Sharing in Dynamic Software Architectures: The Hypercell Framework'. In: *FORTE 2019 - 39th International Conference on Formal Techniques for Distributed Objects, Components, and Systems (FORTE)*. Ed. by J. A. Pérez and N. Yoshida. Vol. LNCS-11535. Formal Techniques for Distributed Objects, Components, and Systems. Part 1: Full Papers. Copenhagen, Denmark: Springer International Publishing, 2019, pp. 242–260. DOI: 10.1007/978-3-030-21759-4_14. URL: https://hal.inria.fr/hal-02313751.

[54] F. Yao, A. Demers and S. Shenker. 'A scheduling model for reduced CPU energy'. In: *Proceedings of IEEE Annual Foundations of Computer Science*. 1995, pp. 374–382.