IN PARTNERSHIP WITH:
CNRS

Université Nice - Sophia Antipolis

Activity Report 2011

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 - Rocquencourt

THEME
Embedded and Real Time Systems

Table of contents

1. Members 1
2. Overall Objectives 2
2.1. Embedded System Design 2
2.2. Highlights 2
3. Scientific Foundations 2
3.1. Models of Computation and Communication (MoCCs) 2
3.1.1. K-periodic static scheduling and routing in Process Networks 3
3.1.2. Endochrony and GALS implementation of conflict-free polychronous programs 3
3.2. Logical Time in Model-Driven Embedded System Design 3
3.3. The AAA (Algorithm-Architecture Adequation) methodology and Real-Time Scheduling 4
3.3.1. Algorithm-Architecture Adequation 4
3.3.2. Distributed Real-Time Scheduling and Optimization 5
4. Application Domains 6
4.1. Multicore System-on-Chip design 6
4.2. Automotive and avionic embedded systems 6
5. Software 7
5.1. TimeSquare 7
5.2. K-Passa 7
5.3. SynDEx 8
5.4. SAS 8
6. New Results 9
6.1. Logical time in Model-Based embedded design 9
6.2. Model-Based approaches to SoC design 10
6.3. Process Network analysis 10
6.4. Correct and efficient implementation of polychronous formalisms 11
6.5. Uniprocessor Real-time Scheduling 12
6.5.1. Strict periodic harmonic tasks 12
6.5.2. Combination of strict periodic and sporadic tasks 12
6.5.3. Exact cost of RTOS 12
6.5.4. Formal proofs of real-time scheduling theorems 12
6.6. Multiprocessor Real-time Scheduling 13
6.6.1. Partitioned scheduling with exact RTOS cost 13
6.6.2. Semi-partitioned scheduling 13
6.6.3. Fault tolerance on electric vehicles 13
6.6.4. Scicos/SynDEx gateway and code generation for multi-core 13
6.6.5. SynDEx updates 14
7. Contracts and Grants with Industry 14
7.1. ID/TL-M project with ST Microelectronics 14
7.2. Thales ARCADIA/Melody 14
8. Partnerships and Cooperations 14
8.1. Regional Initiatives 14
8.2. National Initiatives 15
8.2.1. ANR RT-Simex 15
8.2.2. ANR HeLP 15
8.2.3. FUI Lambda 15
8.2.4. FUI PARSEC 16
8.2.5. FUI P 16
8.2.6. AS GeMoC 16
8.2.7. CNRS GDR ASR ACTRISS group 16

8.3. European Initiatives 16
8.3.1. FP7 Projects 16
8.3.1.1. CESAR 16
8.3.1.2. PRESTO 17
8.3.2. Collaborations in European Programs, except FP7 18
8.3.2.1. ITEA2 Timmo2U 18
8.3.2.2. ITEA2 OpenProd 18
8.4. International Initiatives 19
8.4.1. INRIA International Partners 19
8.4.2. Participation In International Programs 19
9. Dissemination **19**
9.1. Animation of the scientific community 19
9.2. Teaching 20
10. Bibliography **21**

Project-Team AOSTE

Keywords: Embedded Systems, Synchronous Languages, Model-Driven Engineering, Real-Time, Scheduling, Concurrency, Model Of Computation

Aoste is a joint team with the University of Nice/Sophia-Antipolis (UNS) and the UMR CNRS I3S. It is also co-located between the two INRIA centers of Sophia-Antipolis and Rocquencourt.

1. Members

Research Scientists

Robert de Simone [Team Leader, Senior Researcher, INRIA, HdR]
Yves Sorel [Team Vice-leader, Senior Researcher, INRIA]
Dumitru Potop Butucaru [Junior Researcher,INRIA]
Laurent George [Associate Professor, UPEC, on leave at INRIA, HdR]

Faculty Members

Charles André [Professor, UNS, emeritus, HdR]
Sid-Ahmed-Ali Touati [Professor, UNS, since September 2011, HdR]
Frédéric Mallet [Associate Professor, UNS, HdR]
Marie-Agnès Peraldi Frati [Associate Professor, UNS]
Julien Deantoni [Associate Professor, UNS]

Technical Staff

Nicolas Chleq [INRIA support engineer]
Benoît Ferrero [Temporary Engineer, until September 2011]
Simon Nivault [Temporary Engineer]
Meriem Zidouni [Temporary Engineer]
Daniel de Rauglaudre [INRIA support engineer]
Maxence Guesdon [INRIA support Engineer, until March 2011]
Cécile Stentzel [INRIA support Engineer, from April 2011]

PhD Students

Manel Djemal [CORDI/S scholarship]
Thomas Carle [INRIA scholarship, since September 2011]
Falou Ndoye [INRIA scholarship]
Carlos Gomez Cardenas [MENESR scholarship]
Jean-François Le Tallec [BDE scholarship with ScaleoChip]
Mohamed Marouf [INRIA scholarship, joint with EPI IMARA]
Ling Yin [ECNU Shanghai, from September 2011]

Post-Doctoral Fellows

Anthony Coadou [until March 2011]
Régis Gascon [until October 2011]
Calin Glitia [until June 2011]
Jean-Vivien Millo [since September 2011]
Arda Goknil [since September 2011]
Kelly Garces Pernett
Virginia Papailiopoulou

Administrative Assistants

Patricia Lachaume [TCN INRIA, part-time (50%)]
Christine Anocq [TRS INRIA, part-time (30%)]
Sandra Devauchelle [AI CNRS, part-time (5%)]

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 accelators and manycore processors. The optimized compilation of such kinds of applications ontosuch execution platforms involves complex mappping issues, bioth interms 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 decison are made towards placement and timing of various tasks and operations. This solves in part the constraints between clocks, commiting to schedule and placement decisions. The final version should be totally ordered, and then subjet 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 handling logical time, on Application-Architecture Adequation approach and real-time scheduling results has been progressed over the years, resulting in sofware environments such as SYNDEX or TimeSquare.

2.2. Highlights

Robert de Simone was made Honorary Professor of the Software Engineering Institute (SEI) at East China Normal University (ECNU) in Shanghai.

3. Scientific Foundations

3.1. Models of Computation and Communication (MoCCs)

Participants: Charles André, Anthony Coadou, Robert de Simone, Jean-Vivien Millo, Dumitru Potop Butucaru.

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 this 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/SYNCCHARTS and SIGNAL/POLYCHRONY [51], [52], [46], [15], [4], [13] share one main characteristics: they are specificed 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 refered 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 [47], [45] (consider SDF balance equations for instance [54]).

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 [48], or N-synchronous processes [49], 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 ultimatelly 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 5.2.

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: Charles André, Julien deAntoni, Frédéric Mallet, Marie-Agnès 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 phenonema 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. In practice we focused on the PAPYRUS open-source editor, mainly from CEA LIST. We developed under Eclipse the TIMESQUARE solver and emulator for CCSL constraints (see 5.1), with its own graphical interface, as a stand-alone software module, while strongly coupled with MARTE and Papyrus.

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: 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 [50]. 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 defines 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, that result is a set of algorithm graphs, "architecture-aware", corresponding to refinements of the initial algorithm graph, by computing spatial (distribution) and timing (scheduling) allocations of the operations according to the architecture graph resource availability. In that context "Adequation" refers to some search amongst the solution space of resulting algorithm graphs, labelled by timing characteristics, for one 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 adress two main issues: monoprocessor real-time scheduling 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 our monoprocessor real-time scheduling work, beside the classical deadline constraint, often equal to a period, we take into consideration dependences beetween tasks and several, possibly related, latencies. A latency is a generalization of the typical “end-to-end” constraint. Dealing with multiple real-time constraints raises the complexity of that issue. Moreover, because the preemption leads to a waste of resources due to its approximation in the WCET (Worst Execution Time) of every task as proposed by Liu and Leyland [55], we first studied non-preemtive real-time scheduling with dependences, periodicities, and latencies constraints. Although a bad approximation 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, but seeking the exact cost of the preemption such that it can be integrated in schedulability conditions, 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 exact cost of preemption is the most difficult part because it varies according to the instance of each task [10]. Finally, we investigate also the problem of mixing hard real-time and soft real-time constraints that arises in the most complex applications.

The second research area is devoted to distributed real-time scheduling with embedding constraints. We use the results obtained in the monoprocessor case in order to derive solutions for the problem of multiprocessor (distributed) real-time scheduling. In addition to satisfy the multiple real-time constraints mentioned in the monoprocessor case, we have to minimize the total execution time (makespan) since we deal with automatic control applications involving feedback. Furthermore, the domain of embedded systems leads to solve minimization resources problems. Since these optimization problems are of NP-hard complexity 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 such as variants of “simulated annealing”.

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 4.2).

Since real-time distributed systems are often safety-critical we address dependability issues, to tolerate faults in processors and communication interconnects. We maily focus on software redondancy, rather than hardware, to ensure real-time behaviour preservation in presence of faulty processors and/or communication media (where possible failures are predictively specified by the designer). We investigate fail silent, transient, intermittent, and Byzantine faults.

4. Application Domains

4.1. Multicore System-on-Chip design

Synchronous formalisms and GALS or multiclock extensions are natural model representations of hardware circuits at various abstraction levels. They may compete with HDLs (Hardware Description Languages) at RTL and even TLM levels. The main originality of languages built upon these models is to be based on formal *synthesis* semantics, rather than mere simulation forms.

The flexibility in formal Models of Computation and Communication allows specification of modular Latency-Insensitive Designs, where the interconnect structure is built up and optimized around existing IP components, respecting some mandatory computation and communication latencies prescribed by the system architect. This allows a real platform view development, with component reuse and timing-closure analysis. The design and optimization of interconnect fabric around IP blocks transform at modeling level an (untimed) asynchronous versions into a (scheduled) multiclock timed one.

Also, Network on Chip design may call for computable switching patterns, just like computable scheduling patterns were used in (predictable) Latency-Insensitive Design. Here again formal models, such as Cyclo-static dataflow graphs and extended Kahn networks with explicit routing schemes, are modeling elements of choice for a real synthesis/optimization approach to the design of systems.

Multicore embedded architecture platform may be represented as Marte UML component diagrams. The semantics of concurrent applications may also be represented as Marte behavior diagrams embodying precise MoCCs. Optimized compilations/syntheses rely on specific algorithms, and are represented as model transformations and allocation (of application onto architecture).

Our current work aims thus primarily at providing Theoretical Computer Science foundations to this domain of multicore embedded SoCs, with possibly efficient application in modeling, analysis and compilation wherever possible due to some natural assumptions. We also deal with a comparative view of Esterel and SystemC TLM for more practical modeling, and the relation between the Spirit IP-Xact interface standard in SoC domain with its Marte counterpart.

4.2. Automotive and avionic embedded systems

Model-Driven Engineering is in general well accepted in the transportation domains, where design of digital software and electronic parts in usually tighly coupled with larger apsects of system design, where models from physics are being used already. The formalisms AADL (for avionics) and AutoSar [53] (for automotive) are providing support for this, unfortunately not always with a clean and formal semantics. Thus there is a strong need here for approaches that bring closer together formal methods and tools on the one hand, engineering best practices on the other hand.

From a structural point of view AUTOSAR succeeded in establishing a framework that provides significant confidence in the proper integration of software components from a variety of distinct suppliers. But beyond those structural (interface) aspects, dynamic and temporal views are becoming more of a concern, so that AUTOSAR has introduced the AUTOSAR Specification of Timing Extension. AUTOSAR (discrete) timing models consist of timing descriptions, expressed by events and event chains, and timing constraints that are imposed on these events and event chains.

An important issue in all such formalisms is to mix in a single design framework heterogeneous time models and tasks: based on different timebases, with different triggering policy (event-triggered and time-triggered), and periodic and/or aperiodic tasks, with distinct periodicity if ever. Adequate modeling is a prerequisite to the process of scheduling and allocating such tasks onto complex embedded architectural platforms (see AAA appoach in foundations 3.3. Only then can one devise powerful synthesis/analysis/verification techniques to guide designers towards optimized solutions.

Traceability is also an important concern, to close the gap between early requirements and constraints modelling on the one hand, verification and correct implementation of these constraints at the different levels of the development on the other hand.

5. Software

5.1. TimeSquare

Participants: Charles André, Nicolas Chleq, Julien Deantoni, Benoît Ferrero, Frédéric Mallet [correspondant].

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 (see section 3.2), and more accurately on the associated Clock Constraint Specification Language (CCSL) for the expression of timing constraints.

TimeSquare offers four main functionalities:

1. graphical and/or textual interactive specification of logical clocks and relative constraints between them;
2. definition and handling of user-defined clock constraint libraries;
3. automated simulation of concurrent behavior traces respecting such constraints, using a Boolean solver for consistent trace extraction;
4. call-back mechanisms for the traceability of results (animation of models, display and interaction with waveform representations, generation of sequence diagrams...).

In practice TimeSquare is a plug-in developed with Eclipse modeling tools. The software is registered by the *Agence pour la Protection des Programmes*, under number IDDN.FR.001.170007.000.S.P.2009.001.10600. It can be downloaded from the site http://timesquare.inria.fr/. It has been integrated in the OpenEmbeDD ANR RNTL platform, and other such actions are under way.

5.2. K-Passa

Participants: Anthony Coadou, Jean-Vivien Millo [correspondant], Robert de Simone.

This software is dedicated to the simulation, analysis, and static scheduling 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.

The tool was originally developed mainly as support for experimentations following our research results on the topic of Latency-Insensitive Design. This research was conducted and funded in part in the context of the CIM PACA initiative, with initial support from ST Microelectronics and Texas Instruments.

KPASSA is registered by the Agence pour la Protection des Programmes, under the number IDDN.FR.001.310003.000.S.P.2009.000.20700. it can be downloaded from the site http://www-sop.inria.fr/aoste/index.php?page=software/kpassa.

5.3. SynDEx

Participants: Maxence Guesdon, Yves Sorel [correspondant], Cécile Stentzel, Meriem Zidouni.

SynDEx is a system level CAD software implementing the AAA methodology for rapid prototyping and for optimizing distributed real-time embedded applications. Developed in OCaML it can be downloaded free of charge, under INRIA copyright, from the general SynDEx site http://www.syndex.org.

The AAA methodology is described in section 3.3. Accordingly, SYNDEX explores the space of possible allocations (spatial distribution and temporal scheduling), from application elements to architecture resources and services, in order to match real-time requirements; it does so by using schedulability analyses and heuristic techniques. Ultimately it generates automatically distributed real-time code running on real embedded platforms. The last major release of SYNDEX (V7) allows the specification of multi-periodic applications.

Application algorithms can be edited graphically as directed acyclic task graphs (DAG) where each edge represent a data dependence between tasks, or they may be obtained by translations from several formalisms such as Scicos (http://www.scicos.org), Signal/Polychrony (http://www.irisa.fr/espresso/Polychrony), or UML2/MARTE models (http://www.omg.org/technology/documents/profile_catalog.htm).

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 shedulability analyses and optimization heuristics based on the minimization of temporal or resource criteria. For example while satisfying deadlines and latencies 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, executives 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.

5.4. SAS

Participants: Daniel de Rauglaudre [correspondant], Yves Sorel.

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 additionaly 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.

For a schedulable task system under EDF, when the exact preemption cost is considered, the period of the permanent phase may be much longer than the least commun multiple (LCM) of the periods of all tasks, as often found in traditional scheduling theory. Specific effort has been made to improve display in this case. The classical utilization factor, the permanent exact utilization factor, the preemption cost in the permanent phase, and the worst response time for each task are all displayed when the system is schedulable. Response times of each task relative time can also be displayed (separately).

SAS is written in OCaML, using CAMLP5 (syntactic preprocessor) and OLIBRT (a graphic toolkit under X). Both are written by Daniel de Rauglaudre.

6. New Results

6.1. Logical time in Model-Based embedded design

Participants: Charles André, Frédéric Mallet, Julien Deantoni, Robert de Simone, Marie-Agnès Peraldi Frati, Régis Gascon, Calin Glitia, Kelly Garces Pernett, Benoît Ferrero, Nicolas Chleq, Arda Goknil.

The foundational basis of our approach to modeling and analysis of embedded systems using logical time and logical clock specification contraints (CCSL) is recalled in 3.2, and was surveyed in [2]. This year we conducted a number of works exploiting this approach and promoting its introduction to various application domains.

Charles André presented the general approach in an invited lecture at the French Summer School on Real-Time, in Brest [21].

The HDR manuscript of Frédéric Mallet, where the MARTE Time Model is deeply considered, also in relation with other standards such as AADL, was published in book format [39].

In the article [19] we showed how CCSL observers could be encoded in the synchronous language Esterel, using crucial features of simultaneity, and how otherwise simultaneity could be obtained in simulation. This work was also presented internally as deliverable of the FUI Lambda project (see 8.2.3).

We drew a definite link with our activities on Process Network analysis (see 6.3), by showing how the CCSL primitives could be used to provide the loose timing semantic constraints of exiting PN models such as SDF (Synchronous Data-Flow domain of UC Berkeley's Ptolemy), and its Multi-Dimensionla extension (MD-SDF). This resulted in a journal publication [38]. Existing static schedules can then be obtained by analysis with K-Passa 5.2, or simulated using TimeSquare 5.1 (with an ASAP strategy).

In a collaboration with researchers at East China Normal University (ECNU Shanghai), we showed how CCSL constraints could be translated towards the PROMELA language implemented in the SPIN model-checker, which once again raises the issues of faithfully modeling simultaneity. This work resulted in a communication at the ICECCS conference [33]. Following this work one of our co-author, Yin Ling, earned a one-year scholarship from the Chinese government to visit us as part of her PhD.

The usage of CCSL expressions in the role of predicate property formulas, and their comparison with the more classical temporal formalisms such as PSL (Property Specification Language), was investigated in [24]. A longer internal report version can be found at [42].

In [23] we tackle the issue of recovering global information from multiple execution traces living in distinct logical time bases, with polychronous constraints relating them. The use for efficient debug of embedded systems from distributed traces is examplified on a case study of terrestrial robot. This work was conducted in the framework of the ANR RT-Simex project, see 8.2.1.

A case study in modeling with logical time and CCSL, from requirements to implementation, based on an automotive spark ignition system, is provided in [31]. We worked more generally on the introduction of our approach to existing formalism in the automotive domain, such as EAST-ADL2 and AutoSar, as part of our contribution to the new ITEA2 Timmo2U project. Premises of this effort are described in [32].

The use of CCSL constraints in general requirement engineering was also studied and demonstrated in a conference article, jointly with colleagues at ECNU Shanghai, presented at APSEC'2011 [22].

The use and modeling of priorities amongst timed events (i.e., logical clock ticks), which has strong impacts on efficient logical clock based simulations and scheduling (as the choice of next event), is still a topic of ongoing work. Several advanced considerations are to be found as part of jean-François Le Tallec PhD thesis, to be defended in January 2012 [16].

6.2. Model-Based approaches to SoC design

Participants: Charles André, Robert de Simone, Benoît Ferrero, Carlos Gomez Cardenas, Jean-François Le Tallec.

The main practical goal of this work was to combine in a sensible way the various formalisms SystemC, IP-XACT, UML MARTE, and UPF (for power consumption representation) (see 4.1 for further descriptions). There were true motivations for this: SystemC is a de-facto standard for SoC simulation at various levels, but currently lacks any sort of formal description so that systems can be analysed, reasoned about for correctness and optimized (and it becomes even more so with newer draft standard evolutions). IP-XACT was introduced as an ADL to ease composition and assembly of IP components (written in SystemC or not), but again it currently fails short of its goal, and in particular does not allow standard decoration of model attributes in prominent non-functional domains such as timing/performance and low-power/energy consumption. These could be provided with the help of dedicated features in UML MARTE, and aligned on the UPF standard for power management modeling.

While the intended design flow would take the UML MARTE and UPF to IP-XACT to SystemC direction, it was important to extract IP-XACT and MARTE structural representation from existing SystemC programs, both to populate the flow with existing legacy models, and to explore better the requirements for complete and consistent modeling towards IP block assembly. This work was conducted in Jean-François Le Tallec PhD, to be defended in January 2012 [16]. Together with Benoît Ferrero he defined and realized a software tool named SCiPX (SystemC to IP-XACT translator), originally based on the PinaVM tool by VERIMAG and the DoxyGen syntactic analyzer.

SCiPX is available in prototype version from our site http://www-sop.inria.fr/aoste/index.php?page=software/scipx. It can be combined with the former software transformation modules IPXACT2Marte and Marte2IPXACT developed previously. These results were partly supported by the ID/TL-M contract wih ST Microelectronics (see 7.1), and the ANR HeLP project (see 8.2.2), and were presented in [26], [25]

As part of his PhD thesis, Carlos Gomez Cardenas described a subset of UPF standard as a metamodel inside UML MARTE. He also considered compatibility and interconnections with the industrial environments AcePlorer (by Docea Power), and Synopsys Virtualizer (formerly CoWare), provided to us in the context of the CIM PACA tool farm 8.1.1. Preliminary results were presented in [36]. This work was also presented during internal meetings of the ANR HeLP project, and coordinated with work conducted in the team of Michel Auguin at CNRS UMR LEAT (also in Sophia-Antipolis).

6.3. Process Network analysis

Participants: Anthony Coadou, Robert de Simone, Jean-Vivien Millo, Sid-Ahmed-Ali Touati.

This year we comforted the type of analysis on regular static scheduling and routing in dedicated process network models such as studied in the successive PhD thesis of Julien Boucaron, jean-Vivien Millo, and Anthony Coadou, and recently surveyed in [5]. This resulted mainly in further implementation upgrades of our K-Passa tool (see 5.2), performed first by Anthony Coadou (before he left on a postdoc internship), then continued by Jean-Vivien Millo (on a return postdoc position with us).

In a work mostly conducted while member of the Alchemy EPI in Saclay, but which draws a clear link to our past and present activities in the subject, Sid-Ahmed-Ali Touati studied efficient heuristics to the general problem of one-dimensional periodic task scheduling under storage requirements, using a modeling framework akin to Process Networks. This resulted in a journal article accepted once the author had become attached to the Aoste EPI [17].

6.4. Correct and efficient implementation of polychronous formalisms

Participants: Thomas Carle, Manel Djemal, Virginia Papailiopoulou, Dumitru Potop Butucaru, Robert de Simone, Yves Sorel.

Existing analysis techniques for synchronous and polychronous languages, such as *clock calculi*, are meant to extract relations of *simultaneity* (time inclusion) and *exclusiveness* (time exclusion) between the various computations and communications. This approach is well-suited when targeting sequential processors. For distributed or multi-threaded implementations, further *independence* relations are needed to express potential concurrency. This resulted in a general theory of endochronous systems, meant to support this additional analysis [11].

Last year we completed a first prototype tool implementation for weak endochrony checking. This was completed this year in two directions:

- connecting our tool with Signal as input language, and interface it in practice to the Polychrony/SME environment developed by the Espresso EPI;
- Improving algorithmic complexity and internal data representaion, so that our tool can now handle reasonable size Signal programs.

This work was of course conducted in collaboration with Espresso members. Experimental results were presented at the ESLsyn 2011 conference [30]. We are currently expanding the framework in orer to take modes/states into account in the program specifications. Effective generation of multi-threaded GALS wrappers for Signal programs is also under way.

We worked at extending the AAA methodology for polychronous processes by providing a better integration of clock analysis in the various phases of the implementation process (allocation, scheduling, pipelining, etc.). We also considered a wider range of implementation targets (time-triggered, MPSoC). We defined a dedicated *software pipelining* algorithm to match conditional scheduling/reservation tables such as used in SynDEx, with the goal of improving throughput with the same duration of individual computation cycles (as is the goal of any pipelning techniques). The originality here is to make logical clocks of polychronous systems act as triggers for the *predicated executions* as used in traditional software pipelining. First results have been presented during the Synchron 2011 workshop and in a research report [41].

Further work on time-triggered systems was submitted inside the FUI Parsec 8.2.4 and P 8.2.5 projects, including real-time implementation methods for the IMA/ARINC 653 avionics platforms. In particular we conducted experiments to replace the scheduling policy of the second-level scheduler (L1 in the standard) from dynamic priority-driven to dynamic Time Division time triggered (TT-IMA). Preliminary results are under way, and were informally presented at the yearly Synchron seminar.

An important emerging trend in target MPSoC platforms is that On-chip networks are progressively introduced to cope with the botleneck of iter processor communications. Correct implementation of polychronous systems in this context thus relies on efficient routing of data in such networks, and ultimately may assume that on-chip NoC routers may be programmed in one way or another to behave predictably according to the global

application distributed on the cores. We started a collaboration on this topic with the "Embedded Systems-on-Chips" department of the LIP6 laboratory, one of the main site of expertise for SoC/NoC design and Hardware/software codesign. This collaboration first materialized with the co-supervision of M. Djemal's PhD thesis. A generic MPSoC architecture is being defined, which includes a 2D mesh network-on-chip with programmable routers, on which static routing schedules such as synthesized by our tools may be implemented and run.

6.5. Uniprocessor Real-time Scheduling

Participants: Laurent George, Mohamed Marouf, Daniel de Rauglaudre, Yves Sorel.

6.5.1. Strict periodic harmonic tasks

This year, we focused our work on scheduling of strict periodic tasks to the particular case of harmonic tasks [28]. After transforming the scheduling problem into a bin-packing problem, we performed a schedulability analysis and proposed schedulability conditions in each sub-case of harmonic tasks: we proposed a necessary and sufficient condition in the case where all tasks periods are distinct, and we proposed a sufficient condition in the case where some tasks have the same period. Finally, we proposed a scheduling algorithm based on the bin-packing problem resolution.

6.5.2. Combination of strict periodic and sporadic tasks

Non-preemptive strict periodic tasks are harder to schedule than preemptive ones. One can hope to extend schedulability results when combining non-preemptive strict periodic tasks with preemptive sporadic one.

We proposed in [27] a schedulability analysis for a combination of strict periodic and sporadic tasks. We considered all tasks with fixed priorities, where the highest priorities are given to strict periodic tasks and the lower priorities are given to sporadic tasks. First, we scheduled strict periodic tasks using our former scheduling algorithm. Then, we computed the critical instants which maximize the response time of a sporadic task. We proved that the critical instants are contained in the permanent phase of strict periodic tasks, and are given by the start times of strict periodic jobs in a hyper-period. We also proved that we can reduce critical instants by eliminating some of them. Then, we gave the analytic expression of the computing time $W_i(t)$ at any time t necessary for the execution of a task τ_i taking in consideration all the tasks with higher priorities. That allows the computation of the response time r_i by solving the equation $W_i(t) = r_i$. Therefore, for a sporadic task, if its response time r_i is less of equal to its deadline for all critical instants, then this latter task is schedulable, else it is not schedulable. We proceed similarly for all sporadic tasks to prove that a tasks set is schedulable or not.

6.5.3. Exact cost of RTOS

It is important to determine the exact cost of the real-time operating system (RTOS) when preemptive scheduling is used for better processor utilization compared to non preemptive scheduling [43]. Indeed, in this case it is possible to trust the schedulability conditions when they include that cost, and also to avoid waste resources. This year we developed a generic RTOS modelled with Petri nets and we determined its exact cost on an ARM9 processor. We used Petri nets on the one hand to choose through simulations the best structure of that scheduler, and on the other hand to verify non blocking properties. In order to obtain its exact cost the scheduler was programmed in assembly language, and coded such as it is deterministic, i.e. its cost does not depend on alternative statements but only on the number of tasks which is known a priori. Using this RTOS we experimented simple task sets on the ARM9 processor for which we were able to include the exact RTOS cost in the schedulability conditions.

6.5.4. Formal proofs of real-time scheduling theorems

Scheduling involves numerous models and theorems, sometimes dated of several decades, but never formally proved. We made a formal proof in Coq (proof assistant developped at Inria) to check a classical theorem giving a schedulability condition for a set of real-time strictly periodic tasks (about 1500 lines of Coq). This work was published in a paper accepted for publication in the conference JFLA 2012.

A second proof is actually being carried now, dealling with response time of a set of fixed priority real-time preemptive tasks. The theorem states that the worst case of this response time occurs when all tasks start simultaneously. A step in the original argument by Jane W. S. Liu [55] involves the proof of a function whose fixpoint computes the response time of the first instance of the least priority task. This specific step is now formally proved in Coq (3500 lines of Coq), and we are now working on the completion of the full theorem.

6.6. Multiprocessor Real-time Scheduling

Participants: Laurent George, Maxence Guesdon, Mohamed Marouf, Falou Ndoye, Simon Nivault, Yves Sorel, Cécile Stentzel.

6.6.1. Partitioned scheduling with exact RTOS cost

In the case of partitioned scheduling we propose a greedy heuristic to solve the real-time scheduling problem of periodic preemptive tasks on a multiprocessor architecture while taking into account the exact RTOS cost. This is achieved by combining an allocation heuristic, of "best fit" type, and a schedulability condition based on the operation $\oplus$ which takes into account the exact RTOS cost [43]. The allocation heuristic minimizes the makespan (total execution time of the tasks executed on the multiprocessor taking into account inter-processor communication costs). A first version of that work was presented in [29].

6.6.2. Semi-partitioned scheduling

In [18] we study two cases of semi-partitioned scheduling of sporadic tasks with constrained deadlines on homogeneous multi-processor: (i) the case where the Worst Case Execution Time (WCET) of a job can be portioned, each portion being executed on a dedicated processor, according to a static pattern of migration and using for solving the local assignment problem a linear programming approach ; (ii) the case where the jobs of a task are released on a processor, 1 time out of p, where p is an integer less than or equal to the number of processors, according to a Round-Robin migration pattern. The first approach has been investigated in the state-of-the-art by migrating a job at its local deadline, computed from the deadline of the task it belongs to.

6.6.3. Fault tolerance on electric vehicles

We consider applications composed of a real-time task set running on the distributed heterogeneous architecture of the CyCab (electric vehicle developped in the IMARA team-project) based on dsPICs processors, MPC555 micro-controllers, and an embedded PC all together connected through CAN (Controller Area Network) buses. For hardware reasons we suppose that only dsPICs and CAN buses can fail. Our goal is to find a fault-tolerant software solution to tolerate such failures while the applications satisfy the real-time constraints. Because extra hardware for error detection is expensive in such electric vehicle, we proposed a software error detection based on watchdogs. We solved separately two different problems: buses and dsPICs fault-tolerance. In both cases we use active redundancy policies. For buses fault-tolerance, we assume that all processors are reliable, and all but one bus can fail. The same data is sent through all the CAN buses. If a CAN bus fails then the data is sent by the other CAN buses. For processors fault-tolerance, we assume that all communication media are reliable and at least one processor can fail. The first step consists in performing active redundancy for all the tasks of the application. A task and their redundant tasks are assigned to different processors. If processor fails then the data which are not sent by tasks running on that faulty processor, are actually sent by the redundant tasks. All the tasks with their redundant counterparts are scheduled according to the schedulability analysis proposed in [28].

6.6.4. Scicos/SynDEx gateway and code generation for multi-core

This work was carried out in the Openprod project (see 8.3.2.2). The gateway between Scicos and SynDEx has been updated to deal with the last Scicos data structures and the last version of SynDEx. Besides, this gateway has been improved and partially rewritten to support as much Scicos blocks as possible. We use the gateway to automatically produce from a control model specified and simulated in Scicos a real-time executable running on a multi-core platform. The latter platform is described according to the shared memory model defined last year. In order to generate real-time executable code we had to develop a new SynDEx executive kernel based on Windows-RTX which supports shared memory communications and multi-core parallel execution. That executive kernel is used with the macro-code generated by SynDEx to produce the real-time executable code.

6.6.5. SynDEx updates

We continued the developments of future version 8 of SynDEx which will feature a new software architecture to allow better functionality evolutions and maintenance. On the other hand in the COTROS ADT ("*Génération de code temps réel distribué optimisé et sûr*"), we achieved the new automatic code generator for the current version 7 of SynDEx. This generator creates intra and inter-processor synchronizations according to well defined rules, checks deadlock absence in inter-processor synchronizations, manages efficiently buffers and semaphores (declaration, naming, etc.).

7. Contracts and Grants with Industry

7.1. ID/TL-M project with ST Microelectronics

Participants: Charles André, Robert de Simone, Benoît Ferrero, Jean-François Le Tallec.

ID/TL-M is a project launched as part of the larger NANO 2012 programme conducted by ST Microlectronics in Rhône-Alpes. Its main goal is to study the potential use of model-driven engineering techniques (MDE) for Electronic System-Level Design (ESL) of Systems-on-Chip (SoC).

In particular we focused this year on the relations and connexions between UML MARTE profile and the other standard IP-XACT, itself introduced as dedicated Architecture Description Language (ADL) for easy assembly of IP hardware components. One advantage here of MARTE in our view is that it is meant to be extendable to comprise Non-Functional Property annotations, such as consumption for low-power, in a much more open and larger setting as the extensions under way at Accelera (the IP-XACT standardisation body which recently merged with OSCI, the Open SystemC Initiative).

The direct collaboration in ID/TL-M allows implementations of tools and methods whose general descriptions aresomehow shared with the neighboring ANR project HeLP (see below). Nevertheless, due to external reasons of political nature, funding of the general nano2012 programme was halted in 2011, and this project was consequently put on stand-by.

7.2. Thales ARCADIA/Melody

Participants: Frédéric Mallet, Robert de Simone.

During the course of the ARTEMIS CESAR project, we exchanged views with partners at Thales on potential methodologies based on Model-Driven Engineering for Embedded Systems. These considerations were mostly aimed at the support with tools of the various allocation and refinement steps in a V-cycle process, considering joint software and hardware design. Subsequently we were invited to conduct an evaluation survey and expert consulting on their internal MDE development project, the ARCADIA methodology (supported by the Melody tool environment).

The work included identification of potential ambiguous points in the representation models, followed by the definition of a relevant set of questions regarding possible interpretations. This form was then submitted to a panel of development engineers inside the company. Their return answers were analyzed by us, jointly with the promotors of the methodology inside Thales. Recommandations for improvements followed.

This job was conducted under Non-Disclosure Agreement (as the methodology remains proprietary, and is not part of CESAR tool deliverables). It led to a Grant agreement from this company to our team.

8. Partnerships and Cooperations

8.1. Regional Initiatives

8.1.1. CIM PACA

Participants: Robert de Simone, Jean-François Le Tallec, Carlos Gomez Cardenas.

This ambitious regional initiative is intended to foster collaborations between local PACA industry and academia partners on the topics of microelectronic design, though mutualization of equipments, resources and R&D concerns. We are so far actively participating in the Design Platform (one of the three platforms launched in this context), of which INRIA is a founding member.

This year the platform acquired more EDA tools, such as Synopsys Virtualizer (comprising the former CoWare virtual platform environment), and Docea Power AcePlorer (which we are using in th ecourse of the ANR HeLP project). Several Aoste members attended specific trainings on these tools.

Jean-François Le Tallec is currently completing his PhD thesis (expected January 2012), which was partly funded on the CIM PACA initiative. Apart from this, which will close the lifespan of the Sys2RTL CIM PACA project, we are looking for further collaborative associations including the team of Michel Auguin at CNRS UMR LEAT, Texas Instruments, and maybe Synopsys amongst other partners. Discussions for project submissions are under way (one difficulty here is that US companies are not familiar with european or national collaborative fundings).

8.2. National Initiatives

8.2.1. ANR RT-Simex

Participants: Julien deAntoni, Kelly Garces Pernett, Frédéric Mallet.

The RT-Simex project is dedicated to the reverse engineering of analysis traces of simulation and execution back up to the source code, or in our case most likely into the original models in a MARTE profile representation. The prime contractor is OBEO, a software publishing company based in Nantes.

8.2.2. ANR HeLP

Participants: Jean-François Le Tallec, Carlos Gomez Cardenas, Dumitru Potop Butucaru, Robert de Simone.

The ANR HeLP project deals with joint modeling of functional behavior and energy consumption for the design of low-power heterogeneous SoCs. Partners are ST Microelectronics and Docea Power (SME) as industrial; INRIA, UNS (UMR LEAT), and VERIMAG (coordinator) as academics. Our goal in this project is twofold: first, combine SoC modeling with temporal behavior and logical time (as obtained in the ID/TL-M collaboration, see 7.1) with energy/power modeling as extra annotations on MARTE models; second, compare the capacities of hig-level SystemC TLM abstraction with that of Esterel seen as a multiclock formalism based on logical abstract time.

The PhD thesis of Jean-François Le Tallec, originally funded in the CIM PACA programme, is being continued as part of the HeLP project. Additionaly, part of Carlos Gomez Cardenas PhD work on metamodeling in MARTE of power consumption and links to dedicated tools is also presented to this project (in connection with complementary work at LEAT on this topic).

8.2.3. FUI Lambda

Participants: Charles André, Frédéric Mallet.

In the context of embedded software deployed on "off the shelf" execution platforms, the LAMBDA project has two major goals:

- To demonstrate the technical feasibility and the interest of model libraries by formalizing the key properties of execution platforms,
- To reconcile appropriated standards (SysML, MARTE, AADL, IP-XACT) with de facto standards (already implemented by widespread analysis and simulation tools.)

In this context we provided expertise mainly on the SyncCharts, MARTE, and SysML formalisms (our involvement in this project is only marginal, in support of other INRIA teams). The final project review was held at the end of September, 2011.

8.2.4. FUI PARSEC

Participants: Dumitru Potop Butucaru, Thomas Carle, Virginia Papailiopoulou, Yves Sorel.

The Parsec project is a large collaboration with partners such as Thales, CEA, Elidiss, INRIA, Systerel, OpenWide, Alstom, and TelecomParisTech. The project aims at defining a framework for the development of distributed real-time embedded systems that are subject to strict certification standards such as DO-178B (for avionics), IEC 61508 (for transportation systems), or ISO/IEC 15408 (the Common Criteria for information technology security evaluation).

The AOSTE team uses its expertise in the modeling and distributed real-time implementation of embedded applications using synchronous formalisms and associated tools. The two main scientific challenges of the project are (1) a better modeling of the distributed implementation architectures, allowing code generation for novel architectures and better code generation for architectures we currently handle, and (2) the modeling and efficient implementation of mode changes, as they are specified in an industrial context.

Virginia Papailiopoulou was partially funded as post-doc on this project, which will also finance the PhD scholarship of Thomas Carle.

8.2.5. FUI P

Participants: Dumitru Potop Butucaru, Yves Sorel.

The main purpose of this project is to define a *P*ivot format that allows the automatic generation of certified code for safety critical applications. Partners of this project are: Aboard, ACG, Airbus, Adacore, Altair, Astrium, Atos, Continental, ENPC, INRIA, IRIT, LABSTICC, ONERA, RCF, SAGEM, Scilab, STI, Thales-AS, Thales-AV.

The project was only recently started, and first concrete results are expected for next year.

8.2.6. AS GeMoC

Participants: Julien deAntoni, Kelly Garces Pernett, Frédéric Mallet.

The purpose of the Action Spécifique by CNRS is to gather the French research community working around heterogeneous modeling of complex systems. Funding was granted for a couple of internal visits and plenary meetings this year. TimeSquare was presented in this context, and a survey of methods (including ours) is being conducted.

8.2.7. CNRS GDR ASR ACTRISS group

Participant: Laurent George.

The ACTRISS working group, supported by GDR ASR (CNRS, France), is meant to federate and promote research on real-time systems in France. A workshop on multiprocessor systems was organized in this framework in May 2011 (see http://www-roc.inria.fr/who/Laurent.George/ACTRISS/).

8.3. European Initiatives

8.3.1. FP7 Projects

8.3.1.1. CESAR

Participants: Régis Gascon, Yves Sorel, Robert de Simone.

Title: CESAR

Duration: February 2009 - July 2012

Coordinator: AVL - GmbH (Austria)

Others partners: AIRBUS Operations GbmH (Germany), AIRBUS Operations SAS (France), ABB AS (Norway), ABB AB (Sweden), AbsInt Angewandte Informatik GmbH (Germany), ACCIONA Infraestructuras S.A. (Spain), Ansaldo STS S.p.A. (Italy), ASTRIUM SAS (France), AIRBUS Operations Limited (United Kingdom), Aristotle University of Thessaloniki (Greece), Commissariat à l´Energie Atomique (France), CNRS (France), Centro Ricerche Fiat S.C.p.A. (Italy), Critical Software S.A. (Poland), Danieli Automation S.p.A. (Italy), Delphi France SAS (France), Deutsches Zentrum für Luft- und Raumfahrt e.V. (Germany), Dassault Systemes (France), EADS Deutschland GmbH (Germany), Fondación Tecnalia Research & Innovation (Italy), ESTEREL Technologies SA (France), Fraunhofer Gesellschaft zur Förderung der Angewandten Forschung e.V. (Germany), Auvation Software Limited (United Kingdom), Hellenic Aerospace Industry S.A. (Greece), Infineon Technologies Austria AG (Austria), Infineon Technologies AG (Germany), Institut National de Recherche en Informatique et Automatique (France), ATHENA - Industrial Systems Institute (Greece), Kungliga Tekniska Høgskolan (Sweden), Norweigan University of Science and Technology (Norway), National Technical University of Athens (Greece), OFFIS e.V. (Germany), Office national d'Etudes et de Recherches Aérospatiales (France), BTC - Embedded Systems AG (Germany), Oxford University (United Kingdom), Sagem Défense Sécurité (France), AleniaSIA Spa (Italy), Siemens AG (Germany), Stiftelsen SINTEF (Norway), Quintec Associates (Thales Consulting and Engineering) (United Kingdom), Thales Communications S.A. (France), Thales Avionics S.A. (France), Thales TRT (France), Alma Mater Studiorum - Universià di Bologna (Italy), The University of Manchester (United Kingdom), Università degli Studi di Trieste (Italy), The Virtual Vehicle Competence Center (Austria), Volvo Technology Corporation (Sweden), Messier-Bugatti S.A. (France), TURBOMECA (France), SNECMA S.A. (France), Geensoft (France), Selex Sistemi Integrati (Italy).

See also: http://www.cesarproject.eu/

Abstract: CESAR stands for Cost-efficient methods and processes for safety relevant embedded systems and is a European funded project from ARTEMIS JOINT UNDERTAKING (JU). The three transportation domains automotive, aerospace, and rail, as well as the automation domain share the need to develop ultra-reliable embedded systems to meet societal demands for increased mobility and ensuring safety in a highly competitive global market. To maintain the European leading edge position in the transportation as well as automation market, CESAR aims to boost cost efficiency of embedded systems development and safety and certification processes by an order of magnitude. CESAR pursuits a multi-domain approach integrating large enterprises, suppliers, SME's and vendors of cross sectoral domains and cooperating with leading research organizations and innovative SME's.

8.3.1.2. PRESTO

Participants: Frédéric Mallet, Marie-Agnès Peraldi Frati, Julien DeAntoni.

Title: PRESTO

Duration: April 2011 - March 2014

Coordinator: Miltech (Greece)

Others partners: TELETEL S.A. (Greece), THALES Communications (France), Rapita Systems Ltd. (United Kingdom), VTT (Finland), Softeam (France), THALES (Italy), MetaCase (Finland), INRIA (France), University of L'Aquila (Italy), MILTECH HELLAS S.A (Greece), PragmaDev (France), Prismtech (United Kingdom), Sarokal Solutions (Finland).

See also: http://www.presto-embedded.eu/

Abstract: The PRESTO project aims at improving test-based embedded systems development and validation, while considering the constraints of industrial development processes. This project is based on the integration of test traces exploitation, along with platform models and design space exploration techniques. Such traces are obtained by execution of test patterns, during the software

integration design phase, meant to validate system requirements). The expected result of the project is to establish functional and performance analysis and platform optimisation at early stage of the design development. The approach of PRESTO is to model the software/hardware allocation, by the use of modelling frameworks, such as the UML profile for model-driven development of Real Time and Embedded Systems (MARTE). The analysis tools, among them timing analysis including Worst Case Execution Time (WCET) analysis, scheduling analysis and possibly more abstract system-level timing analysis techniques will receive as inputs on the one hand information from the performance modelling of the HW/SW-platform, and on the other hand behavioural information of the software design from tests results of the integration test execution.

The PRESTO project (ARTEMIS-2010-1-269362) is co-funded by the European Commission under the ARTEMIS Joint Undertaking Programme.

8.3.2. Collaborations in European Programs, except FP7

8.3.2.1. ITEA2 Timmo2U

Participants: Marie-Agnès Peraldi Frati, Julien DeAntoni, Arda Goknil, Jean-Vivien Millo, Yves Sorel.

Program: ITEA2

Project acronym: Timmo2Use

Project title: TIMing MOdel, TOols, algorithms, languages, methodology, and USE cases

Duration: October 2010 - October 2012

Coordinator: Volvo Technology AB (Sweden)

Other partners: AbsInt Angewandte Informatik GmbH (Germany), Arcticus Systems AB (Sweden), Chalmers University of Technology (Sweden), Continental Automotive GmbH (Germany), Delphi France SAS (France), dSPACE GmbH (Germany), INCHRON GmbH (Germany), Institut National de Recherche en Informatique et Automatique (France), Mälardalen University (Sweden), Rapita Systems Ltd. (United Kingdom), RealTime-at-Work (France), Robert Bosch GmbH (Germany), Symtavision GmbH (Germany), Technische Universität Braunschweig (Germany), Time Critical Networks (Sweden), Universität Paderborn (Germany).

See also: http://timmo-2-use.org/

Abstract: TIMMO develops different types of timing constraints and dynamic behaviour in the supply chain of the complex development process is of crucial importance when designing distributed real-time automotive systems. TIMMO-2-USE stands for TIMing MOdel - TOols, algorithms, languages, methodology, and USE cases which summarizes the main objectives of the project, i.e., the development of novel tools, algorithms, languages, and a methodology validated by use cases.

The project provides partial funding for the postdoctoral positions of Jean-Vivien Millo and Arda Goknil.

8.3.2.2. ITEA2 OpenProd

Participants: Simon Nivault, Yves Sorel.

Program: ITEA2

Project acronym: OpenProd

Project title: Open Model-Driven Whole-Product Development and Simulation Environment

Duration: June 2009 - May 2012

Coordinator: Siemens Industrial TurboMachinery AB (Sweden)

Other partners: Appedge (France), Bosch Rexroth AG (Sweden), CEA LIST (France), EADS Innovation Works (France), Electricité De France (France), Equa Simulation AB (Sweden), ETH Zürich (Switzerland), Fachhochschule Bielefeld (Germany), Fraunhofer FIRST (Germany), IFP (France), INRIA Rocquencourt (France), INSA Lyon (France), Linköping University (Sweden), LMS Imagine (France), MathCore Engineering AB (Sweden), Metso Automation (France), Nokia (Finland), Plexim GmbH (Germany), Pöyry Forest Industry (Finland), PSA Peugeot Citroen (France), Siemens AG, Sector Energy (Germany), SKF Sverige AB (Sweden), Technische Universität Braunschweig (Germany), TLK Thermo GmbH (Germany), VTT Technical Research Centre (Finland), XRG Simulation GmbH (Germany).

See also: http://www.ida.liu.se/~pelab/OpenProd/

Abstract: The OPENPROD project is developing an open whole-product, model-driven systems development, modelling and simulation (M&S) environment that integrates the leading open industrial software development platform Eclipse with open-source modelling and simulation tools such as OpenModelica and industrial M&S tools and applications. The project will enable a more formalised validation of production to cut time to market and ensure higher quality, using open solutions which will have a high impact, based on easy uptake and wide dissemination.

8.4. International Initiatives

8.4.1. INRIA International Partners

We are continuing a collaboration with East China Normal University (ECNU) in Shanghai, through the Software Engineering Institute headed by He Jifeng. This collaboration is shared with the OASIS EPI. As part of this we held a dedicated Workshop in Shanghai in November, met some of the LIAMA staff while there, and participated to a proposal for a new Associated Team mainly headed by OASIS. We are also hosting for a year Yin Ling, a PhD student from ECNU, on a chinese government grant.

8.4.2. Participation In International Programs

8.4.2.1. NoE Artist-Design

We are affiliated to this european Network of Excellence Artist-Design (http://www.artist-embedded.org/artist/), which sponsors events in our field.

9. Dissemination

9.1. Animation of the scientific community

Robert de Simone

Technical Program Committee: MemoCODE 2011, EmSOFT 2011, FDL 2011, ESLsyn 2011, SIES 2011, UML&AADL 2011.

Program co-organizer: UML&FM 2011

Board of Administrators: CIM PACA Design Platform association

PhD reviewer: Florence Plateau (University Paris XI), Giovanni Funchal (VERIMAG Grenoble), Stéphane Lecomte (Supelec Rennes)

PhD examiner: Mathieu Acher (UNS) (President of Jury).

Yves Sorel

Technical Program Committee: RTNS 2011, DASIP 2011, Traitement du Signal (Journal).

PhD reviewer: Ahmad Al Sheikh (INSA Toulouse), Frédéric Fauberteau (University Paris Est).

PhD examiner: Dominique Bertrand (University Toulouse), Cyril Faure (University Paris Est).

Steering Committee: OCDS/SYSTEM@TIC Paris-Region Competitivity Cluster.

Frédéric Mallet

Technical Program Committee: TASE 2011, M-BED 2011, ESLsyn 2011, SAFA 2011

PhD reviewer: Daniel Knorreck (Telecom-ParisTech)

PhD examiner: Kelly Garcès-Perrett (Ecole Centrale de Nantes)

DumitruPotop-Butucaru

Technical Program Committee: ACSD 2011, ESLsyn 2011, APRES 2011

Julien Deantoni

Action co-leader: CNRS Specific Action AS GeMoC

Laurent George

Program Chair: RTNS 2011 (with Alan Burns, University of York)

Technical Program Committee: RTNS 2011, OPODIS 2011, WFCS 2011, ICONS 2011

Co-leader of the ACTRISS group, supported by GDR ASR (CNRS, France),

Organisation: ACTRISS workshop (http://www-roc.inria.fr/who/Laurent.George/ACTRISS/).

PhD reviewer: Henri Bauer (ENSEEIHT, Toulouse)

PhD examiner: Frédéric Fauberteau (University Paris Est).

9.2. Teaching

Julien Deantoni

Licence: Computer Environnement, 30 h, L2 level, Polytechnical engineering school of Université de Nice/Sophia-Antipolis (UNS EPU) France.

Master: Model Driven Engineering, 22 h, M2, UNS EPU.

Master : C++ and Object Oriented Programming, 54 h, M1, UNS EPU.

Master: Embedded Software and systems, 7 h, M2, UNS EPU.

Master: VHDL, 20 h, M1, UNS EPU.

Sid-Ahmed-Ali Touati

Licence: Assembleurs et jeux d'instructions, 52h, L2, UNS Licence info.

Licence: Systèmes d'exploitation, 27h, L2, UNS Licence info.

Master: Systèmes d'exploitation, 36h, M1, UNS Master ISI.

Master: Programmation efficace et Optimisation de code, 23h, M1, UNS Master ISI.

Frédéric Mallet

Licence : Introduction à la Programmation Objet, 45h, L1, UNS.

Licence: Architecture des ordinateurs, 45h, L3, UNS.

Master: Programmation Avancée et Design Patterns, 93h, M1, UNS.

Master: Java pour l'Informatique Industrielle, 24h, M1, UNS.

Master: Architectures des ordinateurs, 12h, M1, UNS.

Master: Formal Models for Network-On-Chips, 12h, M2, UNS.

Marie-agnes Peraldi-Frati

Licence : Algorithms and programming 60h,L1, UNS Institute of technology.

Licence : System and Networks administration 80h, L2, UNS Institute of technology .

Licence : Web Programming 50 h , L2, UNS Institute of technology.

Robert de Simone

Master UNS Ubinet : Formal Models for Networks-on-Chip (together with Frédéric Mallet), 24h , M2, UNS.

Master UNS ISI : Semantics of Embedded and Distributed Systems, 24 h (together with Ludovic Henrio of Oasis EPI), M1, UNS.

Yves Sorel

Master: Optimization of distributed real-time embedded systems, 24H, M2, University Paris Sud.

Master: Distributed real-time systems, 26H, M2, University Paris Est

Master: Specification and formal models for embedded systems, 26H, M2, ENSTA Engineering School Paris

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

Dumitru Potop-Butucaru

Master: Programmation synchrone des systèmes temps-réel, 8h, M1, EPITA Engineering School Paris

PhD in progress : Jean-François le Tallec, entitled *Extraction de modèles pour la conception de systèmes sur puce*, started November 2007, defense scheduled on January 25, 2012; supervised by Charles André.

PhD in progress : Mohamed Marouf, entitled *Implementation of real-time multi-periodic fault tolerant robotics applications onto distributed architecture*, started January 2009; supervised by Yves Sorel.

PhD in progress : Manel Djemal, entitled *??*, started 2010; co-supervised by Dumitru Potop (with Alix Munier, professsor at University Paris 6).

PhD in progress : Carlos Gomez Cardenas, entitled *??*, started October 2010; supervised by Frédéric Mallet.

PhD in progress : Thomas Carle, entitled *??*, started October 2011; supervised by Dumitru Potop.

PhD in progress : Falou Ndoye, entitled *Distributed real-time scheduling with optimized preemption*, started January 2011; supervised by Yves Sorel.

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, RR-6925, 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, p. 201–227, Chapter 7, http://hal.inria.fr/inria-00495664.

[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, p. 41–78, Chapter 2, http://hal.inria.fr/inria-00495667.

[6] J. BOUCARON, R. DE SIMONE, J.-V. MILLO. *Latency-insensitive design and central repetitive scheduling*, in "MEMOCODE", 2006, p. 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, p. 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, p. 229–262, Chapter 8, http://hal.inria.fr/inria-00495666.

[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 "MEMOCODE'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] J.-F. LE TALLEC. *Extraction de modèles pour la conception de systèmes sur puce*, Université de Nice Sophia Antipolis, january 2012.

Articles in International Peer-Reviewed Journal

[17] K. DESCHINKEL, SID-AHMED-ALI. TOUATI, S. BRIAIS. *SIRALINA: efficient two-steps heuristic for storage optimisation in single period task scheduling*, in "Journal of Combinatorial Optimization", 2011, vol. 22, n^{o} 4, p. 819-844 [*DOI :* 10.1007/s10878-010-9332-8], http://hal.inria.fr/inria-00636028/en/.

[18] L. GEORGE, P. COURBIN, Y. SOREL. *Job vs. portioned partitioning for the earliest deadline first semi-partitioned scheduling*, in "Journal of Systems Architecture", 2011, vol. 57, n^{o} 5, p. 518 - 535, http://www-rocq.inria.fr/syndex/publications/pubs/jsa11/jsa11.pdf.

[19] F. MALLET, C. ANDRÉ, R. DE SIMONE. *Logical Time: observation vs. implementation*, in "ACM SIGSOFT Software Engineering Notes", January 2011, vol. 36, n^{o} 1, p. 1–8 [*DOI :* 10.1145/1921532.1921554], http://hal.inria.fr/inria-00576647/en.

[20] D. POTOP-BUTUCARU, Y. SOREL, R. DE SIMONE, J.-P. TALPIN. *From concurrent multi-clock programs to deterministic asynchronous implementations*, in "Fundamenta Informaticae", 2011, vol. 108, n^{o} 1-2, p. 91-118, http://www-rocq.inria.fr/syndex/publications/pubs/fi11/fi11.pdf.

Invited Conferences

[21] C. ANDRÉ. *Modèles de temps et de contraintes temporelles de MARTE et leurs applications*, in "École d'été temps réel - Brest - Août 2011", November 2011, http://idl.univ-brest.fr/etr11/ETR2011-web.pdf.

International Conferences with Proceedings

[22] X. CHEN, J. LIU, F. MALLET, Z. JIN. *Modeling Timing Requirements in Problem Frames Using CCSL*, in "18th Asia Pacific Software Engineering Conference (APSEC'11)", Ho Chi Minh Ville, Vietnam, december 2011.

[23] K. GARCÉS, J. DEANTONI, F. MALLET. *A Model-Based Approach for Reconciliation of Polychronous Execution Traces*, in "SEAA 2011 - 37th EUROMICRO Conference on Software Engineering and Advanced Applications", Oulu, Finland, IEEE, August 2011, http://hal.inria.fr/inria-00597981/en.

[24] R. GASCON, F. MALLET, J. DEANTONI. *Logical time and temporal logics: comparing UML MARTE/CCSL and PSL*, in "18th International Symposium on Temporal Representation and Reasoning (TIME'11)", Lubeck, Germany, September 2011, http://hal.inria.fr/hal-00597086/en.

[25] J.-F. LE TALLEC, R. DE SIMONE. *SCIPX: a SystemC to IP-XACT extraction tool*, in "ESLsyn : Electronic System Level Synthesis Conference", San Diego, United States, June 2011, http://hal.inria.fr/inria-00601843/en.

[26] J.-F. LE TALLEC, J. DEANTONI, R. DE SIMONE, B. FERRERO, F. MALLET, L. MAILLET-CONTOZ. *Combining SystemC, IP-XACT and UML/MARTE in model-based SoC design*, in "Workshop on Model Based Engineering for Embedded Systems Design (M-BED 2011)", Grenoble, France, March 2011, http://hal.inria.fr/inria-00601840/en.

[27] M. MAROUF, L. GEORGE, Y. SOREL. *Schedulability analysis for a combination of preemptive strict periodic tasks and sporadic tasks*, in "Proceedings of the 10th Workshop on Models and Algorithms for Planning and Scheduling Problems, MAPSP'11", Nymburk, Czech Republic, June 2011, http://www-rocq.inria.fr/syndex/publications/pubs/mapsp11/mapsp11.pdf.

[28] M. MAROUF, Y. SOREL. *Scheduling non-preemptive hard real-time tasks with strict periods*, in "Proceedings of 16th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA'11", Toulouse, France, September 2011, http://www-rocq.inria.fr/syndex/publications/pubs/etfa11/etfa11.pdf.

[29] F. NDOYE, Y. SOREL. *Preemptive Multiprocessor Real-Time Scheduling with Exact Preemption Cost*, in "Proceedings of 5th Junior Researcher Workshop on Real-Time Computing, JRWRTC'11, in conjunction with the 18th International conference on Real-Time and Network Systems, RTNS'11", Nantes, France, September 2011, http://www-rocq.inria.fr/syndex/publications/pubs/jrwrtc11/jrwrtc11.pdf.

[30] V. PAPAILIOPOULOU, D. POTOP-BUTUCARU, Y. SOREL, R. DE SIMONE, L. BESNARD, J.-P. TALPIN. *From concurrent multi-clock programs to concurrent multi-threaded implementations*, in "ESLsyn : Electronic System Level Synthesis Conference", San Diego, United States, June 2011.

[31] M.-A. PERALDI-FRATI, J. DEANTONI. *Scheduling Multi Clock Real Time Systems: From Requirements to Implementation*, in "International Symposium on Object/Component/Service-oriented Real-time Distributed Computing", Newport Beach, United States, IEEE computer society, March 2011, 50; 57, NewPort Beach [*DOI :* 10.1109/ISORC.2011.16], http://hal.inria.fr/inria-00586851/en.

[32] M.-A. PERALDI-FRATI, D. KARLSSON, A. HAMANN, S. KUNTZ, J. NORDLANDER. *The TIMMO-2-USE project: Time modeling and analysis to use*, in "ERTS2012 International Congres on Embedded Real Time Software and Systems", Toulouse, France, February 2012, 10 pages, http://hal.inria.fr/hal-00649781/en/.

[33] L. YIN, F. MALLET, J. LIU, R. DE SIMONE. *Verification of MARTE/CCSL Time Requirements in Promela/SPIN*, in "16th IEEE International Conference on Engineering of Complex Computer Systems, ICECCS 2011", 2011, p. 65–74, http://dx.doi.org/10.1109/ICECCS.2011.14.

National Conferences with Proceeding

[34] L. CUCU-GROSJEAN, Y. SOREL. *A schedulability test for real-time dependant periodic task systems with latency constraints*, in "12e congrès annuel de la Société française de Recherche Opérationnelle et d'Aide à la Décision", Saint Etienne, France, March 2011, http://hal.inria.fr/inria-00584253/en.

[35] F. FAUBERTEAU, L. GEORGE, D. MASSON, S. MIDONNET. *Ordonnancement multiprocesseur global basé sur la laxité avec migrations restreintes*, in "ROADEF'11 - 12ème congrès annuel de la Société française de Recherche Opérationnelle et d'Aide à la Décision", Saint-Étienne, France, 2011, 2 pp., http://hal.inria.fr/hal-00620394/en.

[36] C. GOMEZ, J. DEANTONI, F. MALLET. *Semantic Multi-View model for Low-Power*, in "Journées nationales IDM, CAL, et du GDR GPL", Lille, France, June 2011, 19, 5 pages, http://hal.inria.fr/hal-00596239/en.

Conferences without Proceedings

[37] L. SANTINELLI, L. CUCU-GROSJEAN, L. GEORGE. *Probabilistic Sensitivity Analysis*, in "the 2nd International Real-Time Scheduling Open Problems Seminar (RTSOPS 2011)", Porto, Portugal, July 2011, http://hal.inria.fr/hal-00646999/en.

Scientific Books (or Scientific Book chapters)

[38] C. GLITIA, J. DEANTONI, F. MALLET. *Logical Time @ Work: Capturing Data Dependencies and Platform Constraints*, in "System Specification and Design Languages", T. J. J. KAŹMIERSKI, A. MORAWIEC (editors), Lecture Notes in Electrical Engineering, Springer New York, 2012, vol. 106, p. 223–238, http://dx.doi.org/10.1007/978-1-4614-1427-8_14.

[39] F. MALLET. *Logical Time @ Work for the Modeling and Analysis of Embedded Systems*, LAP LAMBERT Academic Publishing, January 2011, http://hal.inria.fr/inria-00561247/en.

Research Reports

[40] C. ANDRÉ. *Modèles de temps et de contraintes temporelles de MARTE et leurs applications*, INRIA, November 2011, n^{o} RR-7788, Cours donné à l'École d'été temps réel - Brest - Août 2011, http://hal.inria.fr/hal-00639211/en/.

[41] T. CARLE, D. POTOP-BUTUCARU. *Throughput Optimization by Software Pipelining of Conditional Reservation tables*, INRIA, April 2011, n^{o} RR-7606, http://hal.inria.fr/inria-00587319/en.

[42] R. GASCON, F. MALLET, J. DEANTONI. *Logical time and temporal logics: Comparing UML MARTE/CCSL and PSL*, INRIA, January 2011, n^{o} RR-7459, http://hal.inria.fr/inria-00540738/en.

[43] P. MEUMEU YOMSI, Y. SOREL. *An Algebraic Approach for Fixed-Priority Scheduling of Hard Real-time Systems with Exact Preemption Cost*, INRIA, 2011, n^{o} RR-7702, http://hal.inria.fr/inria-00613347/en.

[44] V. PAPAILIOPOULOU, D. POTOP-BUTUCARU, Y. SOREL, R. DE SIMONE, L. BESNARD, J.-P. TALPIN. *From concurrent multi-clock programs to concurrent multi-threaded implementations*, INRIA, March 2011, n^{o} RR-7577, http://hal.inria.fr/inria-00578585/en.

References in notes

[45] 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.

[46] A. BENVENISTE, G. BERRY. *The Synchronous Approach to Reactive and Real-Time Systems*, in "Proceedings of the IEEE", September 1991, vol. 79, n^{o} 9, p. 1270-1282.

[47] J. CARLIER, P. CHRÉTIENNE. *Problèmes d'ordonnancement*, Masson, 1988.

[48] 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.

[49] 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.

[50] J.B. DENNIS. *First Version of a Dataflow Procedure Language*, in "Lecture Notes in Computer Sci.", Springer-Verlag, 1975, vol. 19, p. 362-376.

[51] S. EDWARDS. *Languages for Digital Embedded Systems*, Kluwer, 2000.

[52] N. HALBWACHS. *Synchronous Programming of Reactive Systems*, in "Computer Aided Verification", 1998, p. 1-16, http://www-verimag.imag.fr/~halbwach/newbook.pdf.

[53] H. HEINECKE. *AUTOSAR, an industrywide initiative to manage the complexity of emerging Automotive E/E-Architecture*, in "Electronic Systems for Vehicles 2003, VDI Congress", Baden-Baden, 2003.

[54] EDWARD A. LEE, D. G. MESSERSCHMITT. *Static Scheduling of Synchronous Data Flow Programs for Digital Signal Processing*, in "IEEE Trans. Computers", 1987.

[55] C.L. LIU, J.W. LAYLAND. *Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment*, in "Journal of the ACM", 1973.