RESEARCH CENTRE

Grenoble - Rhône-Alpes

IN PARTNERSHIP WITH:

Institut polytechnique de Grenoble

2021

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 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]
- 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]

Faculty Member

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

PhD Students

- Giovanni Fabbretti [Inria, from Feb 2021]
- Aurelie Kong Win Chang [Inria]
- Pietro Lami [Inria, from Oct 2021]
- Maxime Lesourd [Inria, until Aug 2021]
- Thomas Mari [Institut polytechnique de Grenoble]
- Aina Rasoldier [Inria]
- Martin Vassor [Inria, until Jun 2021]

Technical Staff

- Roger Pissard-Gibollet [Inria, Engineer]

Interns and Apprentices

- Vincent Bonczak [Inria, from Feb 2021 until Jul 2021]
- Loric Gallier [Univ Grenoble Alpes, from Jun 2021 until Jul 2021]
- Remi Julo [Inria, from May 2021 until Jul 2021]

Administrative Assistant

- Julia Di Toro [Inria]

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 and dynamic reconfigurations 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 [41]. Witness component-based toolsets such as PTOLEMY [32], BIP [26], 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 [27]), 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, 31], 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 [28, 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 [46],

natural sciences, law [47], and statistics [49], 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 [44], to allow the diagnosis of faults in a complex concurrent system [40], or to enforce accountability [43], 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 [36]), or is broken (*e.g.*, by limiting fault propagation [51]).

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). We also consider the development of formal tools that can certify the result of industrial applications (see *e.g.*, CertiCAN in Sec. 7.2.2).

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. We also collaborate with RTaW regarding the integration of our CAN-bus analysis certifier (CertiCAN) in the RTaW-Pegase program suite.

5 Social and environmental responsibility

5.1 Footprint of research activities

2021 has again been quite an unusual year in terms of research activities: almost no travel, remote working... We have not yet computed the footprint of our research activities in 2021, but Roger Pissard has worked on a tool to do so with limited effort, which will soon be available.

5.2 Impact of research results

Our research on certification and fault-tolerance aims at making embedded systems safer. Certified systems tend also to be simpler, less depending on updates and therefore less prone to obsolescence. A potential major application of causality analysis is to help establish liability for accidents caused by software errors.

On the other hand, this type of research may contribute to make more acceptable or even to promote many problematic systems such as IoT, drones, avionics, autonomous vehicles, ... with a potential strong negative environmental impact.

Sophie Quinton and Éric Tannier (from the BEAGLE team in Lyon), with the help of many colleagues, including some in the SPADES team, have set up a series of one-day workshops called “Ateliers SEnS” (for Sciences-Environnements-Sociétés), which offer a venue for members of the research community (in

particular, but not limited to, researchers) to reflect on the social and environmental implications of their research. Ateliers SEnS have taken place for volunteers from almost all teams at INRIA Grenoble and Lyon as well as outside INRIA in Grenoble and Saclay. More Ateliers SEnS are already scheduled in Rennes, Bordeaux, Marseille, and Sophia in 2022. Participants to a workshop are able to replicate it, and some have already done so. Alain Girault will organize the Atelier SEnS for SPADES in the next few months.

Research into the connection between ICT (Information and Communication Technologies) and the environmental crisis has started in 2020 within the SPADES team, see Section 7.4.

6 New software and platforms

6.1 New software

6.1.1 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).

URL: https://team.inria.fr/spades/certican/

Authors: Xiaojie Guo, Pascal Fradet, Sophie Quinton

Contact: Xiaojie Guo

6.1.2 cloudnet

Name: Cloudnet

Keywords: Cloud configuration, Tosca, Docker Compose, Heat Orchestration Template, Alloy

Scientific Description: The multiplication of models, languages, APIs and tools for cloud and network configuration management raises heterogeneity issues that can be tackled by introducing a reference model. A reference model provides a common basis for interpretation for various models and languages, and for bridging different APIs and tools. The Cloudnet Computational Model formally specifies, in the Alloy specification language, a reference model for cloud configuration management. The Cloudnet software formally interprets several configuration languages in it, including the TOSCA configuration language, the OpenStack Heat Orchestration Template and the Docker Compose configuration language.

The use of the software shoes, for examples, how the Alloy formalization allowed us to discover several classes of errors in the OpenStack HOT specification.

Functional Description: Application of the Cloudnet model developed by Inria to software network deployment and reconfiguration description languages.

The Cloudnet model allows syntax and type checking for cloud configuration templates as well as their visualization (network diagram, UML deployment diagram). Three languages are addressed for the moment with the modules:

* Cloudnet TOSCA toolbox for TOSCA inncluding NFV description * cloudnet-hot for HOT (Heat Orchestration Template) from OpenStack * cloudnet-compose for Docker Compose

We can use directly the software from an Orange web portal: https://toscatoolbox.orange.com

URL: https://github.com/Orange-OpenSource/Cloudnet-TOSCA-toolbox

Publication: hal-02940938v1

Contact: Philippe Merle

Participants: Philippe Merle, Jean-Bernard Stefani, Roger Pissard-Gibollet, Souha Ben Rayana, Karine Guillouard, Meryem Ouzzif, Frédéric Klamm, Jean-Luc Coulin

Partner: Orange Labs

6.1.3 WasmCert

Name: WasmCert-Coq

Keywords: WebAssembly, Coq, Formalisation

Functional Description: WasmCert-Coq is a formalisation of WebAssembly (Wasm) in the Coq proof assistant. It precisely follows the standard Wasm 1.0, featuring a formalisation of the type system, the runtime semantics, the binary decoding, numeric operations, and the instantiation phase. It also features proof of soundness of the type system, as well as an executable interpreter proven correct with respect to the semantics.

URL: https://github.com/WasmCert/WasmCert-Coq

Contact: Martin Bodin

Partners: University of Cambridge, Imperial College London

7 New results

7.1 Design and Programming Models

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

7.1.1 Hypercells

The Hypercell framework, presented in [52], 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 the completion of Martin Vassor's PhD thesis [53], which reports on the formal characterization in the Hypercell framework of several encapsulation policies, and on a first implementation of the Hypercell framework in the Rust programming language.

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) [33, 48], we have written a survey providing a comprehensive description of the existing parametric dataflow MoCs [29], and we have studied *symbolic* analyses of dataflow graphs [30]. More recently, we have proposed an original method to deal with lossy communication channels in dataflow graphs [34].

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) [35]. 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 was the research topic of Arash Shafiei's PhD defended in Dec. 2021 [17] in collaboration with Orange Labs. This work has been summarized in a long paper [19] submitted for publication to a journal.

7.2 Certified Real-Time Programming

Participants: Pascal Fradet, Alain Girault, Xavier Nicollin, Sophie Quinton, Paolo Torrini.

7.2.1 A Markov Decision Process approach for energy minimization policies

In 2021, we have worked on a very general model of real-time systems, made of a single-core processor equipped with DVFS and an infinite sequence of preemptive real-time jobs. Each job J_i is characterized by the triplet (τ_i, w_i, d_i), where τ_i is the *inter-arrival time* between J_i and J_{i-1}, w_i is the *Actual Execution Time* (AET), upper-bounded by the WCET W, and d_i is the *relative deadline*, upper-bounded by Δ. The key point is that the system is *non-clairvoyant*, meaning that, at release time, w_i is not known until the job J_i actually terminates. What is available to the processor are the *statistical information* on the jobs' characteristics: release time, AET, and relative deadline. In this context, we have proposed a Markov Decision Process (MDP) solution to compute the optimal online speed policy guaranteeing that each job completes before its deadline and minimizing the energy consumption. To the best of our knowledge, our MDP solution is *the first to be optimal*. We have provided counter examples to prove that the two previous state of the art algorithms, namely OA [54] and PACE [45], are both sub-optimal.

Simulations show that our MDP solution outperforms the existing online solutions (OA and PACE), and can be very attractive in particular when the mean value of the execution time distribution is far from the WCET.

This was the topic of Stephan Plassart's PhD [50][10], funded by the CASERM Persyval project, who defended his PhD in June 2020.

7.2.2 Formal proofs for schedulability analysis of real-time systems

We contribute to Prosa [22], 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. the formal specification of real-time concepts
2. a better understanding of the role played by some assumptions in existing proofs;

3. a formal verification and comparison of different analysis techniques; and
4. the certification of (results of) existing analysis techniques or tools.

In the recent past, we have developed CertiCAN, a tool produced using the Coq proof assistant, allowing the formal certification of CAN bus analysis results. CertiCAN is able to certify the results of industrial CAN analysis tools, even for large systems. We have described this work in a long paper [20] submitted for publication to a journal.

Paolo Torrini has been pursuing his work on the development of certified schedulers in Coq [15].

Additional work (to be submitted for publication in 2022) has been done on the formal connection between Network Calculus and Response Time Analysis, as well as on the formalization in Prosa of Compositional Performance Analysis.

7.3 Fault Management and Causal Analysis

Participants: Gregor Goessler, Jean-Bernard Stefani, Aurélie Kong Win Chang, Thomas Mari, Giovanni Fabbretti, Pascal Fradet, Vincent Bonczak.

7.3.1 Causal Explanations for Embedded 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.

In particular, as part of the SEC project, we are investigating how such techniques can be extended to classes of hybrid systems. As a first result we have studied the problem of explaining faults in real-time systems [14]. We have provided a formal definition of causal explanations on dense-time models, based on the well-studied formalisms of timed automata and zone-based abstractions. We have proposed a symbolic formalization to effectively construct such explanations, which we have implemented in a prototype tool.

7.3.2 Causal Explanations in Concurrent Programs

As part of the DCore project on causal debugging of concurrent programs, the goal of Aurélie Kong Win Chang's PhD thesis is to investigate the use of abstractions to construct causal explanations for Erlang programs. We are interested in developing abstractions that "compose well" with causal analyses, and understanding precisely how explanations found on the abstraction relate to explanations on the concrete system. It is worth noting that the presence of abstraction, which inherently comes with some induction and extrapolation processes, completely recasts the issue of reasoning about causality. Causal traces do no longer describe only potential scenarios in the concrete semantics, but also mix some approximation steps coming from the computation of the abstraction itself. Therefore, not all explanations are replayable counter-examples: they may contain some steps witnessing some lack of accuracy in the analysis. Vice versa, a research question to be addressed is how to define causal analyses that have a well understood behavior under abstraction.

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 [9] we have proposed a self-modeling approach and an active diagnosis process for virtual networks that considers two types of knowledge to build the model: acquired knowledge and learned knowledge provided by fault injection to expand and validate the proposed model. Our experimental results from their application to a real-world virtual IP Multimedia Subsystem (vIMS) use case have shown their effectiveness in determining the root cause(s) of a failure and explaining fault propagation.

7.3.4 Reversibility for concurrent and distributed debugging

Concurrent and distributed debugging is a promising application of the notion of reversible computation [37]. As part of the ANR DCore project, we have contributed to the theory behind, and the development of the CauDEr reversible debugger for the Erlang programming language. Among our contributions, we expect to tackle the shared memory features and the distributed features if the Erlang environment, including its crash and recovery failure model. Initial steps in this direction, which tackled distributed primitives in Erlang allowing Erlang processes to gather a view of their environment, are reported in the publication [13].

7.3.5 Automatic transformations for fault tolerant circuits

In the past years, we have proposed automatic transformations to ensure fault-tolerance properties in digital circuits. We considered program transformations for hardware description languages (HDL) to tolerate single-event transients (SET) and fault models of the form “at most 1 SEU or SET within n clock cycles”. This year, we have studied program transformations to tolerate single-event multiple transients (SEMT) a more general kind of fault where a particle strike may cause multiple transient glitches in adjacent wires, cells or gates. A generalization of triple modular redundancy (TMR) that tolerates SEMT has been proposed. This work was the topic of Vincent Bonczak’s master project.

7.4 Transversal activity: ICT and the Anthropocene

Participants: Martin Bodin, Pascal Fradet, Alain Girault, Gregor Goessler, Xavier Nicollin, Roger Pissard, Sophie Quinton, Aina Rasoldier, Jean-Bernard Stefani.

Digital technologies are often presented as a powerful ally in the fight against climate change (see *e.g.*, the discourse around the “convergence of the digital and the ecological transitions”). We study this claim on the specific example of digital solutions for car sharing at a local scale (*i.e.*, a urban area of medium scale). Based on existing data (in particular the RPC, standing for Registre de Preuves de Covoiturage, a public data base, we investigate the profile of current users of car sharing applications and the scaling potential of such a solution. This work will be presented at the Archipel conference in 2022.

The SPADES team has started working together on a project proposal to investigate the current role played by ICT in the Anthropocene as well as new approaches to their design. We have identified the following main challenges: How do local measures meant to reduce the environmental impact of ICT relate (or not) to global effects? What can we learn from, and what are the limits of, current quantitative approaches for environmental impact assessment and their use for public debate and policy making? Which criteria could/should we take into account to design more responsible computer systems (other than efficiency, which is already well covered and subject to huge rebound effects in the case of digital technologies)? To come up with a solid research agenda, we are thus studying the state of the art of many new topics, including STS (Science and Technology Studies), low tech software and hardware, lifecyle assessment, (digital) commons... A new network of collaborations is also in the making, in particular with colleagues from social sciences.

8 Bilateral contracts and grants with industry

Participants: Jean-Bernard Stefani.

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 the verification of system configurations in cloud computing environments and software-defined networks.

9 Partnerships and cooperations

9.1 National initiatives

9.1.1 ANR

RT-proofs

Participants: Pascal Fradet, Sophie Quinton, Paolo Torrini.

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 2021 in connection with the RT-proofs project are described in Section 7.2.2.

DCORE

Participants: Gregor Goessler, Jean-Bernard Stefani, Giovanni Fabbretti, Pietro Lami, Aurélie Kong Win Chang.

DCORE is an ANR project between Inria project teams ANTIQUE, FOCUS and SPADES, and the IRIF lab, running from 2019 to 2024.

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 [42, 38].

10 Dissemination

Participants: Pascal Fradet, Alain Girault, Gregor Gössler, Xavier Nicollin, Sophie Quinton, Jean-Bernard Stefani.

10.1 Promoting scientific activities

10.1.1 Scientific events: selection

Member of the conference program committees

- Alain Girault has been PC member of FDL'21.
- Gregor Gössler has been PC member of EMSOFT'21.

10.1.2 Journal

Member of the editorial boards

- Alain Girault is associate editor of Eurasip Journal of Embedded Systems and of Real-Time Systems Journal. He was also guest editor for a special issue of ACM TECS on Specification and Design Languages.

Reviewer - reviewing activities

- Alain Girault has reviewed articles for ACM Trans. on Embedded Computing Systems, IEEE Embedded Systems Letters, and J. of Grid Computing.
- Gregor Gössler has reviewed an article for Mathematical Structures in Computer Science.

10.1.3 Invited talks

- Sophie Quinton gave invited talks on the environmental impact of ICT at ECRTS'21, CEA-LETI, the HiPEAC Autumn Computer Systems Week, and RDA VP18.
- Sophie Quinton gave a talk on the history of ECRTS industrial challenges at ECRTS'21.

10.1.4 Leadership within the scientific community

- Sophie Quinton is a member of the ECRTS Executive Committee.
- Sophie Quinton was a member of the ACM SIGBED Executive Committee and Associate Editor of the SIGBED Blog until June 2021.
- Sophie Quinton co-chairs a working group of the GDR CIS associated with the Center for Internet and Society focused on environmental issues.

10.1.5 Scientific expertise

- Sophie Quinton was interviewed by France Stratégie for their report on the Internet of Things.
- Sophie Quinton contributed to the upcoming "Guide de bonnes pratiques numérique responsable" by the DINUM (Direction interministérielle du numérique).

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 research center. He is the local correspondent for the young researchers Inria mission (“Mission jeunes chercheurs”) and the substitute of the director of the Inria Grenoble research center at the doctoral school council (MSTII).
- Alain Girault is Deputy Scientific Director for Inria, in charge of the domain “Algorithmics, Programming, Software and Architecture”.
- Sophie Quinton leads the SEnS-GRA group which hosts discussions and proposes actions regarding the environmental and societal impact of our research at Inria Grenoble Rhône-Alpes.
- Jean-Bernard Stefani has been the Délégué Scientifique of the INRIA Grenoble-Rhône-Alpes research center till June 2021.

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: Jean-Bernard Stefani, Formal Aspects of Component Software, 9h, MOSIG, Univ. Grenoble Alpes, France.
- Master: Sophie Quinton gave a lecture (3h) on the environmental impact of ICT ("Les enjeux environnementaux du numérique") at Polytech Grenoble (niveau M2).

- Master: Sophie Quinton gave a guest lecture (2h) on "Formal Methods for Real-Time Systems" at TU Kaiserslautern (niveau M2).

- Formation continue: Sophie Quinton co-organized two work groups during the ANF (Action Nationale de Formation) organized by EcoInfo on the environmental impact of ICT.

10.2.2 Supervision

- Pascal Fradet, Alain Girault & Xavier Nicollin: PhD completed: Arash Shafiei, "RDF : A Reconfigurable Dataflow Model of Computation", Université Grenoble - Alpes, December 2021 [17].

- 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.

- Gregor Gössler: PhD in progress: Aurélie Kong Win Chang, "Abstractions for causal analysis and explanations in concurrent programs"; since January 2021; co-advised by Gregor Gössler and Jérôme Feret.

- Sophie Quinton and Alain Girault: PhD in progress: Aina Rasoldier, ICT in the Anthropocene: Technical and social challenges at the local scale.

- Jean-Bernard Stefani: PhD in progress: Giovanni Fabbretti, "Reversibility for distributed computations", since February 2021.

- Jean-Bernard Stefani: PhD in progress: Pietro Lami, "Reversibility and transactions", since October 2021.

- Jean-Bernard Stefani: PhD completed: Martin Vassor, "Graphes de localités : une approche formelle à l’encapsulation et implémentation", Université Grenoble-Alpes, May 2021 [53]

10.2.3 Juries

- Alain Girault was referee for the PhD thesis of Gautham Nayak Seetanadi (Lund University).

- Alain Girault was referee for the PhD thesis of Amna Gharbi (Telecom Paris Tech).

- Gregor Gössler was president of the PhD jury of Karim Assaad (Univ. Grenoble Alpes).

- Sophie Quinton was member of the PhD jury of Zakaria Ournani (University of Lille).

- Sophie Quinton was a member of the Master thesis committee of Bastien Béchadergue (EHESS).

- Sophie Quinton was member of a recruiting committee for a MCF position in Nantes.

10.3 Popularization

10.3.1 Articles and contents

- Sophie Quinton wrote two articles ("fiches concepts") for the INRIA MOOC on the environmental impacts of ICT: one on the environmental footprint of a piece of personal equipment and the second on the main concepts behind the Life Cycle Analysis method.

10.3.2 Education

- Sophie Quinton was part of the scientific committee of the "COP2 étudiante".

10.3.3 Interventions

- Sophie Quinton gave a keynote on the environmental impact of ICT ("Numérique et environnement : Quels impacts ? Quels enjeux ?") at the MathC2+ event organized by Inria and at the Fête de la science.
- Sophie Quinton contributed to a workshop on the ethics of ICT at Sharin'Grenoble 2021.
- Sophie Quinton gave an interview on the environmental impact of ICT on AlligreFM and participated to a panel on digital sobriety ("Faut-il faire des usages un levier de la sobriété numérique ?") organized the FGI (Forum pour la Gouvernance d'Internet).
- Sophie Quinton contributed to a quiz on sustainable development for the DGD-I seminar.

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 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.

[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] S. Cherrared, S. Imadali, E. Fabre and G. Gössler. 'SFC Self-Modeling and Active Diagnosis'. In: *IEEE Transactions on Network and Service Management* 18.3 (Sept. 2021), pp. 2515–2530. DOI: 10.1109/TNSM.2021.3086424. URL: https://hal.inria.fr/hal-03352706.

[10] 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* (2021). DOI: 10.1007/s10626-020-00327-9. URL: https://hal.archives-ouvertes.fr/hal-03030416.

[11] 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.

[12] R. Le Guillou, M. Schmoll, B. Sijobert, D. Lobato Borges, E. Fachin-Martins, H. Resende, R. Pissard-Gibollet, C. Fattal and C. Azevedo Coste. 'A Novel Framework for Quantifying Accuracy and Precision of Event Detection Algorithms in FES-Cycling'. In: *Sensors* 21.13 (2021), pp. 1–13. DOI: 10.3390/s21134571. URL: https://hal.archives-ouvertes.fr/hal-03278596.

International peer-reviewed conferences

[13] G. Fabbretti, I. Lanese and J.-B. Stefani. 'Causal-Consistent Debugging of Distributed Erlang Programs'. In: *Reversible Computation*. RC 2021 - 13th Conference on Reversible Computation. Vol. 12805. Lecture Notes in Computer Science. Nagoya, Japan: Springer, 23rd June 2021, pp. 79–95. DOI: 10.1007/978-3-030-79837-6_5. URL: https://hal.inria.fr/hal-03338670.

[14] T. Mari, T. Dang and G. Gössler. 'Explaining Safety Violations in Real-Time Systems'. In: *Proc. 19th FORMATS 2021, Paris, France*. FORMATS 2021 - Formal Modeling and Analysis of Timed Systems. Paris, France, Aug. 2021, pp. 100–116. DOI: 10.1007/978-3-030-85037-1_7. URL: https://hal.inria.fr/hal-03348010.

Conferences without proceedings

[15] X. Guo, L. Rieg and P. Torrini. 'A generic approach for the certified schedulability analysis of software systems'. In: Real-Time Computing Systems and Applications. Houston (online), United States, 18th Aug. 2021. URL: https://hal.archives-ouvertes.fr/hal-03540548.

[16] C. Watt, X. Rao, J. Pichon-Pharabod, M. Bodin and P. Gardner. 'Two Mechanisations of WebAssembly 1.0'. In: FM 2021 - Formal Methods. Beijing, China, 20th Nov. 2021, pp. 1–19. URL: https://hal.archives-ouvertes.fr/hal-03353748.

Doctoral dissertations and habilitation theses

[17] A. Shafiei. 'RDF : A Reconfigurable Dataflow Model of Computation'. Université Grenoble - Alpes, 16th Dec. 2021. URL: https://tel.archives-ouvertes.fr/tel-03531869.

Reports & preprints

[18] G. Fabbretti, I. Lanese and J.-B. Stefani. *Causal-Consistent Debugging of Distributed Erlang Programs - Technical Report*. Inria - Research Centre Grenoble – Rhône-Alpes, 7th July 2021. URL: https://hal.inria.fr/hal-03247624.

[19] P. Fradet, A. Girault, R. Krishnaswamy, X. Nicollin and A. Shafiei. *RDF: A Reconfigurable Dataflow Model of Computation*. RR-9439. Inria Grenoble Rhône-Alpes, Université de Grenoble, 20th Dec. 2021. URL: https://hal.inria.fr/hal-03495883.

[20] P. Fradet, X. Guo and S. Quinton. *CertiCAN: Certifying CAN Analyses and Their Results*. RR-9443. Inria - Research Centre Grenoble – Rhône-Alpes, 20th Dec. 2021, pp. 1–32. URL: https://hal.inria.fr/hal-03499968.

[21] T. Mari, T. Dang and G. Gössler. *Explaining Safety Violations in Real-Time Systems*. RR-9420. Paris, France: INRIA; Verimag, Université Grenoble Alpes, Sept. 2021. URL: https://hal.inria.fr/hal-03348046.

11.3 Cited publications

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

[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] 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).

[27] 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).

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

[29] 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.

[30] 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.

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

[32] 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).

[33] 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.

[34] 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.

[35] 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.

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

[37] E. Giachino, I. Lanese and C. A. Mezzina. 'Causal-Consistent Reversible Debugging'. In: *17th International Conference Fundamental Approaches to Software Engineering (FASE)*. Vol. 8411. Lecture Notes in Computer Science. 2014, pp. 370–384.

[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] 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.

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

[42] 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.

[43] 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.

[44] 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.

[45] J. Lorch and A. Smith. 'PACE: A New Approach to Dynamic Voltage Scaling'. In: *IEEE Trans. on Computers* 53.7 (2004), pp. 856–869.

[46] 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.

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

[48] 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.

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

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

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

[52] 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.

[53] M. Vassor. 'Graphes de localités : une approche formelle à l'encapsulation et implémentation'. Theses. Université Grenoble Alpes [2020-....], May 2021. URL: https://tel.archives-ouvertes.fr/tel-03354281.

[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.