Activity Report 2015

Team HYCOMES

Modélisation hybride & conception par contrats pour les systèmes embarqués multi-physiques

Inria teams are typically groups of researchers working on the definition of a common project, and objectives, with the goal to arrive at the creation of a project-team. Such project-teams may include other partners (universities or research institutions).

RESEARCH CENTER
Rennes - Bretagne-Atlantique

THEME
Embedded and Real-time Systems

Table of contents

Team HYCOMES

Creation of the Team: 2013 July 01

Keywords:

Computer Science and Digital Science:

2. - Software
2.1. - Programming Languages
2.1.1. - Semantics of programming languages
2.1.10. - Domain-specific languages
2.1.5. - Constraint programming
2.1.8. - Synchronous languages
2.2. - Compilation
2.3. - Embedded and cyber-physical systems
2.3.2. - Cyber-physical systems
2.4. - Reliability, certification
2.4.1. - Analysis
2.4.2. - Verification
2.5. - Software engineering
6. - Modeling, simulation and control
6.1. - Mathematical Modeling
6.1.1. - Continuous Modeling (PDE, ODE)
6.1.5. - Multiphysics modeling

Other Research Topics and Application Domains:

4. - Energy
5. - Industry of the future
5.2. - Design and manufacturing
5.2.1. - Road vehicles
5.2.2. - Railway
5.2.3. - Aviation
5.2.4. - Aerospace
5.9. - Industrial maintenance
7. - Transport and logistics
7.1. - Traffic management
7.1.3. - Air traffic
8.1. - Smart building/home
8.1.1. - Energy for smart buildings

1. Members

Research Scientists

Benoît Caillaud [Team leader, Inria, Senior Researcher, HdR]
Albert Benveniste [Inria, Senior Researcher, HdR]

Khalil Ghorbal [Inria, Researcher, from Oct 2015]

PhD Student

Ayman Aljarbouh [Inria, granted by Conseil Régional de Bretagne]

Administrative Assistant

Angélique Jarnoux [Inria]

Other

Valentin Leblanc [Inria,Internship, from Feb 2015 until Mar 2015]

2. Overall Objectives

2.1. Overall Objectives

Hycomes has been created as a new team of the Rennes — Bretagne Atlantique Inria research center in July 2013. The team builds upon the most promising results of the former S4 team-project and of the Synchronics large scale initiative. Two topics in embedded system design are covered:

- Hybrid systems modelling, with applications to the design of multi-physics embedded systems, often referenced as cyber-physical systems;
- Contract-based design and interface theories, with applications to requirements engineering in the context of safety-critical systems design.

3. Research Program

3.1. Hybrid Systems Modeling

Systems industries today make extensive use of mathematical modeling tools to design computer controlled physical systems. This class of tools addresses the modeling of physical systems with models that are simpler than usual scientific computing problems by using only Ordinary Differential Equations (ODE) and Difference Equations but not Partial Differential Equations (PDE). This family of tools first emerged in the 1980's with SystemBuild by MatrixX (now distributed by National Instruments) followed soon by Simulink by Mathworks, with an impressive subsequent development.

In the early 90's control scientists from the University of Lund (Sweden) realized that the above approach did not support component based modeling of physical systems with reuse [1]. For instance, it was not easy to draw an electrical or hydraulic circuit by assembling component models of the various devices. The development of the Omola language by Hilding Elmqvist was a first attempt to bridge this gap by supporting some form of Differential Algebraic Equations (DAE) in the models. Modelica quickly emerged from this first attempt and became in the 2000's a major international concerted effort with the Modelica Consortium [2]. A wider set of tools, both industrial and academic, now exists in this segment [3]. In the EDA sector, VHDL-AMS was developed as a standard [18].

Despite these tools are now widely used by a number of engineers, they raise a number of technical difficulties. The meaning of some programs, their mathematical semantics, can be tainted with uncertainty. A main source of difficulty lies in the failure to properly handle the discrete and the continuous parts of systems, and their interaction. How the propagation of mode changes and resets should be handled? How to avoid artifacts due to the use of a global ODE solver causing unwanted coupling between seemingly non interacting subsystems? Also, the mixed use of an equational style for the continuous dynamics with an imperative style for the mode changes and resets is a source of difficulty when handling parallel composition. It is therefore not uncommon that tools return complex warnings for programs with many different suggested hints for fixing them. Yet, these "pathological" programs can still be executed, if wanted so, giving surprising results — See for instance the Simulink examples in [24], [3] and [21].

[1] http://www.lccc.lth.se/media/LCCC2012/WorkshopSeptember/slides/Astrom.pdf
[2] https://www.modelica.org/
[3] SimScape by Mathworks, Amesim by LMS International, now Siemens PLM, and more.

Indeed this area suffers from the same difficulties that led to the development of the theory of synchronous languages as an effort to fix obscure compilation schemes for discrete time equation based languages in the 1980's. Our vision is that hybrid systems modeling tools deserve similar efforts in theory as synchronous languages did for the programming of embedded systems.

3.2. Background on non-standard analysis

Non-Standard analysis plays a central role in our research on hybrid systems modeling [3], [24], [22], [21]. The following text provides a brief summary of this theory and gives some hints on its usefulness in the context of hybrid systems modeling. This presentation is based on our paper [3], a chapter of Simon Bliudze's PhD thesis [29], and a recent presentation of non-standard analysis, not axiomatic in style, due to the mathematician Lindström [49].

Non-standard numbers allowed us to reconsider the semantics of hybrid systems and propose a radical alternative to the *super-dense time semantics* developed by Edward Lee and his team as part of the Ptolemy II project, where cascades of successive instants can occur in zero time by using $\mathbb{R}_+ \times \mathbb{N}$ as a time index. In the non-standard semantics, the time index is defined as a set $\mathbb{T} = \{n\partial \mid n \in {}^*\mathbb{N}\}$, where ∂ is an *infinitesimal* and ${}^*\mathbb{N}$ is the set of *non-standard integers*. Remark that 1/ $\mathbb{T}$ is dense in $\mathbb{R}_+$, making it "continuous", and 2/ every $t \in \mathbb{T}$ has a predecessor in $\mathbb{T}$ and a successor in $\mathbb{T}$, making it "discrete". Although it is not effective from a computability point of view, the *non-standard semantics* provides a framework that is familiar to the computer scientist and at the same time efficient as a symbolic abstraction. This makes it an excellent candidate for the development of provably correct compilation schemes and type systems for hybrid systems modeling languages.

Non-standard analysis was proposed by Abraham Robinson in the 1960s to allow the explicit manipulation of "infinitesimals" in analysis [55], [43], [17]. Robinson's approach is axiomatic; he proposes adding three new axioms to the basic Zermelo-Fraenkel (ZFC) framework. There has been much debate in the mathematical community as to whether it is worth considering non-standard analysis instead of staying with the traditional one. We do not enter this debate. The important thing for us is that non-standard analysis allows the use of the non-standard discretization of continuous dynamics "as if" it was operational.

Not surprisingly, such an idea is quite ancient. Iwasaki et al. [45] first proposed using non-standard analysis to discuss the nature of time in hybrid systems. Bliudze and Krob [30], [29] have also used non-standard analysis as a mathematical support for defining a system theory for hybrid systems. They discuss in detail the notion of "system" and investigate computability issues. The formalization they propose closely follows that of Turing machines, with a memory tape and a control mechanism.

The introduction to non-standard analysis in [29] is very pleasant and we take the liberty to borrow it. This presentation was originally due to Lindstrøm, see [49]. Its interest is that it does not require any fancy axiomatic material but only makes use of the axiom of choice — actually a weaker form of it. The proposed construction bears some resemblance to the construction of $\mathbb{R}$ as the set of equivalence classes of Cauchy sequences in $\mathbb{Q}$ modulo the equivalence relation $(u_n) \approx (v_n)$ iff $\lim_{n\to\infty}(u_n - v_n) = 0$.

3.2.1. Motivation and intuitive introduction

We begin with an intuitive introduction to the construction of the non-standard reals. The goal is to augment $\mathbb{R} \cup \{\pm\infty\}$ by adding, to each x in the set, a set of elements that are "infinitesimally close" to it. We will call the resulting set ${}^*\mathbb{R}$. Another requirement is that all operations and relations defined on $\mathbb{R}$ should extend to ${}^*\mathbb{R}$.

A first idea is to represent such additional numbers as convergent sequences of reals. For example, elements infinitesimally close to the real number zero are the sequences $u_n = 1/n$, $v_n = 1/\sqrt{n}$ and $w_n = 1/n^2$. Observe that the above three sequences can be ordered: $v_n > u_n > w_n > 0$ where 0 denotes the constant zero sequence. Of course, infinitely large elements (close to $+\infty$) can also be considered, e.g., sequences $x_u = n$, $y_n = \sqrt{n}$, and $z_n = n^2$.

Unfortunately, this way of defining ${}^*\mathbb{R}$ does not yield a total order since two sequences converging to zero cannot always be compared: if u_n and u'_n are two such sequences, the three sets $\{n \mid u_n > u'_n\}$, $\{n \mid u_n = u'_n\}$, and $\{n \mid u_n < u'_n\}$ may even all be infinitely large. The beautiful idea of Lindstrøm is to enforce that *exactly one of the above sets is important and the other two can be neglected*. This is achieved by fixing once and for all a finitely additive positive measure μ over the set $\mathbb{N}$ of integers with the following properties: [4]

1. $\mu : 2^{\mathbb{N}} \to \{0, 1\}$;
2. $\mu(X) = 0$ whenever X is finite;
3. $\mu(\mathbb{N}) = 1$.

Now, once μ is fixed, one can compare any two sequences: for the above case, exactly one of the three sets must have μ-measure 1 and the others must have μ-measure 0. Thus, say that $u > u', u = u'$, or $u < u'$, if $\mu(\{n \mid u_n > u'_n\} = 1)$, $\mu(\{n \mid u_n = u'_n\}) = 1$, or $\mu(\{n \mid u_n < u'_n\}) = 1$, respectively. Indeed, the same trick works for many other relations and operations on non-standard real numbers, as we shall see. We now proceed with a more formal presentation.

3.2.2. Construction of non-standard domains

For I an arbitrary set, a *filter* $\mathcal{F}$ over I is a family of subsets of I such that:

1. the empty set does not belong to $\mathcal{F}$,
2. $P, Q \in \mathcal{F}$ implies $P \cap Q \in \mathcal{F}$, and
3. $P \in \mathcal{F}$ and $P \subset Q \subseteq I$ implies $Q \in \mathcal{F}$.

Consequently, $\mathcal{F}$ cannot contain both a set P and its complement P^c. A filter that contains one of the two for any subset $P \subseteq I$ is called an *ultra-filter*. At this point we recall Zorn's lemma, known to be equivalent to the axiom of choice:

Lemma 1 (Zorn's lemma) *Any partially ordered set $(X, \leq)$ such that any chain in X possesses an upper bound has a maximal element.*

A filter $\mathcal{F}$ over I is an ultra-filter if and only if it is maximal with respect to set inclusion. By Zorn's lemma, any filter $\mathcal{F}$ over I can be extended to an ultra-filter over I. Now, if I is infinite, the family of sets $\mathcal{F} = \{P \subseteq I \mid P^c \text{ is finite}\}$ is a *free* filter, meaning it contains no finite set. It can thus be extended to a free ultra-filter over I:

Lemma 2 Any infinite set has a free ultra-filter.

Every free ultra-filter $\mathcal{F}$ over I uniquely defines, by setting $\mu(P) = 1$ if $P \in \mathcal{F}$ and otherwise 0, a finitely additive measure [5] $\mu : 2^I \mapsto \{0, 1\}$, which satisfies

$$\mu(I) = 1 \text{ and, if } P \text{ is finite, then } \mu(P) = 0.$$

Now, fix an infinite set I and a finitely additive measure μ over I as above. Let $\mathbb{X}$ be a set and consider the Cartesian product $\mathbb{X}^I = (x_i)_{i \in I}$. Define $(x_i) \approx (x'_i)$ iff $\mu\{i \in I \mid x_i \neq x'_i\} = 0$. Relation $\approx$ is an equivalence relation whose equivalence classes are denoted by $[x_i]$ and we define:

$${}^*\mathbb{X} = \mathbb{X}^I / \approx \tag{1}$$

[4] The existence of such a measure is non trivial and is explained later.

[5] Observe that, as a consequence, μ cannot be sigma-additive (in contrast to probability measures or Radon measures) in that it is *not* true that $\mu(\bigcup_n A_n) = \sum_n \mu(A_n)$ holds for an infinite denumerable sequence A_n of pairwise disjoint subsets of $\mathbb{N}$.

$\mathbb{X}$ is naturally embedded into ${}^*\mathbb{X}$ by mapping every $x \in \mathbb{X}$ to the constant tuple such that $x_i = x$ for every $i \in I$. Any algebraic structure over $\mathbb{X}$ (group, ring, field) carries over to ${}^*\mathbb{X}$ by almost point-wise extension. In particular, if $[x_i] \neq 0$, meaning that $\mu\{i \mid x_i = 0\} = 0$ we can define its inverse $[x_i]^{-1}$ by taking $y_i = x_i^{-1}$ if $x_i \neq 0$ and $y_i = 0$ otherwise. This construction yields $\mu\{i \mid y_i x_i = 1\} = 1$, whence $[y_i][x_i] = 1$ in ${}^*\mathbb{X}$. The existence of an inverse for any non-zero element of a ring is indeed stated by the formula: $\forall x\,(x = 0 \vee \exists y\,(xy = 1))$. More generally:

Lemma 3 (Transfer Principle) Every first order formula is true over ${}^*\mathbb{X}$ iff it is true over $\mathbb{X}$.

The above general construction can simply be applied to $\mathbb{X} = \mathbb{R}$ and $I = \mathbb{N}$. The result is denoted ${}^*\mathbb{R}$; it is a field according to the transfer principle. By the same principle, ${}^*\mathbb{R}$ is totally ordered by $[u_n] \leq [v_n]$ iff $\mu\{n \mid u_n > v_n\} = 0$. We claim that, for any finite $[x_n] \in {}^*\mathbb{R}$, there exists a unique $st([x_n])$, call it the *standard part* of $[x_n]$, such that

$$st([x_n]) \in \mathbb{R} \quad \text{and} \quad st([x_n]) \approx [x_n]\,. \tag{2}$$

To prove this, let $x = \sup\{u \in \mathbb{R} \mid [u] \leq [x_n]\}$, where $[u]$ denotes the constant sequence equal to u. Since $[x_n]$ is finite, x exists and we only need to show that $[x_n] - x$ is infinitesimal. If not, then there exists $y \in \mathbb{R}, y > 0$ such that $y < |x - [x_n]|$, that is, either $x < [x_n] - [y]$ or $x > [x_n] + [y]$, which both contradict the definition of x. The uniqueness of x is clear, thus we can define $st([x_n]) = x$. Infinite non-standard reals have no standard part in $\mathbb{R}$.

It is also of interest to apply the general construction (1) to $\mathbb{X} = I = \mathbb{N}$, which results in the set ${}^*\mathbb{N}$ of *non-standard natural numbers*. The non-standard set ${}^*\mathbb{N}$ differs from $\mathbb{N}$ by the addition of *infinite natural numbers*, which are equivalence classes of sequences of integers whose essential limit is $+\infty$.

3.3. Contract-Based Design, Interfaces Theories, and Requirements Engineering

System companies such as automotive and aeronautic companies are facing significant difficulties due to the exponentially raising complexity of their products coupled with increasingly tight demands on functionality, correctness, and time-to-market. The cost of being late to market or of imperfections in the products is staggering as witnessed by the recent recalls and delivery delays that many major car and airplane manufacturers had to bear in the recent years. The specific root causes of these design problems are complex and relate to a number of issues ranging from design processes and relationships with different departments of the same company and with suppliers, to incomplete requirement specification and testing.

We believe the most promising means to address the challenges in systems engineering is to employ structured and formal design methodologies that seamlessly and coherently combine the various viewpoints of the design space (behavior, space, time, energy, reliability, ...), that provide the appropriate abstractions to manage the inherent complexity, and that can provide correct-by-construction implementations. The following technology issues must be addressed when developing new approaches to the design of complex systems:

- The overall design flows for heterogeneous systems and the associated use of models across traditional boundaries are not well developed and understood. Relationships between different teams inside a same company, or between different stake-holders in the supplier chain, are not well supported by solid technical descriptions for the mutual obligations.
- System requirements capture and analysis is in large part a heuristic process, where the informal text and natural language-based techniques in use today are facing significant challenges. Formal requirements engineering is in its infancy: mathematical models, formal analysis techniques and links to system implementation must be developed.
- Dealing with variability, uncertainty, and life-cycle issues, such as extensibility of a product family, are not well-addressed using available systems engineering methodologies and tools.

The challenge is to address the entire process and not to consider only local solutions of methodology, tools, and models that ease part of the design.

Contract-based design has been proposed as a new approach to the system design problem that is rigorous and effective in dealing with the problems and challenges described before, and that, at the same time, does not require a radical change in the way industrial designers carry out their task as it cuts across design flows of different type. Indeed, contracts can be used almost everywhere and at nearly all stages of system design, from early requirements capture, to embedded computing infrastructure and detailed design involving circuits and other hardware. Contracts explicitly handle pairs of properties, respectively representing the assumptions on the environment and the guarantees of the system under these assumptions. Intuitively, a contract is a pair $C = (A, G)$ of assumptions and guarantees characterizing in a formal way 1) under which context the design is assumed to operate, and 2) what its obligations are. Assume/Guarantee reasoning has been known for a long time, and has been used mostly as verification mean for the design of software [53]. However, contract based design with explicit assumptions is a philosophy that should be followed all along the design, with all kinds of models, whenever necessary. Here, specifications are not limited to profiles, types, or taxonomy of data, but also describe the functions, performances of various kinds (time and energy), and reliability. This amounts to enrich a component's interface with, on one hand, formal specifications of the behavior of the environment in which the component may be instantiated and, on the other hand, of the expected behavior of the component itself. The consideration of rich interfaces is still in its infancy. So far, academic researchers have addressed the mathematics and algorithmics of interfaces theories and contract-based reasoning. To make them a technique of choice for system engineers, we must develop:

- Mathematical foundations for interfaces and requirements engineering that enable the design of frameworks and tools;
- A system engineering framework and associated methodologies and tool sets that focus on system requirements modeling, contract specification, and verification at multiple abstraction layers.

A detailed bibliography on contract and interface theories for embedded system design can be found in [4]. In a nutshell, contract and interface theories fall into two main categories:

Assume/guarantee contracts. By explicitly relying on the notions of assumptions and guarantees, A/G-contracts are intuitive, which makes them appealing for the engineer. In A/G-contracts, assumptions and guarantees are just properties regarding the behavior of a component and of its environment. The typical case is when these properties are formal languages or sets of traces, which includes the class of safety properties [46], [37], [52], [20], [38]. Contract theories were initially developed as specification formalisms able to refuse some inputs from the environment [44]. A/G-contracts were advocated by the SPEEDS project [23]. They were further experimented in the framework of the CESAR project [39], with the additional consideration of *weak* and *strong* assumptions. This is still a very active research topic, with several recent contributions dealing with the timed [28] and probabilistic [33], [34] viewpoints in system design, and even mixed-analog circuit design [54].

Automata theoretic interfaces. Interfaces combine assumptions and guarantees in a single, automata theoretic specification. Most interface theories are based on Lynch Input/Output Automata [51], [50]. Interface Automata [58], [57], [59], [35] focus primarily on parallel composition and compatibility: Two interfaces can be composed and are compatible if there is at least one environment where they can work together. The idea is that the resulting composition exposes as an interface the needed information to ensure that incompatible pairs of states cannot be reached. This can be achieved by using the possibility, for an Interface Automaton, to refuse selected inputs from the environment in a given state, which amounts to the implicit assumption that the environment will never produce any of the refused inputs, when the interface is in this state. Modal Interfaces [5] inherit from both Interface Automata and the originally unrelated notion of Modal Transition System [48], [19], [31], [47]. Modal Interfaces are strictly more expressive than Interface Automata by decoupling the I/O orientation of an event and its deontic modalities (mandatory, allowed or forbidden). Informally, a *must* transition is available in every component that realizes the modal interface, while a *may* transition needs not be. Research on interface theories is still very active. For instance, timed [60], [25],

[27], [41], [40], [26], probabilistic [33], [42] and energy-aware [36] interface theories have been proposed recently.

Requirements Engineering is one of the major concerns in large systems industries today, particularly so in sectors where certification prevails [56]. DOORS projects collecting requirements are poorly structured and cannot be considered a formal modeling framework today. They are nothing more than an informal documentation enriched with hyperlinks. As examples, medium size sub-systems may have a few thousands requirements and the Rafale fighter aircraft has above 250,000 of them. For the Boeing 787, requirements were not stable while subcontractors performed the development of the fly-by-wire and of the landing gear subsystems.

We see Contract-Based Design and Interfaces Theories as innovative tools in support of Requirements Engineering. The Software Engineering community has extensively covered several aspects of Requirements Engineering, in particular:

- the development and use of large and rich *ontologies*; and
- the use of Model Driven Engineering technology for the structural aspects of requirements and resulting hyperlinks (to tests, documentation, PLM, architecture, and so on).

Behavioral models and properties, however, are not properly encompassed by the above approaches. This is the cause of a remaining gap between this phase of systems design and later phases where formal model based methods involving behavior have become prevalent—see the success of Matlab/Simulink/Scade technologies. We believe that our work on contract based design and interface theories is best suited to bridge this gap.

4. Highlights of the Year

4.1. Highlights of the Year

The main progress on hybrid systems modeling can be summarized as follows:

- As part of his PhD work, Ayman Aljarbouh has designed and implemented regularization techniques for hybrid systems with chattering behaviour [9]. His techniques enable the efficient simulation of chattering behavior that can not be simulated with pure *event-driven* simulation techniques.
- A constructive semantics for guarded DAE systems has been proposed. Guarded DAE systems are equivalent to the kernel language used as an intermediate format by several Modelica compilers. This semantics, based on a nonstandard (infinitesimal) time model [3], allows to determine the structural differentiation index and infer the causal dependencies of a system of guarded DAEs. The semantics has been implemented in SUNDAE, a prototype software, developed in the context of the Sys2soft (7.2) and Modrio projects (7.3.1).

5. New Software and Platforms

5.1. Flipflop

Test & Flip Net Synthesis Tool for the Inference of Technical Procedure Models
FUNCTIONAL DESCRIPTION

Flipflop is a Test and Flip net synthesis tool implementing a linear algebraic polynomial time algorithm. Computations are done in the Z/2Z ring. Test and Flip nets extend Elementary Net Systems by allowing test to zero, test to one and flip arcs. The effect of flip arcs is to complement the marking of the place. While the net synthesis problem has been proved to be NP hard for Elementary Net Systems, thanks to flip arcs, the synthesis of Test and Flip nets can be done in polynomial time. Test and flip nets have the required expressivity to give concise and accurate representations of surgical processes (models of types of surgical operations). Test and Flip nets can express causality and conflict relations. The tool takes as input either standard XES log files (a standard XML file format for process mining tools) or a specific XML file format for surgical applications. The output is a Test and Flip net, solution of the following synthesis problem: Given a finite input language (log file), compute a net, which language is the least language in the class of Test and Flip net languages, containing the input language.

- Contact: Benoît Caillaud
- URL: http://tinyurl.com/oql6f3y

5.2. MICA

Model Interface Compositional Analysis Library
KEYWORDS: Modal interfaces - Contract-based desing
SCIENTIFIC DESCRIPTION

In Mica, systems and interfaces are represented by extension. However, a careful design of the state and event heap enables the definition, composition and analysis of reasonably large systems and interfaces. The heap stores states and events in a hash table and ensures structural equality (there is no duplication). Therefore complex data-structures for states and events induce a very low overhead, as checking equality is done in constant time.

Thanks to the Inter module and the mica interactive environment, users can define complex systems and interfaces using Ocaml syntax. It is even possible to define parameterized components as Ocaml functions.
FUNCTIONAL DESCRIPTION

Mica is an Ocaml library implementing the Modal Interface algebra. The purpose of Modal Interfaces is to provide a formal support to contract based design methods in the field of system engineering. Modal Interfaces enable compositional reasoning methods on I/O reactive systems.

- Participant: Benoît Caillaud
- Contact: Benoît Caillaud
- URL: http://www.irisa.fr/s4/tools/mica/

5.3. TnF-C++

FUNCTIONAL DESCRIPTION

TnF-C++ is a robust and portable re-implementation of Flipflop, developed in 2014 and integrated in the S3PM toolchain. Both software have been designed in the context of the S3PM project on surgical procedure modeling and simulation,

- Contact: Benoît Caillaud
- URL: https://bitbucket.org/cpenet/tnf_cpp

6. New Results

6.1. Embedded Systems Design

6.1.1. Loosely Time-Triggered Architectures: Improvements and Comparisons

Participant: Albert Benveniste.

Loosely Time-Triggered Architectures (LTTAs) are a proposal for constructing distributed embedded control systems. They build on the quasi-periodic architecture, where computing units execute 'almost periodically', by adding a thin layer of middleware that facilitates the implementation of synchronous applications. In [7], we have shown how the deployment of a synchronous application on a quasi-periodic architecture can be modeled using a synchronous formalism. Then we have detailed two protocols, Back-Pressure LTTA, reminiscent of elastic circuits, and Time-Based LTTA, based on waiting. Compared to previous work, we presented controller models that can be compiled for execution and a simplified version of the Time-Based protocol. We also compared the LTTA approach with architectures based on clock synchronization.

6.2. Hybrid Systems Modeling

Participants: Ayman Aljarbouh, Albert Benveniste, Benoît Caillaud, Khalil Ghorbal.

6.2.1. Robust Simulation for Hybrid Systems: Chattering Path Avoidance

The sliding mode approach is recognized as an efficient tool for treating the chattering behavior in hybrid systems. However, the amplitude of chattering, by its nature, is proportional to magnitude of discontinuous control. A possible scenario is that the solution trajectories may successively enter and exit as well as slide on switching mani-folds of different dimensions. Naturally, this arises in dynamical systems and control applications whenever there are multiple discontinuous control variables. The main contribution of [9] is to provide a robust computational framework for the most general way to extend a flow map on the intersection of p intersected $(n-1)$-dimensional switching manifolds in at least p dimensions. We explored a new formulation to which we can define unique solutions for such particular behavior in hybrid systems and investigate its efficient computation/simulation. An extended version of this work has been presented at the Baltic Young Scientists Conference [8].

6.2.2. A Hierarchy of Proof Rules for Checking Positive Invariance of Algebraic and Semi-Algebraic Sets

In [6], we studied sound proof rules for checking positive invariance of algebraic and semi-algebraic sets, that is, sets satisfying polynomial equalities and those satisfying finite boolean combinations of polynomial equalities and inequalities, under the flow of polynomial ordinary differential equations. Problems of this nature arise in formal verification of continuous and hybrid dynamical systems, where there is an increasing need for methods to expedite formal proofs. We study the trade-off between proof rule generality and practical performance and evaluate our theoretical observations on a set of benchmarks. The relationship between increased deductive power and running time performance of the proof rules is far from obvious; we discuss and illustrate certain classes of problems where this relationship is interesting.

6.2.3. A Formally Verified Hybrid System for Safe Advisories in the Next-Generation Airborne Collision Avoidance System

The Next-Generation Airborne Collision Avoidance System (ACAS X) is intended to be installed on all large aircraft to give advice to pilots and prevent mid-air collisions with other aircraft. It is currently being developed by the Federal Aviation Administration (FAA). In [16] we determined the geometric configurations under which the advice given by ACAS X is safe under a precise set of assumptions and formally verify these configurations using hybrid systems theorem proving techniques. We considered subsequent advisories and showed how to adapt our formal verification to take them into account. We examined the current version of the real ACAS X system and discussed some cases where our safety theorem conflicts with the actual advisory given by that version, demonstrating how formal, hybrid systems proving approaches are helping to ensure the safety of ACAS X. Our approach is general and could also be used to identify unsafe advice issued by other collision avoidance systems or confirm their safety.

6.2.4. Domain Globalization: Using Languages to Support Technical and Social Coordination

When a project is realized in a globalized environment, multiple stakeholders from different organizations work on the same system. Depending on the stakeholders and their organizations, various (possibly overlapping) concerns are raised in the development of the system. In this context a Domain Specific Language (DSL) supports the work of a group of stakeholders who are responsible for addressing a specific set of concerns. We contributed to a book chapter [11], identifying the open challenges arising from the coordination of globalized domain-specific languages. We identified two types of coordination: technical coordination and social coordination. After presenting an overview of the current state of the art, we discussed first the open challenges arising from the composition of multiple DSLs, and then the open challenges associated to the collaboration in a globalized environment.

6.3. Contracts for Systems Design

Participants: Albert Benveniste, Benoît Caillaud.

6.3.1. Contracts for Systems Design: Theory, Methodology and Application Cases

Aircrafts, trains, cars, plants, distributed telecommunication military or health care systems, and more, involve systems design as a critical step. Complexity has caused system design times and costs to go severely over budget so as to threaten the health of entire industrial sectors. Heuristic methods and standard practices do not seem to scale with complexity so that novel design methods and tools based on a strong theoretical foundation are sorely needed. Model-based design as well as other methodologies such as layered and compositional design have been used recently but a unified intellectual framework with a complete design flow supported by formal tools is still lacking. Recently an "orthogonal" approach has been proposed that can be applied to all methodologies introduced thus far to provide a rigorous scaffolding for verification, analysis and abstraction/refinement: contract-based design. Several results have been obtained in this domain but a unified treatment of the topic that can help in putting contract-based design in perspective is missing. We have published two research reports [13], [12], that intend to provide such treatment where contracts are precisely defined and characterized so that they can be used in design methodologies such as the ones mentioned above with no ambiguity. In addition, the first report [13] provides an important link between interface and contract theories to show similarities and correspondences. This report is complemented by a companion report [12] where contract based design is illustrated through use cases.

6.3.2. Contracts for Schedulability Analysis

In [10] we proposed a framework of Assume / Guarantee contracts for schedulability analysis. Unlike previous work addressing compositional scheduling analysis, our objective is to provide support for the OEM / supplier subcontracting relation. The adaptation of Assume / Guarantee contracts to schedulability analysis requires some care, due to the handling of conflicts caused by shared resources. We illustrate our framework in the context of Autosar methodology now popular in the automotive industry sector.

7. Partnerships and Cooperations

7.1. Regional Initiatives

- Ayman Aljarbouh's PhD is partially funded by an ARED grant of the Brittany Regional Council. His doctoral work takes place in the context of the Modrio and Sys2Soft projects on hybrid systems modeling — see sections 7.2 and 7.2. Ayman Aljarbouh is working on accelerated simulation techniques for hybrid systems. In particular, he is focusing on the regularisation, at runtime, of chattering behaviour and the approximation of Zeno behaviour.

- Benoît Caillaud is participating to the S3PM project of the CominLabs excellence laboratory [6]. This project focuses on the computation of surgical procedural knowledge models from recordings of individual procedures, and their execution [32]. The objective is to develop an enabling technology for procedural knowledge based computer assistance of surgery. In this project, we demonstrate its potential added value in nurse and surgeon training.

7.2. National Initiatives

Program:« Briques génériques du logiciel embarqué » (Embedded Software Generic Building-Blocks)

Project acronym: Sys2soft

Project title: Physics Aware Software

Duration: June 2012 – November 2015

Coordinator: Dassault Systèmes (France)

Other partners: Thales TGS / TRT / TAS, Alstom Transport, Airbus, DPS, Obeo, Soyatec

Abstract: The Sys2soft project aims at developping methods and tools supporting the design of embedded software interacting with a complex physical environment. The project advocates a methodology where both physics and software are co-modeled and co-simulated early in the design process and embedded code is generated automatically from the joint physics and software models. Extensions of the Modelica language with synchronous programming features are being investigated, as a unified framework where interacting physical and software artifacts can be modeled.

7.3. European Initiatives

7.3.1. Collaborations in European Programs, except FP7 & H2020

Program: ITEA2

Project acronym: Modrio

Project title: Model Driven Physical Systems Operation

Duration: September 2012 – May 2016

Coordinator: EDF (France)

Other partners: ABB (Sweden), Ampère Laboratory / CNRS (France), Bielefeld University (Germany), Dassault Systèmes (Sweden), Dassault Aviation (France), DLR (Germany), DPS (France), EADS (France), Equa Simulation (Sweden), IFP (France), ITI (Germany), Ilmenau University (Germany), Katholic University of Leuven (Belgium), Knorr-Bremse (Germany), LMS (France and Belgium), Linköping University (Sweden), MathCore (Sweden), Modelon (Sweden), Pöry (Finland), Qtronic (Germany), SICS (Sweden), Scania (Sweden), Semantum (Finland), Sherpa Engineering (France), Siemens (Germany and Sweden), Simpack (Germany), SKF (Sweden), Supmeca (France), Triphase (Belgium), University of Calabria (Italy), VTT (Finland), Vattenfall (Sweden), Wapice (Finland).

Abstract: Modelling and simulation are efficient and widely used tools for system design. But they are seldom used for systems operation. However, most functionalities for system design are beneficial for system operation, provided that they are enhanced to deal with real operating situations. Through open standards the benefits of sharing compatible information and data become obvious: improved cooperation between the design and the operation communities, easier adaptation of operation procedures wrt. design evolutions. Open standards also foster general purpose technology. The objective of the ITEA 2 MODRIO project is to extend modelling and simulation tools based on open standards from system design to system operation.

[6] http://www.s3pm.cominlabs.ueb.eu/

7.4. International Research Visitors

7.4.1. Research stays abroad

Ayman Aljarbouh has visited for two months Walid Taha's team (http://www.hh.se/english/research/professors/walidmohamedtaha.10235.html) at Halmstad university in Sweden. He has been working on the implementation in the Accumen language of the regularization techniques he is developing in his PhD work.

8. Dissemination

8.1. Promoting Scientific Activities

8.1.1. Scientific events organisation

8.1.1.1. Member of the organizing committees

Benoît Caillaud has served on the program and organizing committees of LMCS 2015 (http://www.acsysteme.com/fr/lmcs-2015), a national workshop on mathematical modeling and scientific computing.

8.1.2. Scientific events selection

8.1.2.1. Member of the conference program committees

Benoît Caillaud has served on the program committee of ACSD 2015 (http://www.ulb.ac.be/di/verif/pn2015acsd2015/), a conference on the applications of concurrency in system design. He is a member of the steering committe of ACSD since 2006.

Albert Benveniste has served on the program committee of MODELICA 2015 (https://www.modelica.org/events/modelica2015), the conference on the Modelica mathematical modeling language.

8.1.2.2. Reviewer

Benoît Caillaud has reviewed papers submitted to the EMSOFT 2015, TACAS 2015 and ACC 2016 conferences.

8.1.3. Journal

8.1.3.1. Reviewer - Reviewing activities

Benoît Caillaud has reviewed papers submitted to Acta Informatica.

8.1.4. Invited talks

Benoît Caillaud has given an invited talk on *Time Domains in Hybrid Systems Modeling* at the LCCC-ACCESS workshop on Model-Based Engineering (http://www.lccc.lth.se/index.php?page=LCCC-ACCESS-2015-05). He has given an invited talk on *Contracts and Interfaces in System Engineering* at the CNRIA 2015 national conference (http://cnria.cci.ucad.sn/), Thiès, Sénégal.

Albert Benveniste has given invited talks at the Modelica 2015 conference (https://www.modelica.org/events/modelica2015) and at the Gretsi 2015 colloquium (http://www.gretsi.fr/colloque2015/). He has given an invited talk at the LCCC-Access workshop on Model-Based Engineering (http://www.lccc.lth.se/index.php?page=LCCC-ACCESS-2015-05).

8.1.5. Research administration

Benoît Caillaud is head of the *Languages and Software Engineering* department of IRISA (http://www.irisa.fr/en/departments/d4-language-and-software-engineering). He has been in charge of editing the synthesis documents, regarding this department, used for the evaluation of IRISA by HCERES in January 2016.

8.2. Teaching - Supervision - Juries

8.2.1. Teaching

Master : Benoît Caillaud is teaching with Marc Pouzet a first year master degree course on *hybrid systems modeling*. The course is open to the students registered to the computer science research and innovation curriculum of the university of Rennes 1 and ENS Rennes.

8.2.2. Supervision

PhD in progress : Ayman Aljarbouh, *Accelerated Simulation of Hybrid Systems*, started january 2014, supervised by Benoît Caillaud

8.2.3. Juries

Albert Benveniste has participated to the HdR jury of Axel Legay (University of Rennes 1, november 2015).

8.3. Popularization

Albert Benveniste, as a member of the French Academy of Technology, has participated to an inquiry of the French Ministry of Education on the teaching of computer science in the secondary school curriculum.

9. Bibliography

Major publications by the team in recent years

[1] A. BENVENISTE, T. BOURKE, B. CAILLAUD, M. POUZET. *A hybrid synchronous language with hierarchical automata: static typing and translation to synchronous code*, in "Proceedings of the 11th International Conference on Embedded Software, EMSOFT", S. CHAKRABORTY, A. JERRAYA, S. K. BARUAH, S. FISCHMEISTER (editors), ACM, 2011, pp. 137-148, http://doi.acm.org/10.1145/2038642.2038664

[2] A. BENVENISTE, T. BOURKE, B. CAILLAUD, M. POUZET. *Divide and recycle: types and compilation for a hybrid synchronous language*, in "Proceedings of the ACM SIGPLAN/SIGBED Conference on Languages, Compilers, and Tools for Embedded Systems, LCTES Chicago, IL, USA, April 11-14", J. VITEK, B. DE SUTTER (editors), ACM, 2011, pp. 61-70, http://doi.acm.org/10.1145/1967677.1967687

[3] A. BENVENISTE, T. BOURKE, B. CAILLAUD, M. POUZET. *Non-standard semantics of hybrid systems modelers*, in "Journal of Computer and System Sciences", 2012, vol. 78, n^{o} 3, pp. 877-910, This work was supported by the SYNCHRONICS large scale initiative of Inria [*DOI :* 10.1016/J.JCSS.2011.08.009], http://hal.inria.fr/hal-00766726

[4] A. BENVENISTE, B. CAILLAUD, D. NICKOVIC, R. PASSERONE, J.-B. RACLET, P. REINKEMEIER, A. L. SANGIOVANNI-VINCENTELLI, W. DAMM, T. A. HENZINGER, K. G. LARSEN. *Contracts for System Design*, Inria, November 2012, n^{o} RR-8147, 65 p. , http://hal.inria.fr/hal-00757488

[5] J.-B. RACLET, E. BADOUEL, A. BENVENISTE, B. CAILLAUD, A. LEGAY, R. PASSERONE. *A Modal Interface Theory for Component-based Design*, in "Fundamenta Informaticae", 2011, vol. 108, n^{o} 1-2, pp. 119-149, http://dx.doi.org/10.3233/FI-2011-416

Publications of the year

Articles in International Peer-Reviewed Journals

[6] K. GHORBAL, A. SOGOKON, A. PLATZER. *A Hierarchy of Proof Rules for Checking Positive Invariance of Algebraic and Semi-Algebraic Sets*, in "Computer Languages, Systems and Structures", November 2015, https://hal.archives-ouvertes.fr/hal-01232288

International Conferences with Proceedings

[7] G. BAUDART, T. BOURKE, A. BENVENISTE. *Loosely Time-Triggered Architectures: Improvements and Comparisons*, in "Proceedings of the 12th International Conference on Embedded Software (EMSOFT '15)", Amsterdam, Netherlands, October 2015 [*DOI :* 10.1109/EMSOFT.2015.7318263], https://hal.inria.fr/hal-01243005

Conferences without Proceedings

[8] A. ALJARBOUH, B. CAILLAUD. *On the Regularization of Chattering Executions in Real Time Simulation of Hybrid Systems*, in "Baltic Young Scientists Conference", Tallinn, Estonia, C. CAP (editor), Universität Rostock, July 2015, 49 p. , https://hal.archives-ouvertes.fr/hal-01246853

[9] A. ALJARBOUH, B. CAILLAUD. *Robust Simulation for Hybrid Systems Chattering Path Avoidance*, in "The 56th Conference on Simulation and Modelling (SIMS 56)", Linköping, Sweden, Linköping University Press, October 2015, vol. 119, n^{o} 018, pp. 175-185 [*DOI :* 10.3384/ECP15119175], https://hal.archives-ouvertes.fr/hal-01247074

[10] P. REINKEMEIER, A. BENVENISTE, W. DAMM, I. STIERAND. *Contracts for Schedulability Analysis*, in "13th International Conference on Formal Modeling and Analysis of Timed Systems, FORMATS 2015", Madrid, Spain, Sriram Sankaranarayanan (University of Colorado at Boulder, USA) and Enrico Vicario (University of Florence, Italy), September 2015, https://hal.inria.fr/hal-01182407

Scientific Books (or Scientific Book chapters)

[11] J. DEANTONI, C. BRUN, B. CAILLAUD, R. FRANCE, G. KARSAI, O. NIERSTRASZ, E. SYRIANI. *Domain Globalization: Using Languages to Support Technical and Social Coordination*, in "Globalizing Domain-Specific Languages", B. COMBEMALE, B. H. CHENG, R. B. FRANCE, J.-M. JÉZÉQUEL, B. RUMPE (editors), Lecture Notes in Computer Science, Springer International Publishing, 2015, vol. 9400, pp. 70-87 [*DOI :* 10.1007/978-3-319-26172-0_5], https://hal.archives-ouvertes.fr/hal-01234653

Research Reports

[12] A. BENVENISTE, B. CAILLAUD, D. NICKOVIC, R. PASSERONE, J.-B. RACLET, P. REINKEMEIER, A. SANGIOVANNI-VINCENTELLI, W. DAMM, T. HENZINGER, K. LARSEN. *Contracts for Systems Design: Methodology and Application cases*, Inria Rennes Bretagne Atlantique ; Inria, July 2015, n^{o} RR-8760, 63 p. , https://hal.inria.fr/hal-01178469

[13] A. BENVENISTE, B. CAILLAUD, D. NICKOVIC, R. PASSERONE, J.-B. RACLET, P. REINKEMEIER, A. SANGIOVANNI-VINCENTELLI, W. DAMM, T. HENZINGER, K. LARSEN. *Contracts for Systems Design: Theory*, Inria Rennes Bretagne Atlantique ; Inria, July 2015, n^{o} RR-8759, 86 p. , https://hal.inria.fr/hal-01178467

Other Publications

[14] A. ALJARBOUH, B. CAILLAUD. *Chattering-Free Simulation of Hybrid Dynamical Systems with the Function Mock-Up Interface 2.0*, February 2016, working paper or preprint, https://hal.archives-ouvertes.fr/hal-01247008

[15] A. ALJARBOUH, Y. ZENG, A. DURACZ, W. TAHA, B. CAILLAUD. *Chattering-Free Simulation of Non-Smooth Hybrid Systems*, January 2016, working paper or preprint, https://hal.archives-ouvertes.fr/hal-01247015

[16] J.-B. JEANNIN, K. GHORBAL, Y. KOUSKOULAS, A. SCHMIDT, R. GARDNER, S. MITSCH, A. PLATZER. *A Formally Verified Hybrid System for Safe Advisories in the Next-Generation Airborne Collision Avoidance System*, November 2015, working paper or preprint, https://hal.archives-ouvertes.fr/hal-01232365

References in notes

[17] N. J. CUTLAND (editor). *Nonstandard analysis and its applications*, Cambridge Univ. Press, 1988

[18] *IEEE Standard VHDL Analog and Mixed-Signal Extensions, Std 1076.1-1999*, 1999, http://dx.doi.org/10.1109/IEEESTD.1999.90578

[19] A. ANTONIK, M. HUTH, K. G. LARSEN, U. NYMAN, A. WASOWSKI. *20 Years of Modal and Mixed Specifications*, in "Bulletin of European Association of Theoretical Computer Science", 2008, vol. 1, n^{o} 94

[20] C. BAIER, J.-P. KATOEN. *Principles of Model Checking*, MIT Press, Cambridge, 2008

[21] A. BENVENISTE, T. BOURKE, B. CAILLAUD, B. PAGANO, M. POUZET. *A Type-Based Analysis of Causality Loops In Hybrid Systems Modelers*, December 2013, Deliverable D3.1_1 v 1.0 of the Sys2soft collaborative project ”Physics Aware Software”, https://hal.inria.fr/hal-00938866

[22] A. BENVENISTE, T. BOURKE, B. CAILLAUD, M. POUZET. *Semantics of multi-mode DAE systems*, August 2013, Deliverable D.4.1.1 of the ITEA2 Modrio collaborative project, https://hal.inria.fr/hal-00938891

[23] A. BENVENISTE, B. CAILLAUD, A. FERRARI, L. MANGERUCA, R. PASSERONE, C. SOFRONIS. *Multiple Viewpoint Contract-Based Specification and Design*, in "Proceedings of the Software Technology Concertation on Formal Methods for Components and Objects (FMCO'07)", Amsterdam, The Netherlands, Revised Lectures, Lecture Notes in Computer Science, Springer, October 2008, vol. 5382

[24] A. BENVENISTE, B. CAILLAUD, B. PAGANO, M. POUZET. *A type-based analysis of causality loops in hybrid modelers*, in "HSCC '14: International Conference on Hybrid Systems: Computation and Control", Berlin, Germany, Proceedings of the 17th international conference on Hybrid systems: computation and control (HSCC '14), ACM Press, April 2014, 13 p. [*DOI :* 10.1145/2562059.2562125], https://hal.inria.fr/hal-01093388

[25] N. BERTRAND, A. LEGAY, S. PINCHINAT, J.-B. RACLET. *A Compositional Approach on Modal Specifications for Timed Systems*, in "11th International Conference on Formal Engineering Methods (ICFEM'09)", Rio de Janeiro, Brazil, LNCS, Springer, December 2009, vol. 5885, pp. 679-697, http://hal.inria.fr/inria-00424356/en

[26] N. BERTRAND, A. LEGAY, S. PINCHINAT, J.-B. RACLET. *Modal event-clock specifications for timed component-based design*, in "Science of Computer Programming", 2011, http://dx.doi.org/10.1016/j.scico.2011.01.007

[27] N. BERTRAND, S. PINCHINAT, J.-B. RACLET. *Refinement and Consistency of Timed Modal Specifications*, in "3rd International Conference on Language and Automata Theory and Applications (LATA'09)", Tarragona, Spain, LNCS, Springer, April 2009, vol. 5457, pp. 152-163 [*DOI :* 10.1007/978-3-642-00982-2_13], http://hal.inria.fr/inria-00424283/en

[28] P. BHADURI, I. STIERAND. *A proposal for real-time interfaces in SPEEDS*, in "Design, Automation and Test in Europe (DATE'10)", IEEE, 2010, pp. 441-446

[29] S. BLIUDZE. *Un cadre formel pour l'étude des systèmes industriels complexes: un exemple basé sur l'infrastructure de l'UMTS*, Ecole Polytechnique, 2006

[30] S. BLIUDZE, D. KROB. *Modelling of Complex Systems: Systems as Dataflow Machines*, in "Fundam. Inform.", 2009, vol. 91, n^{o} 2, pp. 251–274

[31] G. BOUDOL, K. G. LARSEN. *Graphical Versus Logical Specifications*, in "Theor. Comput. Sci.", 1992, vol. 106, n^{o} 1, pp. 3-20

[32] B. CAILLAUD. *Surgical Process Mining with Test and Flip Net Synthesis*, in "Application of Region Theory (ART)", Barcelona, Spain, R. BERGENTHUM, J. CARMONA (editors), July 2013, pp. 43-54, http://hal.inria.fr/hal-00872284

[33] B. CAILLAUD, B. DELAHAYE, K. G. LARSEN, A. LEGAY, M. L. PEDERSEN, A. WASOWSKI. *Compositional design methodology with constraint Markov chains*, in "QEST 2010", Williamsburg, Virginia, United States, September 2010 [*DOI :* 10.1109/QEST.2010.23], http://hal.inria.fr/inria-00591578/en

[34] B. CAILLAUD, B. DELAHAYE, K. G. LARSEN, A. LEGAY, M. L. PEDERSEN, A. WASOWSKI. *Constraint Markov Chains*, in "Theoretical Computer Science", May 2011, vol. 412, n^{o} 34, pp. 4373-4404 [*DOI :* 10.1016/J.TCS.2011.05.010], http://hal.inria.fr/hal-00654003/en

[35] A. CHAKRABARTI. *A Framework for Compositional Design and Analysis of Systems*, EECS Department, University of California, Berkeley, Dec 2007, http://www.eecs.berkeley.edu/Pubs/TechRpts/2007/EECS-2007-174.html

[36] A. CHAKRABARTI, L. DE ALFARO, T. A. HENZINGER, M. STOELINGA. *Resource Interfaces*, in "EMSOFT", R. ALUR, I. LEE (editors), Lecture Notes in Computer Science, Springer, 2003, vol. 2855, pp. 117-133

[37] E. Y. CHANG, Z. MANNA, A. PNUELI. *Characterization of Temporal Property Classes*, in "ICALP", W. KUICH (editor), Lecture Notes in Computer Science, Springer, 1992, vol. 623, pp. 474-486

[38] E. CLARKE, O. GRUMBERG, D. PELED. *Model Checking*, MIT Press, 1999

[39] W. DAMM, E. THADEN, I. STIERAND, T. PEIKENKAMP, H. HUNGAR. *Using Contract-Based Component Specifications for Virtual Integration and Architecture Design*, in "Proceedings of the 2011 Design, Automation and Test in Europe (DATE'11)", March 2011

[40] A. DAVID, K. G. LARSEN, A. LEGAY, U. NYMAN, A. WASOWSKI. *ECDAR: An Environment for Compositional Design and Analysis of Real Time Systems*, in "Automated Technology for Verification and Analysis - 8th International Symposium, ATVA 2010, Singapore, September 21-24, 2010. Proceedings", 2010, pp. 365-370

[41] A. DAVID, K. G. LARSEN, A. LEGAY, U. NYMAN, A. WASOWSKI. *Timed I/O automata: a complete specification theory for real-time systems*, in "Proceedings of the 13th ACM International Conference on Hybrid Systems: Computation and Control, HSCC 2010, Stockholm, Sweden, April 12-15, 2010", 2010, pp. 91-100

[42] B. DELAHAYE, J.-P. KATOEN, K. G. LARSEN, A. LEGAY, M. L. PEDERSEN, F. SHER, A. WASOWSKI. *Abstract Probabilistic Automata*, in "VMCAI", R. JHALA, D. A. SCHMIDT (editors), Lecture Notes in Computer Science, Springer, 2011, vol. 6538, pp. 324-339

[43] F. DIENER, G. REEB. *Analyse non standard*, Hermann, 1989

[44] D. L. DILL. *Trace Theory for Automatic Hierarchical Verification of Speed-Independent Circuits*, ACM Distinguished Dissertations, MIT Press, 1989

[45] Y. IWASAKI, A. FARQUHAR, V. SARASWAT, D. BOBROW, V. GUPTA. *Modeling Time in Hybrid Systems: How Fast Is "Instantaneous"?*, in "IJCAI", 1995, pp. 1773–1781

[46] L. LAMPORT. *Proving the Correctness of Multiprocess Programs*, in "IEEE Trans. Software Eng.", 1977, vol. 3, n^{O} 2, pp. 125-143

[47] K. G. LARSEN, U. NYMAN, A. WASOWSKI. *On Modal Refinement and Consistency*, in "Proc. of the 18th International Conference on Concurrency Theory (CONCUR'07)", Springer, 2007, pp. 105–119

[48] K. G. LARSEN, B. THOMSEN. *A Modal Process Logic*, in "Proceedings of the Third Annual Symposium on Logic in Computer Science (LICS'88)", IEEE, 1988, pp. 203-210

[49] T. LINDSTRØM. *An Invitation to Nonstandard Analysis*, in "Nonstandard Analysis and its Applications", N. J. CUTLAND (editor), Cambridge Univ. Press, 1988, pp. 1–105

[50] N. A. LYNCH. *Input/Output Automata: Basic, Timed, Hybrid, Probabilistic, Dynamic, ...*, in "CONCUR", R. M. AMADIO, D. LUGIEZ (editors), Lecture Notes in Computer Science, Springer, 2003, vol. 2761, pp. 187-188

[51] N. A. LYNCH, E. W. STARK. *A Proof of the Kahn Principle for Input/Output Automata*, in "Inf. Comput.", 1989, vol. 82, n^{O} 1, pp. 81-92

[52] Z. MANNA, A. PNUELI. *Temporal verification of reactive systems: Safety*, Springer, 1995

[53] B. MEYER. *Applying "Design by Contract"*, in "Computer", October 1992, vol. 25, n^{o} 10, pp. 40–51, http://dx.doi.org/10.1109/2.161279

[54] P. NUZZO, A. L. SANGIOVANNI-VINCENTELLI, X. SUN, A. PUGGELLI. *Methodology for the Design of Analog Integrated Interfaces Using Contracts*, in "IEEE Sensors Journal", Dec. 2012, vol. 12, n^{o} 12, pp. 3329–3345

[55] A. ROBINSON. *Non-Standard Analysis*, Princeton Landmarks in Mathematics, 1996, ISBN 0-691-04490-2

[56] E. SIKORA, B. TENBERGEN, K. POHL. *Industry needs and research directions in requirements engineering for embedded systems*, in "Requirements Engineering", 2012, vol. 17, pp. 57–78, http://link.springer.com/article/10.1007/s00766-011-0144-x

[57] L. DE ALFARO. *Game Models for Open Systems*, in "Verification: Theory and Practice", Lecture Notes in Computer Science, Springer, 2003, vol. 2772, pp. 269-289

[58] L. DE ALFARO, T. A. HENZINGER. *Interface automata*, in "Proc. of the 9th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE'01)", ACM Press, 2001, pp. 109–120

[59] L. DE ALFARO, T. A. HENZINGER. *Interface-based design*, in "In Engineering Theories of Software Intensive Systems, proceedings of the Marktoberdorf Summer School", Kluwer, 2004

[60] L. DE ALFARO, T. A. HENZINGER, M. STOELINGA. *Timed Interfaces*, in "Proc. of the 2nd International Workshop on Embedded Software (EMSOFT'02)", Lecture Notes in Computer Science, Springer, 2002, vol. 2491, pp. 108–122