INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team aoste

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

Sophia Antipolis - Méditerranée, Paris - Rocquencourt

Theme : Embedded and Real Time Systems

Table of contents

1. Team 1
2. Overall Objectives 1
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 5
4.1. Multicore System-on-Chip design 5
4.2. Automotive and avionic embedded systems 6
5. Software 6
5.1. TimeSquare 6
5.2. K-Passa 6
5.3. SynDEx 7
5.4. SAS 7
6. New Results 8
6.1. Logical Time modeling and Clock Constraint Specification Language 8
6.2. Model-Driven Engineering for Embedded Systems: OMG UML profile MARTE 9
6.3. SoC Virtual Platform modeling, from MARTE to IP-XACT to SystemC, and back 9
6.4. Requirement modeling and traceability 10
6.5. Process Networks with regular routing schemes: Mathematical foundations and optimizations 10
6.6. Real-time implementations over precise-timed dynamic TDMA architectures 11
6.7. From Concurrent Multi-clock programs to Deterministic Asynchronous Implementations 11
6.8. Uniprocessor real-time scheduling 12
6.9. Multiprocessor real-time scheduling 12
6.10. Upgrades in SAS software 13
6.11. Upgrades in SynDEx 13
7. Contracts and Grants with Industry 13
8. Other Grants and Activities 14
8.1. Regional collaborations 14
8.2. National Initiatives 14
8.2.1. ANR RT-Simex 14
8.2.2. ANR HeLP 15
8.2.3. FUI Lambda 15
8.2.4. FUI PARSEC 15
8.2.5. ARC Triade 15
8.3. European Initiatives 16
8.3.1. ARTEMIS CESAR 16
8.3.2. ITEA OPENPROD 16
8.3.3. IST Network of Excellence ARTIST2 & ARTIST-Design 17
8.3.4. ITEA TIMMO-2-Use 17
8.4. International Initiatives 17
9. Dissemination 17
9.1. Animation of the scientific community 17
9.2. Teaching 18
10. Bibliography 18

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. Team

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]
Charles André [Professor, UNS, emeritus since September 2010, HdR]
Laurent George [Associate Professor, UPEC, on leave at INRIA, HdR]

Faculty Members

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, since September 2010]
Julien Boucaron [Temporary Engineer, until March 2010]
Benoît Ferrero [Temporary Engineer]
Simon Nivault [Temporary Engineer, from March 2010]
Meriem Zidouni [Temporary Engineer, from November 2010]
Daniel de Rauglaudre [INRIA support Engineer]
Maxence Guesdon [INRIA support Engineer]

PhD Students

Anthony Coadou [MESR scholarship]
Manel Djemal [CORDI/S scholarship, from November 2010]
Carlos Gomez Cardenas [MENESR sholarship, from October 2010]
Jean-François Le Tallec [BDE scholarship with ScaleoChip]
Mohamed Marouf [INRIA scholarship, joint with EPI IMARA]
Aamir Mehmood Khan [Pakistani government scholarship, until March 2010]

Post-Doctoral Fellows

Régis Gascon
Calin Glitia
Kelly Garces Pernett [from October 2010]
Virginia Papailiopoulou [from March 2010]

Administrative Assistants

Patricia Lachaume [TCN INRIA, part-time (50%)]
Christine Anocq [TRS INRIA, part-time (33%)]
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 increasing parallel, with dedicated hardware accelators and manycore processors.

The optimized mapping of applications onto execution platforms is still far from automatic compilation as in 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. As progress is made in the design flow towards implementation, by according the application functions to the architecural resources, a *time refinement* takes place which 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 are been progressed over the years, resulting in sofware environments such as SYNDEX or TimeSquare.

3. Scientific Foundations

3.1. Models of Computation and Communication (MoCCs)

Participants: Charles André, Julien Boucaron, Anthony Coadou, Robert de Simone, 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 [46], [48], [41],[12], [3], [10] share one main characteristics: they are specificed in a self-timed or loosely timed fashion (say 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 [42], [40] (consider SDF balance equations for instance [50]).

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 [43], or N-synchronous processes [44], with some of our contributions [4].

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 [32].

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, to benefit from potential distributed implementation without impairing the penalty of high amount of signal communications which often may plague parallelized versions of programs to run slower, and with greater power consumption, as the original ones. This general line of research has come to be known as Endochrony, with some of our contributions [9].

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 expresed 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 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, specially 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, 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 AAA methodology, first proposed by Yves Sorel, now 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: Yves Sorel, Laurent George.

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 [45]. 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 [7] is obtained by an external compositional law, in which the architecture graph operates on the algorithm graph. The output is a new algorithm graph, "architecture-aware", corresponding to a refinement of the initial algorithm graph, by displaying explicit distribution and schedule informations computed according to the architecture graph resource availability.

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 [51], 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 [8]. 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-obtimal for realistic problems corresponding to industrial needs. Long time ago we proposed a very fast "greedy" heuristics [6] whose results were regularly improved, and extended with local neighborhood heuristics, or used as initial solutions for metaheuristics such as variants of "simulated annealing".

Finally, since real-time distributed architectures are prone to failures we study the possibility to tolerate faults in such systems. We focus on software redondance rather than hardware redondance to guarantee the same real-time behaviour of the system, in the presence of a certain number of faulty processors and of communication media beeing 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 progressively pertaining to these fields. The formalisms AADL (for avionics) and AutoSar [49] are providing support for this, unfortunately not always with a clean and formal semantics. Yet, some interesting issues are involved there in the mix of event-triggered and time-triggered processing means, the various related protocols, and the coexistence of periodic and aperiodic tasks, with distinct periodicity if ever. The process of scheduling and allocation of multiple heterogeneous and communicating applications onto complex embedded architectural platforms requires adequate model and synthesis/analysis/verification techniques to help the designers converge to acceptable solutions.

5. Software

5.1. TimeSquare

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

TimeSquare is a software environment for modeling and analyzing timed systems. It supports an implementation of the Time Model introduced in the MARTE UML profile (see section 3.2), and its companion Clock Constraint Specification Language (CCSL).

TimeSquare has four main functionalities:

1. interactive clock-based specifications, through dialog boxes,
2. definition/modeling of user-defined clock constraint libraries,
3. simulation and generation of a consistent trace model, using a Boolean solver,
4. attaching call-backs to the trace model to produce domain-specific feedbacks: animation of models, displaying and exploring waveforms, generation of sequence diagrams...

TimeSquare is a plug-in developed with the Eclipse Modeling Tools. It is integrated in the OpenEmbeDD platform and can be downloaded from the team site (http://www-sop.inria.fr/aoste/dev/time_square).. This software is registered by the *Agence pour la Protection des Programmes*, under the number IDDN.FR.001.170007.000.S.P.2009.001.10600, since February 11, 2009.

5.2. K-Passa

Participants: Julien Boucaron [correspondant], Anthony Coadou, 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 allows to statically schedule such graph. Analytic methods allow to compute additional buffers needed to reach maximum achievable throughput of the graph. They can compute also part of the graph that can be slow-down through addition of both integer and fractional latencies. Such extra latencies are used to alter the static-schedule to minimize for instance dynamic power peak. In the case of KRG the (ultimately k-periodic) routing patterns can also be input and transformed. KPASSA can import/export specific UML and IPXACT models compliant with TimeSquare.

K-PASSA currently relies in part on the BOOST GRAPH library for graph algorithms and Qt4 for the GUI. It also uses LP_SOLVE as its underlying integer linear solver, GNU MP for multiprecision arithmetic and Xerces for XML parsing.

This software was developed as a result of researches on Latency-Insensitive Design conducted in the context of the CIM PACA initiative, with the support of industrial partners providing motivations.

KPASSA can be downloaded on AOSTE website. This software is registered by the Agence pour la Protection des Programmes, under the number IDDN.FR.001.310003.000.S.P.2009.000.20700.

5.3. SynDEx

Participants: Maxence Guesdon, Yves Sorel [correspondant], 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, at the url: http://www.syndex.org.

The AAA methodology is described in section 3.3. Following it, SYNDEX explores the space of possible allocations (distribution and scheduling) from application elements to architecture resources and services to match the real-time requirements, using schedulability analyses and heuristic techniques. It will generate automatically distributed real-time code running on the embedded platform. 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, the real-time distributed embedded code is automatically generated from 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. Presently, executives kernels are provided for: TMS320C40, PIC18F2680, i80386, MC68332, MPC555, i80C196 and Unix/Linux workstations. Executive kernels for other processors can be ported at reasonable cost following these 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, compared to other commercial and academic softwares of the same kind, is that SAS takes into account the exact preemption cost during the schedulability analysis. Now, beside the usual real-time constraints (precedence, strict periodicity, latency, etc.) and fixed-priority scheduling policies (Rate Monotonic, Deadline Monotonic, Audsley^{++}, User priorities) that could already be taken into account, SAS has been extended to make it possible 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 named "dameid" that clearly shows the idle slots of the processor in the permanent phase.

For a schedulable task system under EDF when the exact preemtion cost is considered, the period of the permanent phase may be much longer than the classical one least commun multiple (LCM) of the periods of all tasks in the traditional scheduling theory. A specific effort has been done in SAS in order to improve the 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 displayed when the system is schedulable. In addition, another graphic, showing the response times of each task relative time, can be displayed.

The software is written in OCAML, using CAMLP5 (syntactic preprocessor) and OLIBRT (a graphic toolkit under X). Both are written by Daniel de Rauglaudre. New software upgrades for 2010 are described in section 6.10.

6. New Results

6.1. Logical Time modeling and Clock Constraint Specification Language

Participants: Charles André, Julien deAntoni, Benoît Ferrero, Régis Gascon, Calin Glitia, Carlos Gomez Cardenas, Frédéric Mallet, Robert de Simone.

This year we worked on the dissemination of our Logical Time approach (see 3.2) and the associated Time Model and CCSL language, both internally to the OMG standardization consortium as part of the MARTE profile, and externally by a number of publications.

This logical time *as and at* design time was the subject of a book chapter on correct-by-construction embedded software design [31]. It is also the main federating topic of Frédéric Mallet's Habilitation Thesis [14], defended in November 2010.

We have conducted a number of studies aimed at demonstrating the potential use and practical benefits of Logical Time and CCSL:

- In [16], [23] we establish their ability to explicitly model the dynamic behavior of various formal Models of Computation and Communication (MoCC) in a provably correct way, while being cast in a practical model-driven engineering setting.
- Domain-specific libraries have been built to allow expression and verification of logical time constraints. Focus was put on the definition of derived time operators in the applicative domains of real-time systems [21], and systems on chip design [33].
- Concrete transformations from (subsets of) CCSL properties towards operational observers written in HDL syntax (VHDL, SystemC, or Esterel) were implemented and are available as code libraries. They support automated verification of safety properties expressed in CCSL, by joint simulation in these target languages [19].
- We studied the specialization of the platform-based approach, with machine-assisted mapping of applications towards execution architectures, to the case of intensive data processing and Array-OL, with the support of CCSL for explicit formal modeling of these data dependencies [23].

In previous years we had only considered for CCSL the definition of efficient simulation schemes, as implemented in TimeSquare 5.1. This allowed verification by single-run simulations. This year we turned to perspectives of exhaustive verification and model-checking, at least in the finite-state case. This was only preliminary, and the topic will be progressed further in the next years. Related to this goal we currently provided transformations into two distinct formalisms: Polychrony-SME/Signal on the one hand, the linear fragment of *Temporal Logic* as in PSL/SuGaR on the other one. In the former case, we specifically considered the issues of SIGNAL's so-called *clock hierarchization* mechanism [30], and the relative expressivity of CCSL *vs.* this language. In the second case we studied the transformation from declarative CCSL properties to operational automata constructs; this led to the consideration of temporal modality quantifiers as potential extensions to the CCSL formalim [38].

6.2. Model-Driven Engineering for Embedded Systems: OMG UML profile MARTE

Participants: Julien deAntoni, Frédéric Mallet.

We continued our efforts towards the standardization of MARTE at OMG. Julien deAntoni attended the OMG Technical Meeting in Boston in September 2010, and Frédéric Mallet attended the one in Santa Clara in December. Version 1.1 of the standard was adopted in September. Main updates regarding our work deal with the qualification of UML events as logical clocks. Other features are the combination of our logical time framework with the timing notions exploited in schedulability and performance analysis in other parts of the standard. A better integration of MARTE and SysML in the Allocation Model was also successfully proposed [17], as a result of the FUI LAMBDA project collaboration (see 8.2.3).

We conducted several experiments with software prototype implementations, mostly around the PAPYRUS UML profile modeler. They were mostly aimed at improving the efficient graphical representation of stereotypes (and the combination of them) annotating UML elements. Appealing visual traces for simulation report were also introduced. The motivation here was to try and make the UML look just as natural and intuitive as the ones in dedicated proprietary design tools. A number of intern students from the University of Nice/Sophia-Antipolis collaborated to this effort. Next we should work on the integration of such prototypes into a more stable software basis, partially centered around PAPYRUS (but also possibly stand-alone). This will be done in part by our current ADT support engineer Nicolas Chleq. Apart from the PAPYRUS version of MARTE, our Time Model was also integrated into the commercial tool OBEO-DESIGNER, as part of the ANR RT-Simex project (see 8.2.1).

Meanwhile, some of our former contributions to the foundations of automatic profile building from metamodels were accepted for publication [25]. The framework of MARTE was extensively used in our descriptions of how its Time Model and Clock Constraints could be used to provide explicit precise timed semantics to various formalisms, such as AADL and East-ADL2. This was reported in a book chapter [31] (see 6.1). Finally, we disseminated our results on MARTE modeling in the context of a French-Vietnamese academic collaboration [28].

6.3. SoC Virtual Platform modeling, from MARTE to IP-XACT to SystemC, and back

Participants: Charles André, Julien deAntoni, Benoît Ferrero, Jean-François Le Tallec, Aamir Mehmood Khan, Frédéric Mallet, Robert de Simone.

The new industrial standard IP-XACT is being proposed for easy integration of black-box software or hardware component IP blocks into virtual SoC platforms. It is essentially an ADL (Architecture Description Language), which should be coupled with SystemC, both at RTL and TLM level. This initiative met some of our concerns in many regards: first, we were concerned with the field as part of our CIM PACA implications; second, easy component assembly is part of model-based design purposes; third, the temporal issues of simulation semantics in SystemC could be compared and contrasted with our approach of synchronous reactive modeling lifted to logical time.

IP-XACT is currently a very ad-hoc standard, developed by people with down-to-earth concrete needs, but low expertise in model-driven high-level software engineering. Therefore, our first attempt was to provide a comprehensive metamodel for it, using MARTE as support [33]. Indeed, this would allow us to introduce in a clean fashion numbers of non-functional aspects, such as energy consumption and performance charactistics, which are swept under the rug in current IP-XACT as *"vendor extensions"*. This work on a profile specialization from MARTE to IP-XACT was the topic of Aamir Mehmood Kahn thesis, defended in March [15].

Despite the fact that our ultimate goal was to start model applications in MARTE (with adequate non-functional property extensions, and logical timing aspects for behaviors), then translate towards a combination of IP-XACT and SystemC for inclusion into EDA CAD environments, we took the current option to define transformations from SystemC to IP-XACT (for the structural interface of components), and then MARTE. We did so because of existing component and platform libraries written in SystemC, which we wanted to translate into our formalims to populate our approach. A translator for SystemC RTL to IP-XACT was thus defined and realized. It is based originally on an open-source software from VERIMAG named PINAVM. The main difficulty of the approach is that the actual structural interface of a SystemC module is in fact *dynamically constructed* during a so-called *elaboration phase*, which configures a number of parameters. Our new SCIPX prototype borrows through PINAVM to the LLVM C++ compiler, but also to DOXYGEN for static analysis of code [24]. This result was obtained in the course of the (on-going) PhD thesis of Jean-François Le Tallec.

These research developments were funded in part by the ID/TL-M collaboration with ST Microelectronics (see 7.1), and the ANR HeLP project.

6.4. Requirement modeling and traceability

Participants: Julien deAntoni, Kelly Garces Pernett, Frédéric Mallet, Marie-Agnès Peraldi Frati.

As a result of the ANR MeMVaTEx project, closed at the end of 2009, a dedicated UML profile for requirements modeling and their traceability had been defined. This year we continued work on its further evolution. The profile offers various links between requirements and system specifications, or between various requirement themselves (as for refinement for instance). The connections between requirements and their validation, simulation, and analysis acording to specific tools such as TimeSquare or SynDEx are explicitly explored.

Results from analysis tools may have an impact back on the requirements elaboration. In the ANR project RT-Simex we handle this feedback, switching the role of requirements between specification and assertion, in a way that makes it possible to verify if a design is correct with regards to these requirements (see [21] for details).

We developed an Eclipse plugin around these ideas, called RVT (for *"Requirement, Validation and Traceability"*). It allows handling and management of dependency links between requirements and models, and support validation procedures associated with such requirements (tests, simulation, formal analysis). Results were published in [22] and [27].

6.5. Process Networks with regular routing schemes: Mathematical foundations and optimizations

Participants: Julien Boucaron, Anthony Coadou, Robert de Simone.

We considered at length a new dataflow process network model, namely K-periodic Routing Extended Marked Graphs (KRGs). The model is original in that it allows route switching nodes (multiplers/demultiplexers), but with specific routing patterns provided as ultimately k-periodic infinite binary words. They are therefore strict extensions of Marked Graphs, specialize Boolean DataFlow models from UC. Berkeley Ptolemy, and provide primitive operators that may generate Cyclo-Static DataFlow models, as well as a specific class of Kahn Proces Networks.

The precise mathematical definition of all parts of KRGs allows to establish a number of formal property for analysis: balance equations for matching consumption/production of data can be extracted by faithful abstraction into SDF domains (thereby solving safety issues). Expansions into finite quasi-Marked Graphs can also allow to solve liveness issues and existence of sufficient data in initial markings (a quasi-Marked Graph consists of an acyclic initial/transitory MG, followed by a single-loop periodic/stationary one). Finally, expanded canonical normal forms for the nodes representing the interconnect routing networks can also be established. With this result, the functional equivalence of two such networks can always be reducible to a finite number of local algebraic graph transformations. In other words, the network topology can be transformed and optimized to keep the same computations with the same dependencies, and just work on downsizing the buffering queues in these interconnects.

Next step shall consist in merging our results on k-periodic schedules and optimization of the previous years, to combine them with similar k-periodic routings.

This activity resulted in the PhD thesis of Anthony Coadou [13], defended in December 2010, and a book chapter [32]. A technical report is also submitted for publication [36]

6.6. Real-time implementations over precise-timed dynamic TDMA architectures

Participant: Dumitru Potop Butucaru.

The static schedule tables produced by SynDEx are traditionally implemented using event-driven techniques. We considered alternative time-triggered (cycle-based) implementations, of various flavors. Time-triggered architectures are fairly developed in the automotive and avionic domains (ARINC 653, TTA, FlexRay, etc.), and have been promoted recently for design of embedded systems *"temporally correct by construction"* with precise timed (PRET) architectures [47]) .

This year we developed a technique which automatically synthesize distributed time-triggered implementations from synchronous dataflow models, using a *dynamic* TDMA communication policy. We implemented this approach by connecting a SynDEx-like prototype to the Network Code toolset, from Waterloo University, Canada. Network Code can be considered as an "assembly code" level in this setting. The main difficulty was to ensure the functional and real-time correctness of the schedule transformation, in presence of clock drifts (which are abstracted away in the scheduling model of SynDEx). The result is a seamless, automatic model transformation flow from high-level specifications to time-triggered implementations [29].

This work was presented in a EmSoft 2010 paper, jointly with Akramul Azim and Sebastian Fischmeister, from the University of Waterloo, Canada. A proposal for INRIA associated-team collaboration along these lines was applied for.

6.7. From Concurrent Multi-clock programs to Deterministic Asynchronous Implementations

Participants: Dumitru Potop Butucaru, Virginia Papailiopoulou, Robert de Simone, Yves Sorel.

Large synchronous programs in SIGNAL or ESTEREL are in fact polychronous (or *"multiclock synchronous"*). From this remark one may hope to extract potential behavioral independence between concurrent processes. The goal here is to devise asynchronous or distributed implementations that retain the original functionality, while efficient in terms of inter-process signaling and communications. This line of research has come to be known as (weak or strong) *endochrony* checking, and we have amply contributed to it in recent years (see 3.1.2).

Central to this approach is the general algorithm for checking weak endochrony on multi-clock synchronous programs. The method is based on the construction of so-called generator sets, which contain minimal synchronization patterns that characterize all possible reactions of a multi-clocked program. This year we studied important improvements to this algorithm. This was achieved by:

- Defining efficient data structures and analysis algorithms based on a symbolic and hierarchic representation of the generator sets.
- Providing a prototype analysis tool and a link with the Signal/Polychrony high-level specification language.

We are currently working on 1) further optimization of the hierarchical generator sets data structures, 2) effective generation of multi-threaded GALS wrappers for Signal programs, and 3) application of these results for the generation of simpler communication protocols in the SynDEx tool.

A partial progress report on these results was presented at the Synchron 2010 seminar week. This work was conducted in connection with the ESPRESSO EPI, and partly funded by INRIA ARC Triade for the postdoctoral position of Virginia Papailiopoulou

6.8. Uniprocessor real-time scheduling

Participants: Laurent George, Mohamed Marouf, Yves Sorel.

Our work on uniprocessor real-time scheduling considers separately the cases of preemptive and non-preemptive task systems.

In the non-preemptive case a necessary and sufficient condition for schedulability of two dependent tasks exists. It is only a sufficient condition in the case of more than two tasks, in practice rather restrictive. This year we proposed a weaker schedulability condition which applies to a set of tasks, and a corresponding scheduling heuristics (the general problem is NP-Hard). This work is reported in [26].

We compared and contrasted non-preemptive Fixed Priority (FP-NP) scheduling w.r.t. the optimal non-preemptive dynamic priority Earliest Deadline First scheduling (EDF-NP), known optimal for non concrete task sets (where first release time is unknown a priori). We considered non preemptive sporadic tasks defined by their WCET, period and deadline. We characterized the efficiency of FP-NP by the notion of processor speedup factor, defined as the factor by which the processor speed should be increased so that a schedulable task with EDF becomes schedulable with FP-NP. We showed tit to be comprised between 1.76322 and 2 [20].

In the preemptive case we studied dynamic reconfiguration. A reconfiguration can be decided off-line, to determine if a task can be run on a new processor while preserving its real-time constraints, or on-line, to handle temporal faults originated from hardware or software. The task model is preemptive sporadic. We considered FP and EDF scheduling. We showed that sensitivity analysis can be used to determine if a reconfiguration is possible, by characterizing the maximum acceptable deviations of the task parameters (WCETs, periods and deadlines) that do not compromise the schedulability of tasks.

6.9. Multiprocessor real-time scheduling

Participants: Laurent George, Mohamed Marouf, Yves Sorel.

Our work on multiprocessor real-time scheduling was based until now on non-preemptive scheduling, as this approach is the best suited for critical (hard) real-time systems. Indeed, non preemptive scheduling prevents from approximating the RTOS (Real-Time Operating System) cost, which is difficult to compute (preemption cost varies with every instance of each task). This difficulty may lead to miss some deadlines at real-time execution, or at least to waste resources since WCETs (Worst Case Execution Time) for tasks must include this approximation cost. For that reason we recently proposed a way to compute the exact RTOS extra cost, and to consider it in the classical real-time schedulability conditions for scheduling algorithms (Rate Monotonic, Deadline Monotonic, Earliest Deadline First,...).

This year we investigated the potential benefits of these results. In a first step we focused on the partitioned scheduling approach, which reduces scheduling to (i) finding a partition assigning tasks onto processors; (ii) finding a local uniprocessor scheduling on each processor for its assigned tasks. We introduced the exact RTOS cost in schedulability conditions for the classical scheduling heuristics in that case (the problem is NP-hard, equivalent to bin-packing): Next-Fit, First-Fit, Best-Fit, Worst-Fit, and we performed benchmark performance comparisons.

In a second step, we propose fault-tolerance scheduling techniques for multiprocessor real-time embedded systems, such as the ones that are automatically implemented with the SynDEx software. These techniques will be tested on electric vehicles developed in the IMARA EPI. This year we have focused on strategies to tolerate permanent faults on CAN buses using software active and passive redundancy (since active redundancy leads to simpler error detection mechanisms, we focused mainly on that technique). We designed a fault-tolerance testbench, containing two dsPICs connected with two CAN buses, a PeakCAN-USB interface for programming and supervision, and a compiler/supervisor software running under Windows. We solved the problem of error detection for the CAN bus using bus error interruptions and watchdogs, and we tested the software active redundancy technique in this case.

6.10. Upgrades in SAS software

Participants: Daniel de Rauglaudre, Yves Sorel.

The core of SAS (which performs the uniprocessor schedulability analysis taking into account the exact cost of the RTOS, see 5.4), was extracted as a separate function. We studied how this function could efficiently be taken into account, and integrated inside the three partitioned real-time scheduling heuristics Next-Fit, First-Fit, Best-Fit and Worst-Fit, and we compared their performances.

Besides, the graphical user interface of SAS was made compliant with new standards as well as consistent with the one of SynDEx, in the perspectives of a future merge of SAS into SynDEx.

6.11. Upgrades in SynDEx

Participants: Maxence Guesdon, Yves Sorel, Meriem Zidouni.

Apart from minor bug fixes in SynDEx v7, we studied an innovative software architecture, which will lead to a new SynDEx 8. The goal is to be able to include much more easily new theoretical research results. As a motivating example, our results on heuristics for multi-periodic applications 6.8 were readily included in this upcoming SynDEx v8 prototype.

This work was partially conducted by Meriem Zidoun, temporary engineer on the COTROS ADT (Action de Développement Technologique).

7. Contracts and Grants with Industry

7.1. ID/TL-M project with ST Microelectronics

Participants: Charles André, Julien Boucaron, Robert de Simone, Benoît Ferrero, Aamir Mehmood Khan.

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). While SystemC is a de-facto standard in this domain, it suffers a number of lacks. One is the absence of clear formal semantics (unlike Esterel), which largely proscribes high-level synthesis; the second is the absence, at the transaction-level modeling (TLM) level, of a clear associated interface, as an Architecture Description Language (ADL), which could support annotations for extra non-functional properties.

The ongoing standard IP-XACT is a candidate for being such an ADL but, being developed by a consortium of industrial partners with too low-level intents, it does not fit with all our aim perfectly. As a result we studied the specialization of MARTE to encompass most existing features of IP-XACT, while being truly extensible to non-functional properties with genericity. As a by-product, formal semantic definition using our CCSL language and MARTE Time model are becoming feasible, with synchronous artefacts borrrowed from Esterel as guidelines.

This year we worked more specifically on a prototype tool extracting IP-XACT structural description from SystemC interface and elaboration phase. The tool is itself built from another public-domain software, named PinaVM, from VERIMAG. It works currently on RTL SystemC code, and we intent to lift it to TLM level. While this transformation may seem to go in the counter direction of our goals, we hope to use it to provide easily IP-XACT wrappers around existing SystemC library components, and then MARTE structural descriptions from IP-XACT (following our work in previous years). Then we shall consider in the next future the addition of extra-functional properties at the MARTE level on the resulting models. This should be done in connection with our efforts on dual timed-functional and energy-thermal modeling as part of the HeLP ANR project (see 8.2.2).

8. Other Grants and Activities

8.1. Regional collaborations

8.1.1. CIM PACA

Participants: Jean-François Le Tallec, Julien Boucaron, Aamir Mehmood Khan, Robert de Simone.

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 actively participating in the Design Platform (one of the three platforms launched in this context).

Inside this platform we were coordinator of the Sys2RTL project, focused on methodological flows for high-level SoC synthesis. Participants were Texas Instruments, ST-Ericsson, Synopsys, Esterel-EDA, and Scaleo Chip as industrial partners, INRIA, I3S (CNRS/UNSA) and ENST on the academic side. The project reached its conclusion this year.

Jean-François Le Tallec conducted his PhD thesis on a BDE funding in connection with Scaleo Chip, a local SME company developing SoC platform simulators. The PhD topic was to investigate new *virtual* platform environments at ESL TLM level, and their relation to formal modeling in multiclock ESTEREL.We have considered possible future extensions leading to a better and more realistic use of Virtual Hardware Platforms (VHP). In this direction we addressed the Synopsys tools, which were acquired as mutualized softwares by the Design Platform: INNOVATOR, SYSTEM STUDIO, COREASSEMBLER, COREBUILDER.

This PhD work will only reach its conclusion after the end of the Sys2RTL project. Some of the premisses achieved in this context are now progressed in the setting of the ANR HeLP project, which will finance the PhD extension.

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 editing company.

Some early results of this project were published as conference paper [21]. We hired in October Kelly Garces Pernett on a post-doctoral position in this project.

8.2.2. ANR HeLP

Participants: Julien Boucaron, 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 two-fold: 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 .

8.2.3. FUI Lambda

Participants: Charles André, Julien Deantoni, Robert de Simone, 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.)

We worked in two directions:

- Following the lines of the translation defined last year from CCSL requirements to ESTEREL observers, we built this year a similar library of SCADE SCADE patterns enabling such a translation in the case of this language. This technical shift was in part due to the replacement of the Esterel EDA company (developing Esterel Studio) by Esterel Technologies (owner of SCADE suite).
- We strenghtened further the convergence between MARTE and SysML [17], mostly at the level of allocation model (which links application functions to architectural resources and services). Meanwhile, an action has been taken by the project members towards OMG, in order to make UML itself compatible with such a notion of allocation.

8.2.4. FUI PARSEC

Participants: Dumitru Potop Butucaru, Yves Sorel.

The Parsec project is scheduled to last from 2010 to 2012. This a large project with The partners of the project are 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.

8.2.5. ARC Triade

Participants: Dumitru Potop Butucaru, Virginia Papailiopoulou, Robert de Simone, Yves Sorel.

The ARC Triade is instrumental in enforcing joint actions between EPIs Espresso, Aoste, and DaRT, in the comparative evolutions and combinations of formalisms and approaches, with a definite impact on further joint partnership into contractual collaborations, such as ARTEMIS CESAR (see 8.3.1) or nano2012 ID/TL-M (see 7.1).

This year we acknowledge INRIA support for the postdoctoral period of Virginia Papailiopoulou, which is working on a tool for extraction of weakly endochronous criteria and corresponding implementations, from polychronous models as inpired by Signal/Polychrony from EPI Espresso. A number of internal meetings between ARC Triade partners were held this year (mostly in Paris), with invited talks and demos given at the INRIA-industry meeting days in Toulouse.

8.3. European Initiatives

8.3.1. ARTEMIS CESAR

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

CESAR is a large project with over 60 participants, mostly important industrial partners from avionics, railways, and automotive domains. It is currently the flagship project of the European ARTEMIS programme. It aims at defining a common integrated environment to organize collaborative use of many tools and methods relevant to embedded system design, positioned according to effective design flows validated by industrial needs and experience.

The project is split between 6 main subprojects:

- SP1 is devoted to the construction of a common reference technological platform (RTP), on which all tools and models are to be plugged. It borrows partly on experience gained in the former OPENEMBEDD RNTL platform;
- SP2 deals with requirements engineering and the issues of consistency when conducting large development projects inside multiple organizations (evolving product lines, contracts, non-functional property requirements, and so on);
- SP3 deals with individual tools and methods, both from classical approaches and from innovative academic sources that are found useful to support prospective design flows and thus be connected onto the RTP. EPI Aoste is most specifically leading this part for INRIA;
- SP5/6/7 are devoted to industrial domains, where industrial partners provide use cases and potential scenarios, showing needs (and evaluating solutions in a latter phase).

We promoted the appraoch of platform-based design and safe-by-construction approaches in SP3, which resulted in a proposal to integrate SYNDEX and its connections to SIGNAL/POLYCHRONY in the third version of the RTP (first version consisted mostly of existing industrial tools interconnected together, and v2 contains POLYCHRONY/SME itself. A joint demo was put up and presented to the ARTEMIS JU evaluation board in October (in Graz, Austria), with a sucessful appraisal by this Board.

Meanwhile, we started collaboration with reserach groups at Thales TRT to evaluate the update and connection of their proposed ARCADIA development methods with its related tool suite, into the CESAR framework.

The project provides funding for Regis Gascon post-doctoral position.

8.3.2. ITEA OPENPROD

Participants: Simon Nivault, Yves Sorel.

The partners of this project are Bosch, Siemens, SKF, Nokia, IFP, EDF, PSA, EADS, LMS Imagine, VTT, CEA, Fraunhofer, etc.

The project aims at providing an open environment for model-driven rapid systems development, modeling, and simulation of complete products, integrating open-source tools (OpenModelica, etc.) together with existing industrial tools, all under the Eclipse software development platform. Applications shoould be conducted also as use-cases for the platform.

This year, in cooperation with IFP Energie Nouvelle, we delivered the specification of the Scicoslab/Scicos code generation solution for their multicore-based platform. This involved both SynDEx and the specific Scicos/SynDEx gateway, and so two distinct upgrade adaptations were conducted. First, we provided the two following SynDEx executive kernels: one for Linux with support for multicore architectures with shared memory communications, another for Windows and RTX again with support for multi-core architecture, presently under some limitations. Second, the Scicos/SynDEx gateway has been upgraded to match the recent updates of SynDEx and Scicos, and integrated in the last version of Scicos and Scicoslab 4.4.b7 ; it is available from the SynDEx web site.

8.3.3. IST Network of Excellence ARTIST2 & ARTIST-Design

Participants: Robert de Simone, Frédéric Mallet.

We attended this year several events sponsored by the ARTISTDESIGN Network of Excellence, such as the workshop UML& AADL.

We also got an important financial support from this NoE for the organisation of the 17th Synchron seminar week, held in Fréjus in late November 2010.

8.3.4. ITEA TIMMO-2-Use

Participants: Marie-Agnès Peraldi Frati, Dumitru Potop Butucaru, Yves Sorel, Julien deAntoni.

This project was officially labeled in 2010, and funded by the French government starting November 2010. It is the follow-up of the TIMMO ITEA project (where French partners did not contribute, as it was not funded on the French side), itself the informal continuation of the ITEA EAST-ADL2 project (where French and Greman partners set up the foundations of real-time model-based design and component engineering for automotive applications). While this former project was a source of inspiration for the AutoSar indiustria standard in the automotive field, the present project aims at enhancing the work on formal model-based scheduling approaches to maintain them in line with AutoSar further evolutions, and also to tackle advanced topics in requirement engineering and product line configurable modeling.

The kick-off meeting was held in Potsdam (Germany), in November 2010.

8.4. International Initiatives

8.4.1. CNRS-NSFC collaboration

As part of a lightweight French-Chinese collaboration with the Software Engineering Institute at East China Normal University (SEI-ECNU, Shanghai), we received Prof. Jing Liu for a two week period in October, while Robert de Simone and Frédéric Mallet visited them in return for two weeks in November. The collaboration topics deal with the use of verification tools (such as Promela/Spin) in connection with Logical Time models and CCSL.

9. Dissemination

9.1. Animation of the scientific community

- Robert de Simone was programme committee member for MemoCode'10, FDL'10, SIES'10, and EmSoft'10. He was main organizer of the 17th annual Synchron seminar week, held in Frejus in late November 2010. He was on the Selection Board of Experts for the ANR programme ARPEGE 2010. He was reviewer for NIcolas Coste PhD thesis (ENSIMAG). He is one of the two INRIA representative to the Doctoral School Council of the university of Nice/Sophia-Antipolis (UNS) . He represents INRIA in the CA board of ARCSIS, the ruling association for the CIM PACA, as well being a member of its Strategic Council. He holds similar positions in CIM PACA Design Platform non-profit organization.

- Charles André was external jury member for the PhD theses of Florent Peres (Toulouse) and Grégory Faraut (Lyon), and local jury member for the defenses of Aamir Mehmood Kahn (PhD) and Frédéric Mallet (HDR).
- Marie-Agnès Peraldi-Frati is a member of the CNRS/I3S laboratory council and member of the CERTEC (conseil d'études et de la recherche technologique) of the IUT of Nice-Sophia Antipolis. She was local organizer of the Colloquium RNSUD 2O1O, and the Synchron2010 seminar.
- Frédéric Mallet was jury member for the PhD thesis of Kelly Garcès, Ecole des Mines de Nantes. He is also a member of the steering committee of the CNRS Action IDM. He was program committee member of TASE 2010. He is elected member to the Joint Technical Committee of the University of Nice/Sophia-Antipolis.
- Dumitru Potop-Butucaru was program committee member of ACSD 2010. Within INRIA Rocquencourt, he is local member of the scientific recruitement committee for non-permanent positions (*Détachements, délégations, post-doctorants*).
- Laurent George was program committee member for RTNS'10, WFCS'10, ICONS'10, ICAS'10 and ICNS'10. He co-founded this year a new national research group ACTRISS on real-time systems supported by GDR ASR from CNRS.
- Yves Sorel was program chair for RTNS'10, and organized the INRIA-Industry Days (RII) on Aeronautics and Aerospace Industries: Modelling and Safe Systems. He was program committee member for: CRTS/RTSS'10, DASIP'10. He is member of the Steering Committee of the Competitivity Cluster OCDS/SYSTEM@TIC Paris-Region. He is in the Editorial Board of the INIST-CNRS journal Traitement du signal.

9.2. Teaching

- Robert de Simone gave a course on Formal Methods and Models for Embedded Systems in the STIC Master program of the university of Nice-Sophia Antipolis (UNS), for approximately 15h.
- Julien DeAntoni lectures in different cursus of PolyTech Sophia, the technical engineer programme of UNS (for a total amount of 206 h). In particular, he gives two courses (with lab sessions) at final year (second year Master level), one on micro-controller and programming for Real-Time operating system, the other on model-driven engineering for embedded real time systems. He also teaches object-oriented programming in C++ in the 4^{th} year of mathematical modelling cursus, and linux shell programming in the second year of the engineering cursus. Finally, he is in charge of the organization of final project assignments for the engineer cursus, gathering project topics for the students.
- Frédéric Mallet lectures on various courses as an associate professor at the University of Nice/Sophia-Antipolis and PolyTech Sophia, from the first year of the Bachelor Degree (Licence) to the master. He teaches software engineering, computer architecture, object-oriented programming and model-driven engineering. He is also administrative responsible of the Master 1 MIAGE cursus.
- Marie-Agnès Peraldi-Frati gives different courses (Systems and Networks, Programming, Web development, Computer Architecture) at L1 (undergraduate) level, in the IUT Computer Science Department.

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, F. MALLET, R. DE SIMONE. *Modeling Time(s)*, in "MoDELS'2007 10th Intern. Conf. on Model Driven Engineering Languages and Systems", 2007.

[3] 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.

[4] 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.

[5] L. CUCU, 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] D. POTOP-BUTUCARU, S. EDWARDS, G. BERRY. *Compiling Esterel*, Springer, 2007.

[11] D. POTOP-BUTUCARU, R. DE SIMONE. *Optimizations for Faster Execution of Esterel Programs*, in "MEMOCODE'03", 2003.

[12] R. DE SIMONE, D. POTOP-BUTUCARU, JEAN-PIERRE. TALPIN. *8*, in "The Synchronous Hypothesis and Synchronous Languages", CRC Press, 2005.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[13] A. COADOU. *Réseaux de processus flot de données avec routage pour la modélisation de systèmes embarqués*, Université de Nice Sophia Antipolis, décembre 2010.

[14] F. MALLET. *Logical Time in Model Driven Engineering*, Université Nice Sophia-Antipolis, November 2010, Habilitation à Diriger des Recherches.

[15] A. MEHMOOD KHAN. *Model-Based Design for On-Chip Systems: using and extending Marte and IP-Xact*, Université de Nice/Sophia-Antipolis, March 2010.

Articles in International Peer-Reviewed Journal

[16] F. MALLET, J. DEANTONI, C. ANDRÉ, R. DE SIMONE. *The Clock Constraint Specification Language for building timed causality models*, in "Innovations in Systems and Software Engineering", Mar 2010, vol. 6, n° 1–2, p. 99–106, http://hal.inria.fr/inria-00464894.

Articles in National Peer-Reviewed Journal

[17] S. DEMATHIEU, Y. BERNARD, L. MAILLET-CONTOZ, M. BOURDING, F. MALLET. *Mise en œuvre de l'IDM pour la conception de systèmes embarqués : premiers résultats et perspectives du projet Lambda*, in "Génie Logiciel", June 2010, vol. 93, n° 90, p. 38–42, http://hal.inria.fr/inria-00482727/.

[18] F. MALLET, C. ANDRÉ, F. LAGARDE. *Un processus automatique pour concevoir les profils UML*, in "Technique et Science Informatiques (TSI)", May 2010, vol. 29, n° 5, p. 391–419, http://hal.inria.fr/inria-00482745.

International Peer-Reviewed Conference/Proceedings

[19] C. ANDRÉ, F. MALLET, J. DEANTONI. *VHDL observers for clock constraint checking*, in "Industrial Embedded Systems (SIES), 2010 International Symposium on", July 2010, p. 98–107, http://dx.doi.org/10.1109/SIES.2010.5551372.

[20] R. DAVIS, L. GEORGE, P. COURBIN. *Quantifying the Sub-optimality of Uniprocessor Fixed Priority Non-Preemptive Scheduling*, in "Proceedings of 18th International Conference on Real-Time and Network Systems, RTNS'10", Toulouse, France, Nov 2010, http://hal.archives-ouvertes.fr/inria-00536363/en/.

[21] J. DEANTONI, F. MALLET, F. THOMAS, G. REYDET, J.-P. BABAU, C. MRAIDHA, L. GAUTHIER, L. RIOUX, N. SORDON. *RT-simex: retro-analysis of execution traces*, in "Proceedings of the eighteenth ACM SIGSOFT international symposium on Foundations of software engineering", New York, NY, USA, FSE '10, ACM, 2010, p. 377–378, http://doi.acm.org/10.1145/1882291.1882357.

[22] H. DUBOIS, M.-A. PERALDI-FRATI, F. LAKHAL. *A model for requirements traceability in an heterogeneous model-based design process. Application to automotive embedded systems*, in "ICECCS'10", Royaume-Uni Oxford, IEEE Computer Society, Mar 2010, vol. ISBN:978-0-7695-4015-3, p. 233–242, http://hal.inria.fr/inria-00484073.

[23] C. GLITIA, J. DEANTONI, F. MALLET. *Logical time at work: capturing data dependencies and platform constraints*, in "Proceedings of International Forum for Design Languages (FDL 2010)", Southampton, UK, sep 2010, p. 240–246, http://hal.inria.fr/inria-00545866/.

[24] 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 "Proceedings of the 2011 Workshop on Model Based Engineering for Embedded Systems Design", Grenoble, France, M-BED'2011, March 2011, to appear.

[25] F. MALLET, F. LAGARDE, C. ANDRÉ, S. GÉRARD, F. TERRIER. *An Automated Process for Implementing Multilevel Domain Models*, in "Software Language Engineering", États-Unis Denver, Colorado, Springer-Verlag Berlin Heidelberg, Mar 2010, p. 314–333, http://hal.inria.fr/inria-00464880.

[26] M. MAROUF, Y. SOREL. *Schedulability conditions for non-preemptive hard real-time tasks with strict period*, in "Proceedings of 18th International Conference on Real-Time and Network Systems, RTNS'10", Toulouse, France, Nov 2010, http://hal.archives-ouvertes.fr/hal-00544496/en/.

[27] M.-A. PERALDI-FRATI, A. ALBINET. *Requirement traceability in safety critical systems*, in "Supp Proceeding of the 8th European Dependable Computing Conference, EDCC'10", Valencia, Spain, April 2010, p. 11–14.

[28] M.-A. PERALDI-FRATI, F. MALLET, J. DEANTONI. *MARTE for modeling and verification of real-time embedded systems*, in "1er Colloque International Relation Université Nice -Université Danang RUNSUD 2010", Nice-Sophia antipolis, France, March 2010, p. 480–489, http://www.i3s.unice.fr/~map/publications.FR.html.

[29] D. POTOP-BUTUCARU, A. AZIM, S. FISCHMEISTER. *Semantics-Preserving Implementation of Synchronous Specifications Over Dynamic TDMA Distributed Architectures*, in "Proceedings of the eighth ACM international conference on Embedded software, EMSOFT'10", Scottsdale, AZ, USA, Oct 2010, http://www-rocq.inria.fr/syndex/publications/pubs/emsoft10/emsoft10.pdf.

[30] H. YU, J.-P. TALPIN, L. BESNARD, T. GAUTIER, F. MALLET, C. ANDRÉ, R. DE SIMONE. *Polychronous Analysis of Timing Constraints in UML MARTE*, in "IEEE International Workshop on Model-Based Engineering for Real-Time Embedded Systems Design", Parador of Carmona, Spain, (RT-MOBES), ISORC, 2010, p. 145–151, http://hal.inria.fr/inria-00497249.

Scientific Books (or Scientific Book chapters)

[31] 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.

[32] 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.

[33] F. MALLET, C. ANDRÉ, R. DE SIMONE. *1*, in "IP-XACT components with Abstract Time Characterization", LNEE, Springer, July 2010, vol. 63, p. 3–18, http://dx.doi.org/10.1007/978-90-481-9303-5_1.

[34] 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.

Research Reports

[35] C. ANDRÉ. *Verification of clock constraints: CCSL Observers in Esterel*, INRIA, Feb 2010, RR-7211, http://hal.inria.fr/inria-00458847.

[36] A. COADOU, R. DE SIMONE. *Normal Forms and Equivalence of K-periodically Routed Graphs*, INRIA, May 2010, RR-7286, http://hal.inria.fr/inria-00485609.

[37] H. DUBOIS, M.-A. PERALDI-FRATI, F. LAKHAL. *A model for requirements traceability in an heterogeneous model-based design process. Application to automotive embedded systems*, INRIA, May 2010, RR-7292, http://hal.inria.fr/inria-00483970.

[38] R. GASCON, F. MALLET, J. DEANTONI. *Logical time and temporal logics: Comparing UML MARTE/CCSL and PSL*, INRIA, 2010, n^{o} 7459, http://hal.archives-ouvertes.fr/inria-00540738/.

[39] F. MALLET, C. ANDRÉ, F. LAGARDE. *Un profil UML pour la modélisation multiniveau*, INRIA, May 2010, RR-7287, http://hal.inria.fr/inria-00482727.

References in notes

[40] 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.

[41] 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.

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

[43] 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.

[44] 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.

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

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

[47] S. EDWARDS, EDWARD A. LEE. *The case for the precision timed (PRET) machine*, in "Proceedings of the 44th annual Design Automation Conference", New York, NY, USA, DAC '07, ACM, 2007, p. 264–265, http://doi.acm.org/10.1145/1278480.1278545.

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

[49] 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.

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

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