



IN PARTNERSHIP WITH:
CNRS

**Université Nice - Sophia
Antipolis**

Activity Report 2016

Project-Team AOSTE

Models and methods of analysis and
optimization for systems with real-time and
embedding constraints

IN COLLABORATION WITH: Laboratoire informatique, signaux systèmes de Sophia Antipolis (I3S)

RESEARCH CENTERS
**Sophia Antipolis - Méditerranée
Paris**

THEME
Embedded and Real-time Systems

Table of contents

1. Members	1
2. Overall Objectives	3
3. Research Program	3
3.1. Models of Computation and Communication (MoCCs)	3
3.1.1. K-periodic static scheduling and routing in Process Networks	4
3.1.2. Endochrony and GALS implementation of conflict-free polychronous programs	4
3.2. Logical Time in Model-Driven Embedded System Design	4
3.3. The AAA (Algorithm-Architecture Adequation) methodology and Real-Time Scheduling	5
3.3.1. Algorithm-Architecture Adequation	5
3.3.2. Distributed Real-Time Scheduling and Optimization	6
4. Application Domains	7
4.1. System Engineering Environments	7
4.2. Many-Core Embedded Architectures	7
4.3. Transportation and the avionic domain	7
5. New Software and Platforms	7
5.1. EVT Kopernic	7
5.2. KPASSA	8
5.3. Lopht	8
5.4. SAS	9
5.5. SynDEX	9
5.6. TimeSquare	10
5.7. Vercors	10
6. New Results	11
6.1. CCSL as a Logical Clock Calculus Algebra: expressiveness and analysis techniques	11
6.2. Industrial design flow for Embedded System Engineering	11
6.3. Coordination of heterogeneous Models of Computation as Domain-Specific Languages	12
6.4. SoC multiview (meta)modeling for performance, power, and thermal aspects	12
6.5. MoCs and novel architectures	12
6.6. Solving AAA constraints analytically	13
6.7. Coupling SystemC and FMI for co-simulation of Cyber-Physical Systems	13
6.8. Behavioural Semantics of Open pNets	13
6.9. Behavioural semantics for GCM components	13
6.10. Performance analysis and optimisation of an HPC scientific application	14
6.11. Formal translation validation of multi-processor real-time schedules	14
6.12. Lopht back-end for TTEthernet-based distributed systems	15
6.13. Uniprocessor Real-Time Scheduling	15
6.14. Multiprocessor Real-Time Scheduling	16
6.15. Probabilistic Solutions for Hard Real-Time Systems	17
7. Bilateral Contracts and Grants with Industry	17
8. Partnerships and Cooperations	17
8.1. National Initiatives	17
8.1.1. ANR	17
8.1.1.1. HOPE	17
8.1.1.2. GeMoC	17
8.1.1.3. FUI CLISTINE	18
8.1.1.4. FUI Waruna	18
8.1.2. Investissements d’Avenir	18
8.1.2.1. DEPARTS	18
8.1.2.2. CLARITY	18

8.1.2.3. Capacites	18
8.2. European Initiatives	19
8.3. International Initiatives	19
8.3.1. FM4CPS	19
8.3.2. Inria International Partners	20
8.4. International Research Visitors	20
9. Dissemination	20
9.1. Promoting Scientific Activities	20
9.1.1. Scientific Events Organisation	20
9.1.2. Scientific Events Selection	21
9.1.2.1. Chair of Conference Program Committees	21
9.1.2.2. Member of the Conference Program Committees	21
9.1.3. Journal	21
9.1.4. Invited Talks	21
9.1.5. Leadership within the Scientific Community	21
9.1.6. Scientific Expertise	21
9.1.7. Research Administration	21
9.2. Teaching - Supervision - Juries	22
9.2.1. Teaching	22
9.2.2. Supervision	22
9.2.3. Juries	23
9.3. Popularization	23
10. Bibliography	23

Project-Team AOSTE

Creation of the Project-Team: 2004 July 01, end of the Project-Team: 2016 December 31

Keywords:

Computer Science and Digital Science:

- 1.1.1. - Multicore
- 1.1.2. - Hardware accelerators (GPGPU, FPGA, etc.)
- 1.1.12. - Non-conventional architectures
- 1.2.3. - Routing
- 1.2.5. - Internet of things
- 1.2.7. - Cyber-physical systems
- 1.5.1. - Systems of systems
- 1.5.2. - Communicating systems
- 2.1.1. - Semantics of programming languages
- 2.1.6. - Concurrent programming
- 2.1.8. - Synchronous languages
- 2.1.10. - Domain-specific languages
- 2.2.4. - Parallel architectures
- 2.2.5. - GPGPU, FPGA, etc.
- 2.3. - Embedded and cyber-physical systems
- 2.4.1. - Analysis
- 2.4.2. - Model-checking
- 4.5. - Formal methods for security
- 6.1.5. - Multiphysics modeling
- 6.2.7. - High performance computing
- 7.2. - Discrete mathematics, combinatorics
- 7.11. - Performance evaluation

Other Research Topics and Application Domains:

- 5.1. - Factory of the future
- 5.4. - Microelectronics
- 6.1.1. - Software engineering
- 6.4. - Internet of things
- 6.6. - Embedded systems
- 8.1. - Smart building/home

1. Members

Research Scientists

Robert de Simone [Inria, Senior Researcher, Team leader, Sophia Antipolis - Méditerranée, HDR]
Yves Sorel [Inria, Senior Researcher, Deputy leader, Paris]
Liliana Cucu [Inria, Researcher, Paris, HDR]
Robert Davis [Inria, Advanced Research position, Paris]
Eric Madelaine [Inria, Researcher, Sophia Antipolis - Méditerranée, HDR]

Dumitru Potop Butucaru [Inria, Researcher, Paris, HDR]

Faculty Members

Julien Deantoni [Univ. Nice Sophia-Antipolis, Associate Professor, Sophia Antipolis - Méditerranée]

Frederic Mallet [Univ. Nice Sophia-Antipolis, Professor, Sophia Antipolis - Méditerranée, HDR]

Marie Agnes Peraldi Frati [Univ. Nice Sophia-Antipolis, Associate Professor, Sophia Antipolis - Méditerranée]

Sid Touati [Univ. Nice Sophia-Antipolis, Professor, Sophia Antipolis - Méditerranée, HDR]

Engineers

Irina-Mariuca Asavaoe [Inria, funded on CAPACITES contract, from Sep 2016, Paris]

Mihail Asavaoe [Inria, Paris]

Antoine Bertout [Inria, funded on CAPACITES then WARUNA contracts, from Feb 2016, Paris]

Adriana Gogonel [Inria, funded on PROXIMA then WARUNA contracts, Paris]

Raul-Adrian Gorcitz [Inria, until Mar 2016, funded by ASSUME project, Paris]

Luc Hogie [IR CNRS, Sophia Antipolis - Méditerranée]

Tomasz Kloda [Inria, funded on CAPACITES then WARUNA contracts, from Feb 2016, Paris]

Mehdi Mezouak [Inria, Intern then Engineer, from Apr 2016, Paris]

Ales Mishchenko [Inria, funded on CLARITY project, Sophia Antipolis - Méditerranée]

Albert Savary [Inria, funded on CLISTINE contract, until Aug 2016, Sophia Antipolis - Méditerranée]

PhD Students

Dongdong An [ECNU Shanghai PhD, from Sep 2016, Sophia Antipolis - Méditerranée]

Slim Ben Amor [Inria, Intern then PhD Student, from May 2016, Paris]

Keryan Didier [Inria, Paris]

Amani Khecharem [Inria, until Apr 2016, Sophia Antipolis - Méditerranée]

Vincent Kherbache [Inria, funded by FP7 DC4CITY project, Sophia Antipolis - Méditerranée]

Emilien Kofman [Inria, Labex UCNSophia, Sophia Antipolis - Méditerranée]

Oleksandra Kulankhina [Inria, until Sep 2016, Sophia Antipolis - Méditerranée]

Cristian Maxim [AIRBUS CIFRE grant, Paris]

Amin Oueslati [Inria, funded on FUI Clistine project, Sophia Antipolis - Méditerranée]

Salah Eddine Saidi [funded by IFP Energies nouvelles, Paris]

Walid Talaboulma [Inria, Paris]

Matias Vara Larsen [Univ. Nice Sophia-Antipolis, ANR GeMoC funding, until Apr 2016., Sophia Antipolis - Méditerranée]

Hui Zhao [Inria, funded on CLARITY prject, from Feb 2016, Sophia Antipolis - Méditerranée]

Post-Doctoral Fellows

Dorin Maxim [Inria, funded on PROXIMA project, until October 2016, Paris]

Pedro Velho [Inria, until Jan 2016, Sophia Antipolis - Méditerranée]

Administrative Assistants

Christine Anocq [Inria, Paris]

Patricia Lachaume [Inria, Sophia Antipolis - Méditerranée]

Others

Jimmy Auk [Inria, Master2 intern, from Mar 2016 until Aug 2016, Sophia Antipolis - Méditerranée]

Nicolas Bicheron [intern, from May 2016 until Jul 2016, Sophia Antipolis - Méditerranée]

Stefano Centomo [Inria, MAster2 intern, until Feb 2016, Sophia Antipolis - Méditerranée]

Laurent George [Univ. Paris XII, external collaborator, Paris, HDR]

Giovanni Liboni [Inria, Master2 intern, from Dec 2016, Sophia Antipolis - Méditerranée]

Hala Mazirh [Inria, intern, until Aug 2016, Sophia Antipolis - Méditerranée]

Luis Agustin Nieto [Inria International Master intern, until Feb 2016, Sophia Antipolis - Méditerranée]

Xudong Qin [ECNU Shanghai Master intern, from Sep 2016, Sophia Antipolis - Méditerranée]

Zhouhang Shao [UCSD Summer intern, from Jul 2016 until Sep 2016, Sophia Antipolis - Méditerranée]

William Zaborowski [Inria, Intern, from Apr 2016 until Jun 2016, Sophia Antipolis - Méditerranée]

2. Overall Objectives

2.1. Embedded System Design

Typical embedded software *applications* display a mix of multimedia signal/data processing with modal interfaces, resulting in heterogenous concurrent data-flow streaming models, and often stringent real-time constraints. Similarly, embedded *architectural platforms* are becoming increasingly parallel, with dedicated hardware accelerators and manycore processors. The optimized compilation of such kinds of applications onto such execution platforms involves complex mapping issues, both in terms of spatial distribution and in terms of temporal scheduling. Currently, it is far from being a fully automatic compilation process as in the case of commodity PC applications. Models are thus needed, both as formal mathematical objects from theoretical computer science to provide foundations for embedded system design, and also as engineering models to support an effective design flow.

Our general approach is directly inspired from the theories of synchronous languages, process networks, and of real-time distributed scheduling. We insist on the introduction of *logical time* as functional design ingredient to be explicitly considered as first-class modeling element of systems. Logical time is based on logical clocks, where such a clock can be defined as any meaningful sequence of event occurrences, usually meant as activation/triggering conditions for actions and operations in the systems. So logical time can be multiform, a global partial order built from local total orders of clocks. In the course of the design flow *time refinement* takes place, as decisions are made towards placement and timing of various tasks and operations. This solves in part the constraints between clocks, committing to schedule and placement decisions. The final version should be totally ordered, and then subject to physical timing verification as to physical constraints.

The general (logical) *Time Model* has been standardized as part of the OMG profile for Modeling and Analysis of Real-Time Embedded systems ([MARTE](#)).

Work on polychronous formalisms (descending from [ESTEREL](#)), on a Clock Constraint Specification Language (CCSL) handling logical time, on Application-Architecture Adequation approach and real-time scheduling results has been progressed over the years, resulting in software environments such as [SYNDEX](#) or [TimeSquare](#).

3. Research Program

3.1. Models of Computation and Communication (MoCCs)

Participants: Julien Deantoni, Robert de Simone, Frédéric Mallet, Dumitru Potop Butucaru.

Esterel, SyncCharts, synchronous formalisms, Process Networks, Marked Graphs, Kahn networks, compilation, synthesis, formal verification, optimization, allocation, refinement, scheduling

Formal Models of Computation form the basis of our approach to Embedded System Design. Because of the growing importance of communication handling, it is now associated with the name, MoCC in short. The appeal of MoCCs comes from the fact that they combine features of mathematical models (formal analysis, transformation, and verification) with those of executable specifications (close to code level, simulation, and implementation). Examples of MoCCs in our case are mainly synchronous reactive formalisms and dataflow process networks. Various extensions or specific restrictions enforce respectively greater expressivity or more focused decidable analysis results.

DataFlow Process Networks and Synchronous Reactive Languages such as ESTEREL/SYNCHARTS and SIGNAL/POLYCHRONY [54], [55], [49], [15], [4], [13] share one main characteristic: they are specified in a self-timed or loosely timed fashion, in the asynchronous data-flow style. But formal criteria in their semantics ensure that, under good correctness conditions, a sound synchronous interpretation can be provided, in which all treatments (computations, signaling communications) are precisely temporally mapped. This is referred to as clock calculus in synchronous reactive systems, and leads to a large body of theoretical studies and deep results in the case of DataFlow Process Networks [50], [48] (consider SDF balance equations for instance [56]).

As a result, explicit schedules become an important ingredient of design, which ultimately can be considered and handled by the designer him/herself. In practice such schedules are sought to optimize other parts of the design, mainly buffering queues: production and consumption of data can be regulated in their relative speeds. This was specially taken into account in the recent theories of Latency-Insensitive Design [51], or N-synchronous processes [52], with some of our contributions [6].

Explicit schedule patterns should be pictured in the framework of low-power distributed mapping of embedded applications onto manycore architectures, where they could play an important role as theoretical formal models on which to compute and optimize allocations and performances. We describe below two lines of research in this direction. Striking in these techniques is the fact that they include time and timing as integral parts of early functional design. But this original time is logical, multiform, and only partially ordering the various functional computations and communications. This approach was radically generalized in our team to a methodology for logical time based design, described next (see 3.2).

3.1.1. *K-periodic static scheduling and routing in Process Networks*

In the recent years we focused on the algorithm treatments of ultimately k-periodic schedule regimes, which are the class of schedules obtained by many of the theories described above. An important breakthrough occurred when realizing that the type of ultimately periodic binary words that were used for reporting *static scheduling* results could also be employed to record a completely distinct notion of ultimately k-periodic route switching patterns, and furthermore that commonalities of representation could ease combine them together. A new model, by the name of K-periodical Routed marked Graphs (KRG) was introduced, and extensively studied for algebraic and algorithmic properties [5].

The computations of optimized static schedules and other optimal buffering configurations in the context of latency-insensitive design led to the K-Passa software tool development (now terminated)

3.1.2. *Endochrony and GALS implementation of conflict-free polychronous programs*

The possibility of exploring various schedulings for a given application comes from the fact that some behaviors are truly concurrent, and mutually *conflict-free* (so they can be executed independently, with any choice of ordering). Discovering potential asynchronous inside synchronous reactive specifications then becomes something highly desirable. It can benefit to potential distributed implementation, where signal communications are restricted to a minimum, as they usually incur loss in performance and higher power consumption. This general line of research has come to be known as Endochrony, with some of our contributions [11].

3.2. Logical Time in Model-Driven Embedded System Design

Participants: Julien Deantoni, Frédéric Mallet, Marie Agnes Peraldi Frati, Robert de Simone.

Starting from specific needs and opportunities for formal design of embedded systems as learned from our work on MoCCs (see 3.1), we developed a Logical Time Model as part of the official **OMG UML profile MARTE** for Modeling and Analysis of Real-Time Embedded systems. With this model is associated a Clock Constraint Specification Language (CCSL), which allows to provide loose or strict logical time constraints between design ingredients, be them computations, communications, or any kind of events whose repetitions can be conceived as generating a logical conceptual clock (or activation condition). The definition of CCSL is provided in [1].

Our vision is that many (if not all) of the timing constraints generally expressed as physical prescriptions in real-time embedded design (such as periodicity, sporadicity) could be expressed in a logical setting, while actually many physical timing values are still unknown or unspecified at this stage. On the other hand, our logical view may express much more, such as loosely stated timing relations based on partial orderings or partial constraints.

So far we have used CCSL to express important phenomena as present in several formalisms: **AADL** (used in avionics domain), **EAST-ADL2** (proposed for the **AutoSar** automotive electronic design approach), **IP-Xact** (for System-on-Chip (*SoC*) design). The difference here comes from the fact that these formalisms were formerly describing such issues in informal terms, while CCSL provides a dedicated formal mathematical notation. Close connections with synchronous and polychronous languages, especially Signal, were also established; so was the ability of CCSL to model dataflow process network static scheduling.

In principle the MARTE profile and its Logical Time Model can be used with any UML editor supporting profiles. It has also evolved to become a Domain-Specific Language, independent of UML. It is connected to the **CAPELLA** environment, and the **PAPYRUS** open-source editor. We developed under Eclipse the **TIMESQUARE** solver and emulator for CCSL constraints (see 5.6), with its own graphical interface, as a stand-alone software module, again now coupled with MARTE and Papyrus, but also as part of the **GeMoC studio** environment developed in the GeMoC ANR project.

The MARTE profile and its Logical Time Model can be used with any UML editor supporting profiles but evolved to become a DSL independent of UML. We developed as a set of eclipse plugins the **TIMESQUARE** tool to edit and simulate CCSL specifications. TimeSquare has been coupled with various tools like Papyrus or Capella and is now part of the concurrent solver integrated in the **GEMOC studio**.

While CCSL constraints may be introduced as part of the intended functionality, some may also be extracted from requirements imposed either from real-time user demands, or from the resource limitations and features from the intended execution platform. Sophisticated detailed descriptions of platform architectures are allowed using MARTE, as well as formal allocations of application operations (computations and communications) onto platform resources (processors and interconnects). This is of course of great value at a time where embedded architectures are becoming more and more heterogeneous and parallel or distributed, so that application mapping in terms of spatial allocation and temporal scheduling becomes harder and harder. This approach is extensively supported by the MARTE profile and its various models. As such it originates from the Application-Architecture-Adequation (AAA) methodology, first proposed by Yves Sorel, member of Aoste. AAA aims at specific distributed real-time algorithmic methods, described next in 3.3.

Of course, while logical time in design is promoted here, and our works show how many current notions used in real-time and embedded systems synthesis can naturally be phrased in this model, there will be in the end a phase of validation of the logical time assumptions (as is the case in synchronous circuits and SoC design with timing closure issues). This validation is usually conducted from Worst-Case Execution Time (WCET) analysis on individual components, which are then used in further analysis techniques to establish the validity of logical time assumptions (as partial constraints) asserted during the design.

3.3. The AAA (Algorithm-Architecture Adequation) methodology and Real-Time Scheduling

Participants: Liliana Cucu, Laurent George, Dumitru Potop Butucaru, Yves Sorel.

Note: The AAA methodology and the SynDEX environment are fully described at <http://www.syndex.org/>, together with **relevant publications**.

3.3.1. Algorithm-Architecture Adequation

The **AAA methodology** relies on distributed real-time scheduling and relevant optimization to connect an Algorithm/Application model to an Architectural one. We now describe its premises and benefits.

The Algorithm model is an extension of the well known data-flow model from Dennis [53]. It is a directed acyclic hyper-graph (DAG) that we call “conditioned factorized data dependence graph”, whose vertices are “operations” and hyper-edges are directed “data or control dependences” between operations. The data dependences define a partial order on the operations execution. The basic data-flow model was extended in three directions: first infinite (resp. finite) repetition of a sub-graph pattern in order to specify the reactive aspect of real-time systems (resp. in order to specify the finite repetition of a sub-graph consuming different data similar to a loop in imperative languages), second “state” when data dependences are necessary between

different infinite repetitions of the sub-graph pattern introducing cycles which must be avoided by introducing specific vertices called “delays” (similar to z^{-n} in automatic control), third “conditioning” of an operation by a control dependence similar to conditional control structure in imperative languages, allowing the execution of alternative subgraphs. Delays combined with conditioning allow the programmer to specify automata necessary for describing “mode changes”.

The Architecture model is a directed graph, whose vertices are of two types: “processor” (one sequencer of operations and possibly several sequencers of communications) and “medium” (support of communications), and whose edges are directed connections.

The resulting implementation model [9] is obtained by an external compositional law, for which the architecture graph operates on the algorithm graph. Thus, the result of such compositional law is an algorithm graph, “architecture-aware”, corresponding to refinements of the initial algorithm graph, by computing spatial (distribution) and timing (scheduling) allocations of the operations onto the architecture graph resources. In that context “Adequation” refers to some search amongst the solution space of resulting algorithm graphs, labelled by timing characteristics, for one algorithm graph which verifies timing constraints and optimizes some criteria, usually the total execution time and the number of computing resources (but other criteria may exist). The next section describes distributed real-time schedulability analysis and optimization techniques for that purpose.

3.3.2. Distributed Real-Time Scheduling and Optimization

We address two main issues: uniprocessor and multiprocessor real-time scheduling where constraints must mandatorily be met, otherwise dramatic consequences may occur (hard real-time) and where resources must be minimized because of embedded features.

In the case of uniprocessor real-time scheduling, besides the classical deadline constraint, often equal to a period, we take into consideration dependences between tasks and several, latencies. The latter are complex related “end-to-end” constraints. Dealing with multiple real-time constraints raises the complexity of the scheduling problems. Moreover, because the preemption leads, at least, to a waste of resources due to its approximation in the WCET (Worst Execution Time) of every task, as proposed by Liu and Leyland [57], we first studied non-preemptive real-time scheduling with dependences, periodicities, and latencies constraints. Although a bad approximation of the preemption cost, may have dramatic consequences on real-time scheduling, there are only few researches on this topic. We have been investigating preemptive real-time scheduling since few years, and we focus on the exact cost of the preemption. We have integrated this cost in the schedulability conditions that we propose, and in the corresponding scheduling algorithms. More generally, we are interested in integrating in the schedulability analyses the cost of the RTOS (Real-Time Operating System), for which the cost of preemption is the most difficult part because it varies according to the instance (job) of each task.

In the case of multiprocessor real-time scheduling, we chose at the beginning the partitioned approach, rather than the global approach, since the latter allows task migrations whose cost is prohibitive for current commercial processors. The partitioned approach enables us to reuse the results obtained in the uniprocessor case in order to derive solutions for the multiprocessor case. We consider also the semi-partitioned approach which allows only some migrations in order to minimize the overhead they involve. In addition to satisfy the multiple real-time constraints mentioned in the uniprocessor case, we have to minimize the total execution time (makespan) since we deal with automatic control applications involving feedback loops. Furthermore, the domain of embedded systems leads to solving minimization resources problems. Since these optimization problems are NP-hard we develop exact algorithms (B & B, B & C) which are optimal for simple problems, and heuristics which are sub-optimal for realistic problems corresponding to industrial needs. Long time ago we proposed a very fast “greedy” heuristics [8] whose results were regularly improved, and extended with local neighborhood heuristics, or used as initial solutions for metaheuristics.

In addition to the spatial dimension (distributed) of the real-time scheduling problem, other important dimensions are the type of communication mechanisms (shared memory vs. message passing), or the source of control and synchronization (event-driven vs. time-triggered). We explore real-time scheduling

on architectures corresponding to all combinations of the above dimensions. This is of particular impact in application domains such as automotive and avionics (see 4.3).

The arrival of complex hardware responding to the increasing demand for computing power in next generation systems exacerbates the limitations of the current worst-case real-time reasoning. Our solution to overcome these limitations is based on the fact that worst-case situations may have a extremely low probability of appearance within one hour of functioning (10^{-45}), compared to the certification requirements for instance (10^{-9} for the highest level of certification in avionics). Thus we model and analyze the real-time systems using probabilistic models and we propose results that are fundamental for the probabilistic worst-case reasoning over a given time window.

4. Application Domains

4.1. System Engineering Environments

Participants: Robert de Simone, Julien Deantoni, Frédéric Mallet, Marie Agnes Peraldi Frati.

In the case of Embedded and Cyber-Physical Systems, the cyber/digital design of discrete controllers is only a part of a larger design process, we other aspects of the physical environment need to be considered as well, involving constraints and requirements on the global system (people even talk of *Systems of Systems*). Dedicated environments are now being defined, also considering system life-cycle and component reuse in this larger setting, under the name of *Atelier Génie Système* (in French). Such efforts usually involve large industrial end-users, together with software houses of tool vendors, and academic partners altogether. An instance of such environment is the Cappella (open-source, Eclipse) environment, promoted by the Clarity project and its associated consortium 8.1.2.2.

4.2. Many-Core Embedded Architectures

Participants: Robert de Simone, Dumitru Potop Butucaru, Liliana Cucu, Yves Sorel.

The AAA approach (fitting embedded applications onto embedded architectures) requires a sufficiently precise description of (a model of) the architecture (description platform). Such platforms become increasingly heterogeneous, and we had to consider a number of emerging ones with that goal in mind, such as Kalray MPPA (in the CAPACITES project 8.1.2.3, IntelCore dual CPU/GPU structure in a collaboration with Kontron, ARM big.LITTLE architecture in the course of the HOPE ANR project 8.1.1.1, or a dedicated supercomputer based on Network-on-Board interconnect in the Clistine project 8.1.1.3.

4.3. Transportation and the avionic domain

Participants: Robert de Simone, Julien Deantoni, Frédéric Mallet, Marie Agnès Peraldi Frati, Dumitru Potop Butucaru, Liliana Cucu, Yves Sorel.

A large number of our generic activities, both on modeling and design, and on analysis and implementation of real-time embedded systems, found specific applications in the avionic field (with partners such as Airbus, Thales, Safran,...), while other targets remained less attainable (car industry for instance).

5. New Software and Platforms

5.1. EVT Kopernic

Extreme Value Theory for Keeping Worst Reasoning Appropriate for Different Criticalities

FUNCTIONAL DESCRIPTION This software provides a probabilistic bound on the worst case execution time of a program. Its third version, released in March 2016, covers the case of statistically dependent execution times. Currently integrated in Rapitime (Rapita tool chain), a lighter version is under preparation for integration in FUI Waruna framework as well as in the preparation of hybrid versions to be released in 2017 as output of Capacites project.

- Participants: Liliana Cucu and Adriana Gogonel
- Contact: Liliana Cucu
- URL: <https://who.rocq.inria.fr/Liliana.Cucu/Software.html>

5.2. KPASSA

K-Periodic Asap Static Schedule Analyser

FUNCTIONAL DESCRIPTION This software is dedicated to the simulation, analysis, and static scheduling of Event/Marked Graphs, SDF and KRG extensions. A graphical interface allows to edit the Process Networks and their time annotations (latency, ...). Symbolic simulation and graph-theoretic analysis methods allow to compute and optimize static schedules, with best throughputs and minimal buffer sizes. In the case of KRG the (ultimately k-periodic) routing patterns can also be provided and transformed for optimal combination of switching and scheduling when channels are shared. KPASSA also allows for import/export of specific description formats such as UML-MARTE, to and from our other TimeSquare tool.

- Participants: Jean Vivien Millo and Robert De Simone
- Contact: Robert de Simone
- URL: <http://www-sop.inria.fr/members/Jean-Vivien.Millo/kpassa/index.php>

5.3. Lopht

Logical to Physical Time Compiler

SCIENTIFIC DESCRIPTION Lopht is a system-level compiler for embedded systems. Its input is formed of three objects:

- A functional specification in a high-level synchronous language.
- A description of the implementation platform, defining the topology of the parallel execution platform, and the capacity of its elements.
- A set of non-functional requirements, provided under the form of annotations on both functional specification and platform description.

The algorithmic core of Lopht is formed of allocation and scheduling heuristics which rely on two fundamental choices: the use of table-based static scheduling and the use of low-complexity heuristics based on list scheduling. The output of Lopht is formed of all the C code and configuration information needed to allow real deployment on the physical target platform.

FUNCTIONAL DESCRIPTION Accepted input languages for functional specifications include Heptagon and Scade v4. Lopht uses as front-end a modified version of the Heptagon compiler developed at Inria. The use of this front-end also allows the use of legacy/business C code satisfying the Heptagon calling convention.

Regarding scheduling, the originality of Lopht resides in a strong focus on classical compiler optimizations e.g. software pipelining), on novel architectural targets (many-core chips and time-triggered embedded systems), and the possibility to handle multiple, complex non-functional requirements covering real-time (release dates and deadlines possibly larger than the period, end-to-end flow constraints), ARINC 653 partitioning, the possibility to preempt or not each task, and allocation.

The output of Lopht is formed of all the C code and configuration information needed to allow compilation, linking/loading, and real-time execution on the target platform. Lopht fully automates the creation of tasks, partition, the full synthesis of C code compliant with the target API (e.g. C/APEX for ARINC 653 platforms), including communication code, and OS configuration for each computer), as well as the synthesis of communication schedules for the system

Two Lopht back-ends provide distinct input languages for platform description:

- One for distributed time-triggered architectures using ARINC 653-based processing nodes (SBCs) and Time-Triggered Ethernet networks
- One for many-core processors with support with timing predictability.

An ongoing research effort aims at providing a unified, formal platform description language allowing the unification of these back-ends.

- Participants: Dumitru Potop Butucaru, Raul Gorcitz, and Keryan Didier
- Contact: Dumitru Potop Butucaru

5.4. SAS

Simulation and Analysis of Scheduling

SCIENTIFIC DESCRIPTION The SAS (Simulation and Analysis of Scheduling) software allows the user to perform the schedulability analysis of periodic task systems in the monoprocessor case.

The main contribution of SAS, when compared to other commercial and academic softwares of the same kind, is that it takes into account the exact preemption cost between tasks during the schedulability analysis. Beside usual real-time constraints (precedence, strict periodicity, latency, etc.) and fixed-priority scheduling policies (Rate Monotonic, Deadline Monotonic, Audsley++, User priorities), SAS additionally allows to select dynamic scheduling policy algorithms such as Earliest Deadline First (EDF). The resulting schedule is displayed as a typical Gantt chart with a transient and a permanent phase, or as a disk shape called "dameid", which clearly highlights the idle slots of the processor in the permanent phase.

FUNCTIONAL DESCRIPTION The SAS software allows the user to perform the schedulability analysis of periodic task systems in the monoprocessor case.

- Participants: Daniel De Rauglaudre and Yves Sorel
- Contact: Yves Sorel
- URL: <http://pauillac.inria.fr/~ddr/sas-dameid/>

5.5. SynDEX

KEYWORDS: Embedded systems - Real time - Optimization - Distributed - Scheduling analyses

SCIENTIFIC DESCRIPTION SynDEX is a system level CAD software implementing the AAA methodology for rapid prototyping and for optimizing distributed real-time embedded applications. It is developed in OCaml.

Architectures are represented as graphical block diagrams composed of programmable (processors) and non-programmable (ASIC, FPGA) computing components, interconnected by communication media (shared memories, links and busses for message passing). In order to deal with heterogeneous architectures it may feature several components of the same kind but with different characteristics.

Two types of non-functional properties can be specified for each task of the algorithm graph. First, a period that does not depend on the hardware architecture. Second, real-time features that depend on the different types of hardware components, ranging amongst execution and data transfer time, memory, etc.. Requirements are generally constraints on deadline equal to period, latency between any pair of tasks in the algorithm graph, dependence between tasks, etc.

Exploration of alternative allocations of the algorithm onto the architecture may be performed manually and/or automatically. The latter is achieved by performing real-time multiprocessor schedulability analyses and optimization heuristics based on the minimization of temporal or resource criteria. For example while satisfying deadline and latency constraints they can minimize the total execution time (makespan) of the application onto the given architecture, as well as the amount of memory. The results of each exploration is visualized as timing diagrams simulating the distributed real-time implementation.

Finally, real-time distributed embedded code can be automatically generated for dedicated distributed real-time executives, possibly calling services of resident real-time operating systems such as Linux/RTAI or Osek for instance. These executives are deadlock-free, based on off-line scheduling policies. Dedicated executives induce minimal overhead, and are built from processor-dependent executive kernels. To this date, executive kernels are provided for: TMS320C40, PIC18F2680, i80386, MC68332, MPC555, i80C196 and Unix/Linux workstations. Executive kernels for other processors can be achieved at reasonable cost following these examples as patterns.

FUNCTIONAL DESCRIPTION Software for optimising the implementation of embedded distributed real-time applications and generating efficient and correct by construction code

- Participants: Yves Sorel and Meriem Zidouni
- URL: <http://www.syndex.org>

5.6. TimeSquare

KEYWORDS: Profil MARTE - Embedded systems - UML - IDM

SCIENTIFIC DESCRIPTION TimeSquare offers six main functionalities:

- graphical and/or textual interactive specification of logical clocks and relative constraints between them,
- definition and handling of user-defined clock constraint libraries,
- automated simulation of concurrent behavior traces respecting such constraints, using a Boolean solver for consistent trace extraction,
- call-back mechanisms for the traceability of results (animation of models, display and interaction with waveform representations, generation of sequence diagrams...).
- compilation to pure java code to enable embedding in non eclipse applications or to be integrated as a time and concurrency solver within an existing tool.
- a generation of the whole state space of a specification (if finite of course) in order to enable model checking of temporal properties on it

FUNCTIONAL DESCRIPTION TimeSquare is a software environment for the modeling and analysis of timing constraints in embedded systems. It relies specifically on the Time Model of the Marte UML profile, and more accurately on the associated Clock Constraint Specification Language (CCSL) for the expression of timing constraints.

- Participants: Frederic Mallet, and Julien Deantoni
- Contact: Frederic Mallet
- URL: <http://timesquare.inria.fr>

5.7. Vercors

KEYWORD:

- Participants: Eric Madelaine, Oleksandra Kulankhina, Jimmy Awk, Xudong Qin
- Contact: Eric Madelaine
- URL: <http://www-sop.inria.fr/oasis/Vercors>

FUNCTIONAL DESCRIPTION The Vercors tools include front-ends for specifying the architecture and behaviour of components in the form of UML diagrams. We translate these high-level specifications, into behavioural models in various formats, and we also transform these models using abstractions. In a final step, abstract models are translated into the input format for various verification toolsets. Currently we mainly use the various analysis modules of the CADP toolset.

We have achieved this year a major version of the platform frontend, named VCE-v4, that is now distributed on our website, and used by some of our partners. This version features a full chain of tools from the design of systems in the graphical component editors (VCE), the checking of static semantics correctness, the generation of a semantic model suitable for model-checking, and finally the generation of executable code for the Proactive/GCM platform. These new features, and the tool architecture, have been described in [29] and [18].

6. New Results

6.1. CCSL as a Logical Clock Calculus Algebra: expressiveness and analysis techniques

Participants: Robert de Simone, Julien Deantoni, Frédéric Mallet, Dongdong An.

CCSL is a simple, half-declarative and half-imperative language describing relations and constraints between sequences of events considered as Logical Clocks. The usage of CCSL for specification of embedded systems is powerful in that it defers the precise setting of physical timing until later implementation design phases (which may vary according to circumstances), see 3.2.

Early this year we established the universal recursive expressivity of CCSL, by encoding the dynamics of Petri Nets with inhibitor arcs in our framework (still unpublished). Those results were presented by Robert de Simone in a keynote talk at Memocode 2016. This result prompts the use of non-automatic methods for establishing actual schedules as solutions of CCSL specifications seen as schedulability constraints. Steps in that direction were made in [37].

We also considered the extension of CCSL towards stochastic modeling of potential input clocks as they emerge from the Cyber-Physical world (mixing probabilistic modeling of external events with discrete transformations by discrete cyber digital controllers). This work was initiated in [28], and should be further extended in the ongoing PhD thesis of Dongdong An.

Finally, we have also investigated to decide on specific schedules (e.g. periodic schedules) valid for a subset of CCSL. We have established a sufficient static condition for the existence of such a periodic schedule as well as a practical implementation to build such a solution [39] based on a SMT solver.

6.2. Industrial design flow for Embedded System Engineering

Participants: Julien Deantoni, Frédéric Mallet, Marie Agnes Peraldi Frati, Robert de Simone, Hui Zhao, Ales Mishchenko.

As part of the PIA LEOC Clarity collaborative project we considered the introduction of formal methods into a high-level model-based design environment for embedded systems, named CAPELLA (<https://polarsys.org/capella/>). CAPELLA is part of the Polarsys Eclipse project. It originates from Thales, and is currently being deployed in real operational divisions in a number of companies.

Our activities consisted in demonstrating how the theoretical models of Logical Time and derives Models of Computation could be used to give precise semantics and provide simulation benefits, when applied to the modeling paradigms used in CAPELLA and advanced in Clarity. In particular we focused on the connection between timing/performance properties and other kinds of non-functional properties, including model variability.

This year we focused on two main tasks:

First, we clarified and extended the notion of Modes and States in the Capella system engineering language. Specifically, a specific diagram has been introduced to deal with the system modes. The notion of mode is then used to specify different configurations of the system, mainly in terms of the active functions, their data dependencies, their deployment on the logical and physical architecture as well as the scenario to be verified in this specific mode. In consequence, the behavioural semantics of the mode diagram strongly interacts with the behavioral semantics of the other diagrams. The execution semantics was given by promoting our contributions in GEMOC and BCOoL (see 6.3).

Second, Capella proposes a consistent multi-view approach accross different engineering domains. At some step in the refinement process, these different views are extracted to a domain specific tool (like Simulink for instance). It is then required 1) to verify that the manipulation done in the domaine specific tool respect the original semantics expected by the architect, and 2) to understand the impact of the decisions made in domain specific tools on the interaction with the other views. To do so we provided a generic approach to confront the race to the behavioral semantics we formally defined in Capella. We are curretyly working on a theoretical approach to improve the overall performance of such approach.

While BCOoL and Gemoc only considers discrete models, the PhD thesis of Hui Zhao, which started in March 2016, explores a possible extension that specifically targets Cyber-Physical Systems where we different timed models combined, including both discrete and dense timed models. In this thesis, we also explore the impact of such an heterogeneous modeling framework to guarantee security and safety properties of the combined models. This is done in collaboration with Ludovic Apvrille (who is co-advisor of the thesis) from Telecom ParisTech.

6.3. Coordination of heterogeneous Models of Computation as Domain-Specific Languages

Participants: Matias Vara Larsen, Julien Deantoni, Frédéric Mallet.

Our work this on coordination of heterogeneous languages produced two major results. The first one is the development of BCOoL (Behavioral Coordination Operator Language. BCOoL is a language dedicated to the specification of coordination patterns between heterogeneous languages. It comes with a tool chain allowing the generation of the coordination given a BCOoL operator and specific models. Our second result is the development of an heterogeneous execution engine, integrated to Gemoc studio, to run conjointly different models. Both works re extensively reported in Matias Vara Larsen PhD thesis [19].

6.4. SoC multiview (meta)modeling for performance, power, and thermal aspects

Participants: Amani Khecharem, Robert de Simone, Emilien Kofman, Julien Deantoni.

In the framework of the ANR HOPE project we progressed the definition of multiview metamodels for the design of Systems-on-Chip) (SoC systems integrating performance, power and thermal aspects. The main concern was to stress regularity and commonalitbetween those views, each developed on "domains" defined as partitions of the original block diagram (clock domains, voltage domains, floorplans,...), and with finite state machine controlers setting the levels of these domains; links between distinct views are originally provided by laws of physics, but then usually identified with discrete allowed values(such as OPP, Operating Performance Points, providing the available frequency-voltage levels for processor clocks).

The corresponding methodology, named MuArch, was reported as Ameni Khacharem PhD document [16].

6.5. MoCs and novel architectures

Participants: Amine Oueslati, Robert de Simone, Albert Savary, Emilien Kofman.

In the context of the FUI Clistine project we considered the links between formal Models of Computation and parallel programming models (MPI mainly). The objective is to figure to what level an abstraction of MPI processes as concurrent communicating processes can help for the AAA design process being applied to theselection of adequate MPI communications. This topic reflects the ongoing PhD thesis of Amine Oueslati, and the engineering work of Albert Savary in the first semester.

6.6. Solving AAA constraints analytically

Participants: Emilien Kofman, Dumitru Potop Butucaru, Robert de Simone, Amine Oueslati.

We experimented on the use of SMT solvers to compute efficient mappings (bothe schedules and placement allocations) for concurrent embedded applications onto specific embedded architectures of big.LITTLE features (where alocation and migration of tasks can follow concern for low-power consumption). In fact, the work consisted greatly in a study of how the various models could be encoded to scale up, allowing the solvers to provide results in reasonable time. The results have been presented [41], [31], and will soon appear as E. Kofman PhD thesis.

6.7. Coupling SystemC and FMI for co-simulation of Cyber-Physical Systems

Participants: Stefano Centomo, Julien Deantoni, Robert de Simone.

In collaboration with Professor Davide Quaglia, from the University of Verona, we are studying the proper joint modeling of interactions between different domains involved in a cyber-physical system (CPS), and specifically between the cyber and physical parts. In our first work, realized in the context of Stefano Centomo master internship, we investigated how an event based hardware description language can be used in an emerging industry standard for co-simulation (FMI/FMU developed originally in a Modelica framework). Preliminary results were published [26], and we hope to start a PhD as follow-up of these results.

6.8. Behavioural Semantics if Open pNets

Participants: Eric Madelaine, Ludovic Henrio, Siqi Li, Min Zhang.

We have extended our preliminary work on Parameterised Networks of Automata (pNets), by looking at the behavioural semantics and at bisimulation equivalences for open pNet systems. These can be used to encode operators of various process algebras, construct of distributed or reactive system programming languages, or even parallel algorithmic skeletons, and generic distributed algorithms. As a first step, we studied the properties of a strong bisimulation equivalence based on logical hypotheses about the behaviour of process variables in the open systems. This has been published in [22], [33] and an extended version as an Inria research report [43]. We are now implementing algorithms for computing the symbolic behavioural semantics of open pNets, and checking strong bisimulation, using a SAT engine for reasoning on the hypotheses.

In order to understand better this behavioural semantics, we also have defined another version with a denotational flavour, namely using a “Universal Theory of Processes (UTP)” style. There we express the communication actions of pNets using traces of interaction events, and we were able to prove axiomatic properties of some simple (open) pNets. This was published in [32]. In the long term, it could be interesting to study the relations between the FH-bisimulation and the UTP semantics, relating both behavioural, denotational and algebraic semantics of pNets.

6.9. Behavioural semantics for GCM components

Participants: Ludovic Henrio, Oleksandra Kulankhina, Eric Madelaine.

With Ludovic Henrio (Comred/I3S) and Rabea Ameer-Boulifa (Labsoc/Telecom-Paristech), we have pursued our research on the Behavioural semantics, in terms of pNets, of the core concepts of Grid Component Model (GCM). The results are currently submitted for publication as a journal paper, under revision.

6.10. Performance analysis and optimisation of an HPC scientific application

Participants: Luis Agustin Nieto, Sid Touati.

In the context of the international Internship of Luis Agustin Nieto we conducted a large-scale experiment of source code optimization for HPC application. This work is meant to identify potential approaches that may be automatized in the future. The current use case was an application named CONVIV. CONVIV is a computer code implementing the VMFCI Method to solve the stationary Schrödinger equation for a set of distinguishable degrees of freedom (<https://svn.oca.eu/trac/conviv>). It is used in Chemistry for computing the energy levels of molecules.

This application is very computer-intensive (many hours of computation on a high performance grid computer). We have been given its source code (fortran with OpenMP), and we have been asked to analyse its performance and to optimise its execution time.

We did an extensive set of experiments for this application on many computers, and mainly on the `ci-cada.unice.fr` shared grid computer used for scientific parallel computing at UNS). We varied many parameters in our experiments:

- The number of threads was 2, 4, 6, 8, 16 threads. We also analysed the sequential code version.
- The thread affinity strategies for scheduling were: none (linux scheduler), scatter, compact.
- We repeated each experience 35 times to analyse performance stability.
- We used 2 compilers (gfortran, ifort) with -O3.
- We did a precise performance profiling using the Intel Vtune tool.

During our experiments we observed that, even with all the parameters above kept fixed, repeating the executions 35 times shows great variability between best and worst execution times (more than double in some cases). The critical-path functions remained the same for each configuration choice, including in particular specific matrix computation functions.

After investigation and experiments, we succeed in getting a spectacular performance improvement by applying the following optimisations:

- Replace one of the matrix computation function by an MKL one (highly optimised and tuned function done by Intel).
- Use the compact thread scheduling strategy (OpenMP parameter).
- By using gfortran compiler with -O3, we reduced the execution time from 18400 seconds to 820 seconds (speedup=22).
- By using the ifort compiler with -O3, we reduced the execution time from 21000 seconds to 620 seconds (speedup=33).

6.11. Formal translation validation of multi-processor real-time schedules

Participants: Keryan Didier, Dumitru Potop-Butucaru.

This research direction is mainly represented by the PhD thesis of Keryan Didier, and takes place in the framework of the ITEA3 ASSUME project. The technical focus of the ASSUME project is on formal compiler verification and on correct real-time implementation for parallel applications. The objective of this PhD thesis is to formally prove the correctness of (part of) the automatic code generation technology of Lopht, considering the respect of non-functional requirements, and in particular real-time requirements such as release dates, deadlines and periods.

During this first year of work we have:

1. Simplified the allocation and scheduling algorithms of Lopht to facilitate proof while still being able to handle the industrial use case. The resulting algorithms consider all the aspects pertaining to functional specification and non-functional requirements, but make simplifying assumptions on the execution platform (by not taking into account memory access interferences during parallel execution).
2. Developed a formally proved translation validation tool to determine the correctness of schedules produced by the algorithms at point (1). The tool is developed and proved in Coq. Coq code extraction is used to produce OCaml code that integrates in the allocation and scheduling flow.
3. Evaluated the tool on a large-scale industrial use case from Airbus (6000 Scade nodes). We demonstrated the tool to our project partners and during the ASSUME project evaluation. This evaluation showed that our scheduling and formally proved validation tools scale up to the size of large applications.

The main limitation of the current work is that it does not take into account the interferences due to concurrent memory accesses. This gives the main research direction for the next year.

We are currently writing a paper on this subject.

6.12. Lopht back-end for TTEthernet-based distributed systems

Participants: Raul Gorcitz, Dumitru Potop-Butucaru.

The global objective of this activity is a large-scale, ongoing effort to assess the possibility of automatically synthesizing full real-time implementations, including the so-called "bus frame" (the network configuration) on complex industrial platforms and for complex functional and non-functional specifications. We worked this year in the context of the post-doctoral position of Raul Gorcitz, funded by the ITEA3 ASSUME project, but also in the framework of our collaboration with CNES and Airbus DS.

The chosen platform was an industry-level evaluation platform using several Single-Board Computers (SBCs) running the VxWorks 653 OS, and connected through a Time-Triggered Ethernet (TTE) network. This platform was provided by CNES, as typical target for embedded applications. TTE is a standardized commercial communication network, on top of a switched Ethernet basis, commercialized by TTTech. TTE adds support for realtime and fault tolerant communications, allows multiple communications of mixed criticalities to share a single physical medium. This is ensured by means of dedicated hardware using a set of configuration files describing the system architecture and behavior. These configurations are synthesized by the proprietary TTEplan tool starting from a global network description file.

The main scientific difficulty was the formal modeling of the behavior of the TTE network, followed by the extension of scheduling algorithms to consider such a network. While preliminary results were obtained and published last year, we completed and demonstrated this work to our industrial partners, and we are currently writing a second paper on the subject.

6.13. Uniprocessor Real-Time Scheduling

Participants: Mehdi Mezouak, Yves Sorel, Walid Talaboulma.

In the context of the master internship of Mehdi Mezouak, we thoroughly tested the offline time triggered scheduler implemented on an ARM Cortex M4 last year. We remind that this scheduler, intended for safety critical applications, uses a scheduling table containing the instants when the scheduler will be called through interruptions triggered by a timer. This table is generated by a uniprocessor offline schedulability analysis which accounts accurately for the scheduler cost itself, and for the cost of all preemptions the data dependent tasks are subjected to. This approach allows accounting for preemptions induced by the cost of other preemptions. We implemented a time measurement system on a LPC4080 microcontroller board of NXP which includes the ARM Cortex M4 and several timers, to determine on the one hand the actual cost of the scheduler and the cost of one preemption, and on the other hand start, resume and completion times of every

task of the task sets. For the ARM Cortex M4 with a 120Mhz clock we obtained 142 cycles ($2.3 \mu s$) for the scheduler cost and 54 cycles ($0.9 \mu s$) for the cost of one preemption. We used these values for schedulability analyses we applied to various task sets. We improved the graphical tools proposed last year to draw the timing diagrams obtained during the schedulability analysis and during the real-time execution of the task set in order to compare them. For example, thanks to these measurement system and tools, we showed that this scheduler, based on a non periodic timer rather than the usual periodic one, allows the periodic execution of tasks without any jitter.

6.14. Multiprocessor Real-Time Scheduling

Participants: Mehdi Mezouak, Salah Eddine Saidi, Yves Sorel.

Always in the context of the master internship of Mehdi Mezouak, we studied the extension to multiprocessor of our offline time triggered scheduler. Since we chose the partitioned multiprocessor scheduling approach rather than the global one which is not suited to safety critical applications due to the prohibitive cost of task migrations, the uniprocessor schedulability analysis is easily extended. Indeed, the main modification consists, for every processor, in accounting for the cost of inter-processor communications and synchronizations due to data dependences when a producer task is allocated to a processor which is different from the one the corresponding consumer task is allocated to. Therefore, new scheduler calls are added to the scheduling table corresponding to instants when awaited data are available, i.e. produced and then transferred. Of course, there are as many scheduling tables, and thus schedulers, as there are processors, and these scheduling tables are supposed to share a unique global time. The implementation of this global time raises a complex problem since it is not possible to dispatch a unique physical clock to all the processors. Among various solutions, we chose to use a physical clock rather than a logical one like in the Lamport's timestamp approach since we are interested in safety critical real-time. In addition, we chose the Berkeley's algorithm based on a master-slave approach where the clock server is maintained by one of the processor of the multiprocessor. This algorithm is more robust to failures than other algorithms based on an external clock server. Finally, using the measurement system mentioned previously, we measured accurately the cost of inter-processor communications according to the number of transferred data, in the case of an ethernet network that we experimented last year to connect several LPC4080 microcontroller boards.

During the second year of the PhD thesis of Salah Eddine Saidi, we continued to study the parallelization on multi-core of FMI-based co-simulation of numerical models, that is increasingly used for the design of Cyber-Physical Systems. Such model developed according to the FMI standard is defined by a number of C functions, called "operations", for computing its variables (inputs, outputs, state) and data dependences between these variables. Each model has an associated integration step and exchanges data with the other models according to its communication step which can be larger or equal to its integration step. These models are represented by a dataflow graph of operations [35] that is compliant with the conditioned repetitive dataflow model of our AAA methodology for functional specification. Our work mainly focused on two aspects. First, we proposed a graph transformation algorithm in order to allow handling multi-rate co-simulation, i.e. where connected models have different communication steps. This algorithm is based on the concept of graph unfolding similarly to the unrolling algorithm of our AAA methodology. The new graph is represented over the hyper-step which is equal to the least common multiple of the communication steps of all the models. Each operation is repeated in the graph according to the ratio between the hyper-step and its communication step. Then, rather than adding edges connecting all the repetitions of dependent operations, specific rules are used to define the repetitions that have to be connected by edges. These rules ensure correct data exchange between the operations as requested in the context of simulation. Second, some FMI functions called to compute model variables may not be "thread-safe", i.e. they cannot be executed in parallel as they may share some resource (e.g. variables). Consequently, if two or more operations belonging to the same model are executed on different cores, a mechanism that ensures these operations are executed in strictly disjoint time intervals must be set up. We proposed an acyclic orientation heuristic to solve this problem. This heuristic adds non directed edges between the operations that belong to the same model, and then assigns directions to these edges with the aim of minimizing the critical path of the resulting graph and subject to the constraint that no cycle is generated in the graph.

6.15. Probabilistic Solutions for Hard Real-Time Systems

Participants: Adriana Gogonel, Dorin Maxim, Antoine Bertout, Tomasz Kloda, Irina Asavae, Mihail Asavae, Cristian Maxim, Walid Talaboulma, Slim Ben-Amor, Robert Davis, Liliana Cucu.

The probabilistic solutions for hard real-time systems are built under the hypothesis that worst case values and worst case execution scenarios have extremely low probability of appearance. While continuing the estimation of bounds for the worst case execution times of a program [34], [25], we have proposed the first utilisation of probabilistic description for mixed-criticality systems [42]. Our result is exploiting the heavy tails of the execution times of a program to propose efficient scheduling solutions. Moreover since the feasibility intervals [21] for a probabilistic real-time system is not formally identified, we have formulated the first feasibility reasoning for such systems [47] under fixed-priority assignment policies [20]. Another important problem for probabilistic real-time systems concerns the feasibility in presence of precedence constraints, often used by our industry partners. The introduction of precedence constraints requires the comparison of probabilistic arrivals and we showed that existing measures are not correct in this context and we proposed and proved correct new measures [24].

7. Bilateral Contracts and Grants with Industry

7.1. Bilateral Contracts with Industry

Airbus CIFRE grant This contract, started on March 2014, provides full support for the PhD thesis of Cristian Maxim. The thesis concerns the statistical timing analysis while different variability factors are taken into account. The proposed methods are built on top of existing statistical approaches while proving appropriate programs for training these methods and thus learning from the history of the execution.

8. Partnerships and Cooperations

8.1. National Initiatives

8.1.1. ANR

8.1.1.1. HOPE

Participants: Carlos Gomez Cardenas, Ameni Khecharem, Emilien Kofman, Robert de Simone.

The ANR HOPE project focused on hierarchical aspects for the high-level modeling and early estimation of power management techniques, with potential synthesis in the end if feasible. Partners were Intel, Synopsys, Magillem, UNS UMR LEAT, and ourselves.

We defined a multi-view, Model-Based design environment named MuVarch, accounting for power-level and performance of embedded hardware architectures, together with representation of abstract applications defining typical use cases for these platforms.

Started in November 2013, the project reached its completion in February 2016, while Ameni Khecharem PhD defense took place in April 2016 [16].

8.1.1.2. GeMoC

Participants: Matias Vara Larsen, Julien Deantoni, Frédéric Mallet.

This project was administratively handled by CNRS for our joint team, on the UMR I3S side. It ended September 2016. Partners were Inria (DiverSE EPC), ENSTA-Bretagne, IRIT, Obeo, Thales TRT and Supelec. The project focused on the executable modeling of heterogeneous systems using Models of Computation and Communication described using meta-languages. Specifically, the operational semantics of languages were equipped with precise timely constraints specified in CCSL. There were many outputs from the project but, from AOSTE perspective, we essentially developed MoCCML, an extension of CCSL with constraint automata (already integrated to TimeSquare) and BCool, a language dedicated to coordination apattern specification, which is described as part of Matias Vara-Larsen PhD thesis[19]. All the development realized in this project will end up as the first official eclipse research consortium.

8.1.1.3. *FUI CLISTINE*

Participants: Robert de Simone, Amin Oueslati, Emilien Kofman.

This project was started in Oct 2013, and provides PhD funding for Amine Oueslati. Partners are SynergieCAD (coordinator), Avantis, Optis, and the two EPIs Aoste and Nachos. The goal is to study the feasibility of building a low-cost, low-power "supercomputer", reusing ideas from SoC design, but this time with out-of-chip network "on-board", and out-of-the-shelf processor elements organized as an array. The network itself should be time predictable and highly parallel (far more than PCI-e for instance). We started a thorough classification of parallel program types (known as "Dwarfs" in the literature), to provide benchmarks and evaluate the platform design options.

8.1.1.4. *FUI Waruna*

Participants: Liliana Cucu, Adriana Gogonel, Walid Talaboulma, Dorin Maxim.

This recent project was started in September 2015. It targets the creation of a framework allowing to connect different existing methods while enriching the description with Waruna results. This framework allows timing analyses for different application domains like avionics, railways, medical, aerospace, automotive, etc.

8.1.2. *Investissements d'Avenir*

8.1.2.1. *DEPARTS*

Participants: Liliana Cucu-Grosjean, Adriana Gogonel, Walid Talaboulma.

This project is funded by the BGLE Call (*Briques Logicielles pour le Logiciel Embarqué*) of the national support programme *Investissements d'Avenir*. Formally started on October 1st, 2012 with the kick-off meeting held on April, 2013 for administrative reasons. Research will target solutions for probabilistic component-based models, and a Ph.D. thesis should start at latest on September 2015. The goal is to unify in a common framework probabilistic scheduling techniques with compositional assume/guarantee contracts that have different levels of criticality.

8.1.2.2. *CLARITY*

Participants: Frédéric Mallet, Julien Deantoni, Ales Mishchenko, Robert de Simone, Marie Agnès Peraldi-Frati.

This project is funded by the LEOC Call (*Logiciel Embarqué et Objets Connectés*) of the national support programme *Investissements d'Avenir*. It was started in September 2014, and a kick-off meeting was held on October 9th. Partners are: Thales (several divisions), Airbus, Areva, Altran, All4Tec, Artal, the Eclipse Foundation, Scilab Enterprises, CESAMES, U. Rennes, and Inria. The purpose of the project is to develop and promote an open-source version of the ARCADIA Melody system design environment from Thales, renamed CAPPELLA for that purpose.

Our technical contributions to the project achievement are described in subsection 6.2.

8.1.2.3. *Capacites*

Participants: Liliana Cucu-Grosjean, Dumitru Potop-Butucaru, Yves Sorel, Walid Talaboulma.

This project is funded by the LEOC Call (*Logiciel Embarqué et Objets Connectés*) of the national support programme *Investissements d'Avenir*. It has started on November 1st, 2014 with the kick-off meeting held on November, 12th 2014. The project coordinator is Kalray, and the objective of the project is to study the relevance of Kalray-style MPPA processor array for real-time computation in the avionic domain (with partners such as Airbus for instance). The post-doc of Mihail Asavoae and the PhD of Walid Talaboulma are funded on this contract.

8.2. European Initiatives

8.2.1. Collaborations in European Programs, Except FP7 & H2020

8.2.1.1. ASSUME

Participants: Dumitru Potop-Butucaru, Keryan Didier, Liliana Cucu.

This project is funded by the ITEA3 program. It has started on September 1st 2015. Project coordinator is Daimler. ASSUME has funded the (now completed) post-doc of Raul Gorcitz, and funds the PhD thesis of Keryan Didier.

Future mobility solutions will increasingly rely on smart components that continuously monitor the environment and assume more and more responsibility for a convenient, safe and reliable operation. Currently the single most important roadblock for this market is the ability to come up with an affordable, safe multi-core development methodology that allows industry to deliver trustworthy new functions at competitive prices. ASSUME will provide a seamless engineering methodology, which addresses this roadblock on the constructive and analytic side.

In this project, most our effort goes to work package "Synthesis of Predictable Concurrent Systems", which we lead. Main scientific results of our work in this project have been presented in sections 6.11 and 6.12. In addition, we closely interacted with our industrial partners to determine their needs, and developed importer tools for their internal formalisms, including Scade v4 and internal formalisms used at Airbus (all importers were developed jointly with EPI PARKAS). This work also resulted in proposals to Airbus on the specification of certain non-functional properties (e.g. the atomic groups of operations that cannot be split during allocation and scheduling). By applying our prototype tools, we have also determined that the use case has significant potential parallelism and will achieve significant speedups through execution on the chosen target architecture (the many-core Kalray MPPA256).

8.3. International Initiatives

8.3.1. FM4CPS

Title: Formal Models and tools for Cyber-Physical Systems

International Partner (Institution - Laboratory - Researcher):

ECNU (China) - Artificial Intelligence Lab - Jifeng He

Start year: 2015

See also: <https://project.inria.fr/fm4cps/>

Cyber-Physical Systems (CPS) and the connected Internet of Things (IoT) are inherently heterogeneous systems, with ("cyber") computer digital parts interacting with their physical sensible environment, under user requirements for functional and temporal correctness. Thus, design of such systems as a whole requires a diversity of models, and the behavior orchestration between such models must be carefully defined and analyzed.

FM4CPS will address several facets of Formal Model-Driven Engineering for Cyber-Physical Systems and Internet of Things. The design of such large heterogeneous systems calls for hybrid modeling, and the combination of classes of models, most previously well-established in their own restricted area: Formal Models of Computations drawn from Concurrency Theory for the "cyber" discrete processors, timed extension and continuous behaviors for physical environments, requirement

models and user constraints extended to non-functional aspects, new challenges for designing and analyzing large and highly dynamic communicating software entities. Orchestration and comparison of models, with their expressive power vs. their decidable aspects, shall be considered with the point of view of hybrid/heterogeneous modeling here. Main aspects are the various timing or quantitative structure extensions relying for instance on a hybrid logical clock model for the orchestration of underlying components.

The associated team aims at various level of research, from formal models, semantics, or complexity, to experimental tools development. This will start for example on one side with building a formal orchestration model for CPSs, based on an hybrid clock model that combine discrete and physical time, synchronous and asynchronous computations or communications. Another goal will be the study of expressiveness and decidability for CPS, based on dedicated sub-families of well-structured push-down systems, addressing both unbounded communication and time-sensitive models.

Beyond their own expertise in this field, the partners will build on the results of previous cooperations in the context of the Liama projects Hades and Tempo, and the associated team DAESD. The current proposal widely broadens the domain of collaboration, and with the inclusion, for the first time, of Jiao Tong University. We expect this is the first step towards the extension of LIAMA in Shanghai with the strengthening of the involvement of E.C.N.U., and the contribution of new top notch universities such as Jiaotong.

8.3.2. Inria International Partners

8.3.2.1. Declared Inria International Partners

We have signed an agreement with the University of Verona, which covers joint activities (see section 6.7, together with the housing of interns.

8.4. International Research Visitors

8.4.1. Visits of International Scientists

8.4.1.1. Internships

Nieto Luis Agustin

Date: Sep 2015 - Feb 2016

Institution: Universidad de Buenos Aires (Argentina)

9. Dissemination

9.1. Promoting Scientific Activities

9.1.1. Scientific Events Organisation

9.1.1.1. General Chair, Scientific Chair

Eric Madelaine is General Chair of the 11th International Symposium on Theoretical Aspects of Software Engineering (TASE'17), and Steering Committee Chair of the International Symposium on Formal Aspects of Components Systems (FACS 2017)

Liliana Cucu-Grosjen and Rob Davis are Steering Committee members of 3 conferences (RTSS, RTAS and RTNS) and 2 workshops (RTSOPS and WMC)

Julien Deantoni was chair of the 4th GEMOC workshop, held in conjunction with the 19th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems

9.1.2. Scientific Events Selection

9.1.2.1. Chair of Conference Program Committees

Robert de Simone will be PC Chair for the forthcoming EMSOFT 2017 conference edition.

Frédéric Mallet will be PC Chair for the forthcoming TASE 2017 conference edition.

9.1.2.2. Member of the Conference Program Committees

Robert de Simone: EmSoft 2016, FDL 2016.

Dumitru Potop-Butucaru: RTNS 2016, ACSD 2016

Yves Sorel: DASIP 2016, EMSOFT 2016

Julien Deantoni: MODELS 2016, CAL 2016, GEMOC 2016, EXE 2016 DSD 2016, MOMO 2016.

Liliana Cucu-Grosjean: RTAS2016, SIES 2016, RTNS2016

Frédéric Mallet: DATE 2016, DSD 2016, ERTS 2016, ICTERI 2016.

9.1.3. Journal

9.1.3.1. Member of the Editorial Boards

Yves Sorel: DASIP 2016, EMSOFT 2016

9.1.4. Invited Talks

Robert de Simone was invited Keynote Speaker at the international conference MeMoCode 2016 in Kanpur (India)

9.1.5. Leadership within the Scientific Community

Eric Madelaine and Frédéric Mallet are Council Members of the International Joint Laboratory of Trustworthy Software, Ministry of Education, China.

9.1.6. Scientific Expertise

Yves Sorel: Steering Committees of System Design and Development Tools Group of Systematic Paris-Region Cluster, and of Technologies and Tools Program of SystemX Institute for Technological Research (IRT)

9.1.7. Research Administration

Robert de Simone is Scientific Correspondant for the Inria/Safran collaboration programme, and (starting 2017) Deputy Director of the EDSTIC Doctoral School of Université Côte d'Azur.

Liliana Cucu-Grosjean is elected member of Inria Evaluation Commission.

9.2. Teaching - Supervision - Juries

9.2.1. Teaching

Master: Robert de Simone, Models of Computation for Networks-on-Chips (MoCs for NoCs), 36h, M2 International, UNS.

Master: Robert de Simone, Functional and Temporal Correctness, 36h, M1 International, UNS.

Master: Yves Sorel, Optimization of distributed real-time embedded systems, 36H, M2, University of Paris Sud

Master: Yves Sorel, Correct by construction design of reactive systems, 18H, M2, ESIEE Engineering School, Noisy-Le-Grand

Master : Julien Deantoni, Systèmes embarqués et Ambient, 10h, M2, Polytech'Nice, France.

Master : Julien Deantoni, Langage C++, 88h, M1, Polytech'Nice, France.

Master : Julien Deantoni, Finite State Machines, 24h, M1, Polytech'Nice, France.

Master : Julien Deantoni, Internship Management, 20h, M2, Polytech'Nice, France.

Master: Dumitru Potop Butucaru, Une approche synchrone des systèmes embarqués temps réel, 12h, M1, EPITA Paris

Master: Dumitru Potop Butucaru and Thomas Carle, L'approche synchrone de la construction des systèmes embarqués temps réel, 12h, M2, Polytech Paris UPMC.

Licence: Laurent George, Java and Shell programming 48h, L1, IUT RT UPEC, France

Master: Laurent George, Distributed Real-Time Systems, 24h, M2, UPEC, France

Licence : Marie-Agnes Peraldi-Frati, Algorithms and programming 60h, L1, UNS Institute of technology.

Licence : Marie-Agnes Peraldi-Frati, System and Networks administration 80h, L2, UNS Institute of technology .

Licence : Marie-Agnes Peraldi-Frati, Web Programming 50 h, L2, UNS Institute of technology.

Licence: Frédéric Mallet, Conception Orientée Objet, 45h, L3, UNS.

Licence: Frédéric Mallet, Programmation Orientée Objet, 45h, L3, UNS.

Master: Frédéric Mallet, Programmation Avancée et Design Patterns, 45h, M1, UNS.

Master: Frédéric Mallet, Vérification temporelle et fonctionnelle, 24h, M1, UNS.

Master: Frédéric Mallet, Model-Driven Engineering, 24h, M1, UNS.

Master: Liliana Cucu, Distributed Databases and Statistics in Computer Science, 64h, U. Dunarea de Jos, Romania (Invited Professor)

Master: Dumitru Potop Butucaru, Une approche synchrone des systèmes embarqués temps réel, 12h, M1, EPITA Paris

9.2.2. Supervision

PhD: Matias Vara-Larsen, *Toward a formal and hierarchical timed model for concurrent heterogeneous model*, UNS, defended April 2016, supervised by Frédéric Mallet, co-supervised by Julien Deantoni.

PhD in progress: Ameni Khecharem, *High-Level modeling of hierarchical power management policies in SoCs*, UNS, defended May 2016, supervised by Robert de Simone.

PhD in progress: Emilien Kofman, *Conception Haut Niveau Low Power d'objets mobiles communicants*, UNS, started Oct 2013, supervised by Robert de Simone, co-supervised by François Verdier (UMR CNRS/UNS LEAT).

PhD in progress: Amin Oueslati, *Modélisation conjointe d'applications et d'architectures parallèles embarqués en pratique*, UNS, started Jan 2014, supervised by Robert de Simone

PhD in progress: Yuanrui Zhang, ECNU-SEI/China, started Sep 2015, co-supervised by Frederic Mallet (joint supervision with Pr. Chen Yixiang(ECNU)).

PhD in progress: Hui (Vincent) Zhao, UNS, started February 2016, supervised by Frédéric Mallet, co-supervised by Ludovic Apvrille (Telecom ParisTech)

PhD in progress: Dongdong An, ECNU-SEI/China, started November 2016, co-supervised by R. de Simone, supervised by Jing Liu (ECNU).

PhD in progress: Cristian Maxim, *End to end constraints using probabilistic approaches*, UPMC, started on March 2014, supervised by Liliana Cucu

PhD in progress: Walid Talaboulma, *Probabilistic timing analysis in presence of dependences*, UPMC, started on November 2015, co-supervised by Liliana Cucu and Adriana Gogonel

PhD in progress: Salah Edinne Saidi, *Distributed real-time scheduling for the co-simulation of several control models*, University of UMPC-Paris-Sorbonne, started December 2014, co-supervised by Nicolas Pernet (IFPEN) and Yves Sorel.

PhD in progress: Keryan Didier, *Formal certification of real-time implementations*, Université Pierre et Marie Curie/EDITE, started November 2015, supervised by Dumitru Potop Butucaru.

PhD: Oleksandra Kulankhina, *A framework for rigorous development of distributed components: formalisation and tools*, UNS, defended October 2016, supervised by Eric Madelaine, co-supervised by Ludovic Henrio (UMR CNRS/UNS I3S).

PhD in progress: Salah Edinne Saidi, *Distributed real-time scheduling for the co-simulation of several control models*, University of UMPC-Paris-Sorbonne, started December 2014, co-supervised by Nicolas Pernet (IFPEN) and Yves Sorel.

PhD in progress: Keryan Didier, *Formal certification of real-time implementations*, Université Pierre et Marie Curie/EDITE, started November 2015, supervised by Dumitru Potop Butucaru.

PhD: Oleksandra Kulankhina, *A framework for rigorous development of distributed components: formalisation and tools*, UNS, defended October 2016, supervised by Eric Madelaine, co-supervised by Ludovic Henrio (UMR CNRS/UNS I3S).

PhD: Vincent Kherbache, *Ordonnancement des migrations à chaud de machines virtuelles*, UNS, defended December 2016, supervised by Eric Madelaine, co-supervised by Fabien Hermenier (UMR CNRS/UNS I3S).

9.2.3. Juries

Robert de Simone: reviewer for the HDR of Xavier Thirioux (ENSIEHT, Sept. 2016)

Dumitru Potop Butucaru: PhD reviewer for Pierre Guillou - Ecole des Mines de Paris. Nov. 2016

Julien Deantoni: PhD reviewer for Florent Latombe (ENSIEHT)

Liliana Cucu-Grosjean: PhD reviewer for Guillaume Phavorin (ENSMA Poitiers, September 2016)

Liliana Cucu-Grosjean: PhD jury member for Abhilash Thekkilakattil (University of Ma'lardalen, May 2016)

9.3. Popularization

Liliana Cucu-Grosjean has supervised the video production of a popularization video regarding the outcomes of the PROXIMA project. The video has been made available on Inria channels and all PROXIMA partners.

10. Bibliography

Major publications by the team in recent years

- [1] C. ANDRÉ. *Syntax and Semantics of the Clock Constraint Specification Language (CCSL)*, Inria, 05 2009, n^o RR-6925, 37 p. , <http://hal.inria.fr/inria-00384077/en/>

-
- [2] C. ANDRÉ, J. DEANTONI, F. MALLET, R. DE SIMONE. *The Time Model of Logical Clocks available in the OMG MARTE profile*, in "Synthesis of Embedded Software: Frameworks and Methodologies for Correctness by Construction", S. K. SHUKLA, J.-P. TALPIN (editors), Springer Science+Business Media, LLC 2010, Jul 2010, pp. 201–227, Chapter 7
- [3] C. ANDRÉ, F. MALLET, R. DE SIMONE. *Modeling Time(s)*, in "MoDELS'2007 10th Intern. Conf. on Model Driven Engineering Languages and Systems", 2007
- [4] A. BENVENISTE, P. CASPI, S. EDWARDS, N. HALBWACHS, P. LE GUERNIC, R. DE SIMONE. *Synchronous Languages Twelve Years Later*, in "Proceedings of the IEEE", January 2003
- [5] J. BOUCARON, A. COADOU, R. DE SIMONE. *Formal Modeling of Embedded Systems with Explicit Schedules and Routes*, in "Synthesis of Embedded Software: Frameworks and Methodologies for Correctness by Construction", S. K. SHUKLA, J.-P. TALPIN (editors), Springer Science+Business Media, LLC 2010, Jul 2010, pp. 41–78, Chapter 2
- [6] J. BOUCARON, R. DE SIMONE, J.-V. MILLO. *Latency-insensitive design and central repetitive scheduling*, in "MEMOCODE", 2006, pp. 175-183, <http://dx.doi.org/10.1109/MEMCOD.2006.1695923>
- [7] L. CUCU-GROSJEAN, N. PERNET, Y. SOREL. *Periodic real-time scheduling: from deadline-based model to latency-based model*, in "Annals of Operations Research", 2007, <http://www-rocq.inria.fr/syndex/publications/pubs/aor07/aor07.pdf>
- [8] T. GRANDPIERRE, C. LAVARENNE, Y. SOREL. *Optimized Rapid Prototyping For Real-Time Embedded Heterogeneous multiprocessors*, in "Proceedings of 7th International Workshop on Hardware/Software Co-Design, CODES'99", 1999
- [9] T. GRANDPIERRE, Y. SOREL. *From Algorithm and Architecture Specification to Automatic Generation of Distributed Real-Time Executives: a Seamless Flow of Graphs Transformations*, in "Proceedings of First ACM and IEEE International Conference on Formal Methods and Models for Codesign, MEMOCODE'03", Mont Saint-Michel, France, June 2003, <http://www-rocq.inria.fr/syndex/publications/pubs/memocode03/memocode03.pdf>
- [10] P. MEUMEU YOMSI, Y. SOREL. *Extending Rate Monotonic Analysis with Exact Cost of Preemptions for Hard Real-Time Systems*, in "Proceedings of 19th Euromicro Conference on Real-Time Systems, ECRTS'07", Pisa, Italy, July 2007, <http://www-rocq.inria.fr/syndex/publications/pubs/ecrts07/ecrts07.pdf>
- [11] D. POTOP-BUTUCARU, BENOÎT. CAILLAUD. *Correct-by-Construction Asynchronous Implementation of Modular Synchronous Specifications*, in "Fundam. Inf.", January 2007, vol. 78, pp. 131–159, <http://portal.acm.org/citation.cfm?id=1366007.1366013>
- [12] D. POTOP-BUTUCARU, R. DE SIMONE, Y. SOREL. *From Synchronous Specifications to Statically-Scheduled Hard Real-Time Implementations*, in "Synthesis of Embedded Software: Frameworks and Methodologies for Correctness by Construction", S. K. SHUKLA, J.-P. TALPIN (editors), Springer Science+Business Media, LLC 2010, Jul 2010, pp. 229–262, Chapter 8
- [13] D. POTOP-BUTUCARU, S. EDWARDS, G. BERRY. *Compiling Esterel*, Springer, 2007

- [14] D. POTOP-BUTUCARU, R. DE SIMONE. *Optimizations for Faster Execution of Esterel Programs*, in "MEM-OCODE'03", 2003
- [15] R. DE SIMONE, D. POTOP-BUTUCARU, JEAN-PIERRE. TALPIN. *The Synchronous Hypothesis and Synchronous Languages*, in "Embedded Systems Handbook", CRC Press, 2005, chap. 8

Publications of the year

Doctoral Dissertations and Habilitation Theses

- [16] A. KHECHAREM. *MuVARCH : a (meta) modeling approach for multi-view representation of heterogeneous embedded architectures*, Université Nice Sophia Antipolis, May 2016, <https://tel.archives-ouvertes.fr/tel-01361473>
- [17] V. KHERBACHE. *Live-migrations scheduling of virtual machines*, Université Côte d'Azur, December 2016, <https://hal.inria.fr/tel-01419310>
- [18] O. KULANKHINA. *A framework for rigorous development of distributed components: formalisation and tools*, Université Nice Sophia Antipolis, October 2016, <https://hal.inria.fr/tel-01419298>
- [19] M. VARA LARSEN. *BCool : the Behavioral Coordination Operator Language*, Université Nice Sophia Antipolis, April 2016, <https://tel.archives-ouvertes.fr/tel-01302875>

Articles in International Peer-Reviewed Journals

- [20] R. I. DAVIS, L. CUCU-GROSJEAN, M. BERTOOGNA, A. BURNS. *A review of priority assignment in real-time systems*, in "Journal of Systems Architecture", April 2016, vol. 65, pp. 64 - 82, <https://hal.inria.fr/hal-01419694>
- [21] G. JOËL, E. GROLLEAU, L. CUCU-GROSJEAN. *Periodicity of real-time schedules for dependent periodic tasks on identical multiprocessor platforms*, in "Real-Time Systems", November 2016, vol. 52, n^o 6, pp. 808–832 [DOI : 10.1007/s11241-016-9256-1], <https://hal.inria.fr/hal-01419704>
- [22] E. MADELAINE, M. ZHANG. *Towards a bisimulation theory for open synchronized networks of automata*, in "Science China Information Sciences", May 2016, vol. 59, n^o 5, 3 p. [DOI : 10.1007/s11432-016-5557-1], <https://hal.inria.fr/hal-01417652>

Invited Conferences

- [23] J. DEANTONI. *Modeling the Behavioral Semantics of Heterogeneous Languages and their Coordination*, in "Architecture Centric Virtual Integration (ACVI)", Venice, Italy, Julien Delange and Jerome Hugues and Peter Feiler, April 2016, <https://hal.inria.fr/hal-01291299>

International Conferences with Proceedings

- [24] S. BEN-AMOR, D. MAXIM, L. CUCU-GROSJEAN. *Schedulability analysis of dependent probabilistic real-time tasks*, in "the 24th International Conference on Real-Time Networks and Systems", Brest, France, November 2016, pp. 99-107 [DOI : 10.1145/2997465.2997499], <https://hal.inria.fr/hal-01419741>
- [25] F. J. CAZORLA, J. ABELLA, J. ANDERSSON, T. VARDANEGA, F. VATRINET, I. BATE, I. BROSTER, M. AZKARATE-ASKASUA, F. WARTEL, L. CUCU-GROSJEAN, C. FABRICE, G. FARRALL, A. GOGONEL,

- A. GIANARRO, B. TRIQUET, H. CARLES, C. LO, C. MAXIM, D. MORALES, E. QUINONES, E. MEZZETTI, L. KOSMIDIS, I. AGIRRE, M. FERNANDEZ, M. SLIJEPCEVIC, C. PHILIPPA, W. TAL-ABOULMA. *PROXIMA: Improving Measurement-Based Timing Analysis through Randomisation and Probabilistic Analysis*, in "2016 Euromicro Conference on Digital System Design", Limassol, France, August 2016 [DOI : 10.1109/DSD.2016.22], <https://hal.inria.fr/hal-01419735>
- [26] S. CENTOMO, J. DEANTONI, R. DE SIMONE. *Using SystemC Cyber Models in an FMI Co-Simulation Environment: Results and Proposed FMI Enhancements*, in "19th Euromicro Conference on Digital System Design 31 August - 2 September 2016", Limassol, Cyprus, 19th Euromicro Conference on Digital System Design, August 2016, vol. 19 [DOI : 10.1109/DSD.2016.86], <https://hal.inria.fr/hal-01358702>
- [27] B. COMBEMALE, C. BRUN, J. CHAMPEAU, X. CRÉGUT, J. DEANTONI, J. LE NOIR. *A Tool-Supported Approach for Concurrent Execution of Heterogeneous Models*, in "8th European Congress on Embedded Real Time Software and Systems (ERTS 2016)", Toulouse, France, 2016, <https://hal.inria.fr/hal-01258358>
- [28] D. DU, P. HUANG, K. JIANG, F. MALLET, M. YANG. *MARTE/pCCSL: Modeling and Refining Stochastic Behaviors of CPSs with Probabilistic Logical Clocks*, in "FACS 2016 - The 13th International Conference on Formal Aspects of Component Software", Besançon, France, October 2016, Best Paper Award, <https://hal.inria.fr/hal-01394769>
- [29] L. HENRIO, O. KULANKHINA, S. LI, E. MADELAINE. *Integrated Environment for Verifying and Running Distributed Components*, in "Fundamental Approaches to Software Engineering", Eindhoven, Netherlands, P. STEVENS, A. WĄSOWSKI (editors), Fundamental Approaches to Software Engineering, Perdita Stevens and Andrzej Wąsowski, April 2016, vol. 9633, pp. 66-83 [DOI : 10.1007/978-3-662-49665-7_5], <https://hal.inria.fr/hal-01303557>
- [30] J. KIENZLE, G. MUSSBACHER, O. ALAM, M. SCHÖTTLE, N. BELLOIR, P. COLLET, B. COMBEMALE, J. DEANTONI, J. KLEIN, B. RUMPE. *VCU: The Three Dimensions of Reuse*, in "The 15th International Conference on Software Reuse (ICSR-15)", Limassol, Cyprus, The 15th International Conference on Software Reuse, 2016, <https://hal.inria.fr/hal-01287720>
- [31] E. KOFMAN, R. DE SIMONE. *A formal approach to the mapping of tasks on an heterogenous multicore, energy-aware architecture*, in "MEMOCODE'16 - 14th ACM-IEEE International Conference on Formal Methods and Models for System Design", Kanpur, India, November 2016, <https://hal.inria.fr/hal-01412790>
- [32] S. LI, E. MADELAINE. *A Denotational Semantics for Parameterised Networks of Synchronised Automata*, in "The 6th International Symposium on Unifying Theories of Programming", Reykjavik, Iceland, Universite de reykjavik, June 2016, 20 p. , <https://hal.inria.fr/hal-01417662>
- [33] E. MADELAINE, L. HENRIO, M. ZHANG. *A Theory for the Composition of Concurrent Processes* , in "FORTE", Heraklion, Greece, June 2016, <https://hal.inria.fr/hal-01344364>
- [34] C. MAXIM, A. GOGONEL, I. ASAVOAE, M. ASAVOAE, L. CUCU-GROSJEAN, W. TALABOULMA. *Reproducibility and representativity - mandatory properties for the compositionality of measurement-based WCET estimation approaches*, in "The 9th International Workshop on Compositional Theory and Technology for Real-Time Embedded System (CRTS2016)", Porto, Portugal, December 2016, <https://hal.inria.fr/hal-01421280>

- [35] S. E. SAIDI, N. PERNET, Y. SOREL, A. B. BEN KHALED. *Acceleration of FMU Co-Simulation On Multi-core Architectures*, in "Japanese Modelica Conference", Tokyo, Japan, MODELICA ASSOCIATION, LINKÖPING UNIVERSITY ELECTRONIC PRESS (editors), Proceedings of first Japanese Modelica Conference, May 2016, vol. 124, pp. 106 - 112 [DOI : 10.3384/ECP16124106], <https://hal.inria.fr/hal-01400033>
- [36] J. WORMS, S. TOUATI. *Going beyond mean and median programs performances*, in "IEEE MCSoc 2016 : IEEE 10th International Symposium on Embedded Multicore/Many-core Systems-on-Chip", Lyon, France, September 2016, <https://hal.inria.fr/hal-01371467>
- [37] Q. XU, J. DEANTONI, R. DE SIMONE. *Divergence Detection for CCSL Specification via Clock Causality Chain*, in "Symposium on Dependable Software Engineering Theories, Tools and Applications", Beijing, China, Programming and Software Engineering, Springer International Publishing, November 2016, vol. Dependable Software Engineering: Theories, Tools, and Applications, n^o 9984, <https://hal.inria.fr/hal-01372694>
- [38] D. YUE, V. JOLOBOFF, F. MALLET. *Flexible Runtime Verification Based On Logical Clock Constraints*, in "Forum on specification & Design Languages", Bremen, Germany, September 2016, <https://hal.inria.fr/hal-01421890>
- [39] M. ZHANG, F. MALLET, H. ZHU. *An SMT-Based Approach to the Formal Analysis of MARTE/CCSL*, in "Formal Methods and Software Engineering", Tokyo, Japan, Lecture Notes in Computer Science, Springer, November 2016, vol. 10009, pp. 433-449 [DOI : 10.1007/978-3-319-47846-3_27], <https://hal.inria.fr/hal-01394677>

Conferences without Proceedings

- [40] A. COHEN, V. PERRELLE, D. POTOPO-BUTUCARU, M. POUZET, E. SOUBIRAN, Z. ZHANG. *Hard Real Time and Mixed Time Criticality on Off-The-Shelf Embedded Multi-Cores*, in "8th European Congress on Embedded Real Time Software and Systems (ERTS 2016)", Toulouse, France, January 2016, <https://hal.archives-ouvertes.fr/hal-01259157>
- [41] E. KOFMAN, R. DE SIMONE, A. KHECHAREM. *Multicore SMT scheduling of periodic task systems with energy minimization*, in "Workshop on Highly-Reliable Power-Efficient Embedded Designs", Barcelone, Spain, March 2016, <https://hal.inria.fr/hal-01282264>
- [42] D. I. MAXIM, R. DAVIS, L. I. CUCU-GROSJEAN, A. EASWARAN. *Probabilistic Analysis for Mixed Criticality Scheduling with SMC and AMC*, in "WMC 2016", Porto, Portugal, November 2016, <https://hal.archives-ouvertes.fr/hal-01416310>

Research Reports

- [43] L. HENRIO, E. MADELAINE, M. ZHANG. *A Theory for the Composition of Concurrent Processes -Extended version*, Inria Sophia Antipolis - I3S, April 2016, n^o RR-8898, 23 p. , <https://hal.inria.fr/hal-01299562>
- [44] A. MEHMOOD KHAN, F. MALLET, M. RASHID. *Natural Interpretation of UML/MARTE Diagrams for System Requirements Specification*, I3S ; Inria, April 2016, n^o RR-8909, <https://hal.inria.fr/hal-01309604>
- [45] J. WORMS, S. TOUATI. *Parametric and Non-Parametric Statistics for Program Performance Analysis and Comparison*, Inria Sophia Antipolis - I3S ; Université Nice Sophia Antipolis ; Université Versailles Saint Quentin en Yvelines ; Laboratoire de mathématiques de Versailles, November 2016, n^o RR-8875, 70 p. , <https://hal.inria.fr/hal-01286112>

Patents and standards

- [46] L. HOGIE, M. SYSKA, N. CHLEQ. *BigGraphs: distributed graph computing*, September 2016, n° IDDN.FR.001.410005.000.S.P.2015.000.31235, <https://hal.archives-ouvertes.fr/hal-01360649>

Other Publications

- [47] A. BERTOUD, D. MAXIM, L. CUCU-GROSJEAN. *Average Probabilistic Response Time Analysis of Tasks with Multiple Probabilistic Parameters*, December 2016, Work in Progress of the 22nd IEEE Real-Time Systems Symposium (RTSS2016), Poster, <https://hal.inria.fr/hal-01421276>

References in notes

- [48] F. BACCELLI, G. COHEN, GEERT JAN. OLSDER, JEAN-PIERRE. QUADRAT. *Synchronization and Linearity: an algebra for discrete event systems*, John Wiley & Sons, 1992, <http://cermics.enpc.fr/~cohen-g/SED/book-online.html>
- [49] A. BENVENISTE, G. BERRY. *The Synchronous Approach to Reactive and Real-Time Systems*, in "Proceedings of the IEEE", September 1991, vol. 79, n° 9, pp. 1270-1282
- [50] J. CARLIER, P. CHRÉTIENNE. *Problèmes d'ordonnancement*, Masson, 1988
- [51] L. CARLONI, K. MCMILLAN, A. SANGIOVANNI-VINCENTELLI. *Theory of Latency-Insensitive Design*, in "IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems", 2001
- [52] A. COHEN, M. DURANTON, C. EISENBEIS, C. PAGETTI, F. PLATEAU, M. POUZET. *N-Synchronous Kahn Networks: a Relaxed Model of Synchrony for Real-Time Systems*, in "ACM International Conference on Principles of Programming Languages (POPL'06)", Charleston, South Carolina, USA, January 2006
- [53] J.B. DENNIS. *First Version of a Dataflow Procedure Language*, in "Lecture Notes in Computer Science", Springer-Verlag, 1975, vol. 19, pp. 362-376
- [54] S. EDWARDS. *Languages for Digital Embedded Systems*, Kluwer, 2000
- [55] N. HALBWACHS. *Synchronous Programming of Reactive Systems*, in "Computer Aided Verification", 1998, pp. 1-16, <http://www-verimag.imag.fr/~halbwach/newbook.pdf>
- [56] EDWARD A. LEE, D. G. MESSERSCHMITT. *Static Scheduling of Synchronous Data Flow Programs for Digital Signal Processing*, in "IEEE Trans. Computers", 1987
- [57] C.L. LIU, J.W. LAYLAND. *Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment*, in "Journal of the ACM", 1973