IN PARTNERSHIP WITH:
CNRS

Ecole normale supérieure de Cachan

Activity Report 2013

Project-Team MEXICO

Modeling and Exploitation of Interaction and Concurrency

IN COLLABORATION WITH: Laboratoire specification et vérification (LSV)

RESEARCH CENTER
Saclay - Île-de-France

THEME
Proofs and Verification

Table of contents

Project-Team MEXICO

Keywords: Concurrency, Discrete Event Systems, Distributed System, Formal Methods, Model Of Computation

The project team is located at LSV, a joint Inria-CNRS-ENSC laboratory at ENS Cachan.

Creation of the Team: 2009 March 01*, updated into Project-Team:* 2011 January 01.

1. Members

Research Scientists
Stefan Haar [Team leader, Inria, Senior Researcher, HdR]
Benedikt Bollig [CNRS, Researcher]

Faculty Members
Paul Gastin [ENS Cachan, Professor, HdR]
Serge Haddad [ENS Cachan, Professor, HdR]
Thomas Chatain [ENS Cachan, Associate Professor, HdR]
Claudine Picaronny [ENS Cachan, Associate Professor, member since Sep 2013]
Stefan Schwoon [ENS Cachan, Associate Professor, HdR]

Engineer
Alban Linard [Inria]

PhD Students
Benoît Barbot [ENS Cachan]
Aiswarya Cyriac [ENS Cachan, until Dec 2013]
Benjamin Monmege [ENS Cachan, until Aug 2013]
Hernan Ponce de Leon [Inria]
César Rodríguez [ENS Cachan, until Oct 2013]
Simon Theissing [Inria, grant by Institut de Recherche SystemX, from Sep 2013]

Post-Doctoral Fellow
Loïg Jezequel [Inria, from Dec 2013]

Administrative Assistant
Thida Iem [Inria]

Other
Gonzalo Amadio [Inria Intern, student at Rosario University (Argentina), from Mar 2013 until Jul 2013]

2. Overall Objectives

2.1. Scientific Objectives

2.1.1. Introduction

In the increasingly networked world, reliability of applications becomes ever more critical as the number of users of, e.g., communication systems, web services, transportation etc., grows steadily. Management of networked systems, in a very general sense of the term, therefore is a crucial task, but also a difficult one.

MExICo strives to take advantage of distribution by orchestrating cooperation between different agents that observe local subsystems, and interact in a localized fashion.

The need for applying formal methods in the analysis and management of complex systems has long been recognized. It is with much less unanimity that the scientific community embraces methods based on asynchronous and distributed models. Centralized and sequential modeling still prevails.

However, we observe that crucial applications have increasing numbers of users, that networks providing services grow fast both in the number of participants and the physical size and degree of spatial distribution. Moreover, traditional *isolated* and *proprietary* software products for local systems are no longer typical for emerging applications.

In contrast to traditional centralized and sequential machinery for which purely functional specifications are efficient, we have to account for applications being provided from diverse and non-coordinated sources. Their distribution (e.g. over the Web) must change the way we verify and manage them. In particular, one cannot ignore the impact of quantitative features such as delays or failure likelihoods on the functionalities of composite services in distributed systems.

We thus identify three main characteristics of complex distributed systems that constitute research challenges:

- *Concurrency* of behavior;
- *Interaction* of diverse and semi-transparent components; and
- management of *Quantitative* aspects of behavior.

2.1.2. Concurrency

The increasing size and the networked nature of communication systems, controls, distributed services, etc. confront us with an ever higher degree of parallelism between local processes. This field of application for our work includes telecommunication systems and composite web services. The challenge is to provide sound theoretical foundations and efficient algorithms for management of such systems, ranging from controller synthesis and fault diagnosis to integration and adaptation. While these tasks have received considerable attention in the *sequential* setting, managing *non-sequential* behavior requires profound modifications for existing approaches, and often the development of new approaches altogether. We see concurrency in distributed systems as an opportunity rather than a nuisance. Our goal is to *exploit* asynchronicity and distribution as an advantage. Clever use of adequate models, in particular *partial order semantics* (ranging from Mazurkiewicz traces to event structures to MSCs) actually helps in practice. In fact, the partial order vision allows us to make causal precedence relations explicit, and to perform diagnosis and test for the dependency between events. This is a conceptual advantage that interleaving-based approaches cannot match. The two key features of our work will be *(i)* the exploitation of concurrency by using asynchronous models with partial order semantics, and *(ii)* distribution of the agents performing management tasks.

2.1.3. Interaction

Systems and services exhibit non-trivial *interaction* between specialized and heterogeneous components. A coordinated interplay of several components is required; this is challenging since each of them has only a limited, partial view of the system's configuration. We refer to this problem as *distributed synthesis* or *distributed control*. An aggravating factor is that the structure of a component might be semi-transparent, which requires a form of *grey box management*.

2.1.4. Quantitative Features

Besides the logical functionalities of programs, the *quantitative* aspects of component behavior and interaction play an increasingly important role.

- *Real-time* properties cannot be neglected even if time is not an explicit functional issue, since transmission delays, parallelism, etc, can lead to time-outs striking, and thus change even the logical course of processes. Again, this phenomenon arises in telecommunications and web services, but also in transport systems.
- In the same contexts, *probabilities* need to be taken into account, for many diverse reasons such as unpredictable functionalities, or because the outcome of a computation may be governed by race conditions.

- Last but not least, constraints on *cost* cannot be ignored, be it in terms of money or any other limited resource, such as memory space or available CPU time.

2.1.5. Evolution and Perspectives

Since the creation of *MExICo*, the weight of *quantitative* aspects in all parts of our activities has grown, be it in terms of the models considered (weighted automata and logics), be it in transforming verification or diagnosis verdict into probabilistic statements (probabilistic diagnosis, statistical model checking), or within the recently started SystemX cooperation on supervision in multi-modal transport systems. This trend is certain to continue over the next couple of years, along with the growing importance of diagnosis and control issues.

In another development, the theory and use of partial order semantics has gained momentum in the past four years, and we intend to further strengthen our efforts and contacts in this domain to further develop and apply partial-order based deduction methods.

As concerns the study of interaction, our progress has been thus far less in the domain of *distributed* approaches than in the analysis of *system composition*, such as in networks of untimed or timed automata. While continuing this line of study, we also intend to turn more strongly towards distributed *algorithms*, namely in terms of parametrized verification methods.

2.2. Highlights of the Year

- We have made two major progresses in diagnosis this year:
  - For non-diagnosable discrete event systems, *active* diagnosis aims at synthesizing a partial-observabion based control for the system in order to make it diagnosable. While some solutions had already been proposed for the active diagnosis problem, their complexity remained to be improved. In [40], we solved both the active diagnosability decision problem and the active diagnoser synthesis problem, proving that (1) our procedures are optimal w.r.t. to computational complexity, and (2) the memory required for the active diagnoser produced by the synthesis is minimal. Furthermore, focusing on the minimal delay before detection, we establish that the memory required for any active diagnoser achieving this delay may be highly greater than the previous one. So we refine our construction to build with the same complexity and memory requirement an active diagnoser that realizes a delay bounded by twice the minimal delay. An extension to *probabilistic* systems has been accepted to *FoSSaCS 2014.*
  - In [41], we present a methodology for fault diagnosis in concurrent, partially observable systems with additional fairness constraints. In this *weak* diagnosis, one asks whether a concurrent chronicle of observed events allows to determine that a non-observable fault will inevitably occur, sooner or later, on any maximal system run compatible with the observation. The approach builds on strengths and techniques of unfoldings of safe Petri nets, striving to compute a compact prefix of the unfolding that carries sufficient information for the diagnosis algorithm. Our work extends and generalizes the unfolding-based diagnosis approaches by Benveniste et al. as well as Esparza and Kern. Both of these focused mostly on the use of sequential observations, in particular did not exploit the capacity of unfoldings to reveal inevitable occurrences of concurrent or future events studied by Balaguer et al. [19]. Our diagnosis method captures such indirect, revealed dependencies. We develop theoretical foundations and an algorithmic solution to the diagnosis problem, and present a SAT solving method for practical diagnosis with our approach.
- The article *Complexity Analysis of Continuous Petri Nets* by Estébaliz Fraca and Serge Haddad [39] received the *outstanding paper award* at the *International Conference on Application and Theory of Petri Nets and Concurrency, June 24-28, 2013, Milano, Italy.*

BEST PAPER AWARD :

[39] **Complexity Analysis of Continuous Petri Nets in 34th International Conference on Applications and Theory of Petri Nets (ICATPN'13)**. E. FRACA, S. HADDAD.

3. Research Program

3.1. Concurrency

Participants: Benedikt Bollig, Thomas Chatain, Aiswarya Cyriac, Paul Gastin, Stefan Haar, Serge Haddad, Hernán Ponce de Léon, Stefan Schwoon, César Rodríguez.

Concurrency: Property of systems allowing some interacting processes to be executed in parallel.

Diagnosis: The process of deducing from a partial observation of a system aspects of the internal states or events of that system; in particular, *fault diagnosis* aims at determining whether or not some non-observable fault event has occurred.

Conformance Testing: Feeding dedicated input into an implemented system IS and deducing, from the resulting output of I, whether I respects a formal specification S.

3.1.1. Introduction

It is well known that, whatever the intended form of analysis or control, a *global* view of the system state leads to overwhelming numbers of states and transitions, thus slowing down algorithms that need to explore the state space. Worse yet, it often blurs the mechanics that are at work rather than exhibiting them. Conversely, respecting concurrency relations avoids exhaustive enumeration of interleavings. It allows us to focus on 'essential' properties of non-sequential processes, which are expressible with causal precedence relations. These precedence relations are usually called causal (partial) orders. Concurrency is the explicit absence of such a precedence between actions that do not have to wait for one another. Both causal orders and concurrency are in fact essential elements of a specification. This is especially true when the specification is constructed in a distributed and modular way. Making these ordering relations explicit requires to leave the framework of state/interleaving based semantics. Therefore, we need to develop new dedicated algorithms for tasks such as conformance testing, fault diagnosis, or control for distributed discrete systems. Existing solutions for these problems often rely on centralized sequential models which do not scale up well.

3.1.2. Diagnosis

Participants: Benedikt Bollig, Stefan Haar, Serge Haddad, Loig Jezequel, Hernán Ponce de Léon, César Rodríguez, Stefan Schwoon.

Fault Diagnosis for discrete event systems is a crucial task in automatic control. Our focus is on *event oriented* (as opposed to *state oriented*) model-based diagnosis, asking e.g. the following questions:
given a - potentially large - *alarm pattern* formed of observations,

- what are the possible *fault scenarios* in the system that *explain* the pattern ?
- Based on the observations, can we deduce whether or not a certain - invisible - fault has actually occurred ?

Model-based diagnosis starts from a discrete event model of the observed system - or rather, its relevant aspects, such as possible fault propagations, abstracting away other dimensions. From this model, an extraction or unfolding process, guided by the observation, produces recursively the explanation candidates.

In asynchronous partial-order based diagnosis with Petri nets [71], [72], [76], one unfolds the *labelled product* of a Petri net model $\mathcal{N}$ and an observed alarm pattern $\mathcal{A}$, also in Petri net form. We obtain an acyclic net giving partial order representation of the behaviors compatible with the alarm pattern. A recursive online procedure filters out those runs *(configurations)* that explain *exactly* $\mathcal{A}$. The Petri-net based approach generalizes to dynamically evolving topologies, in dynamical systems modeled by graph grammars, see [3]

3.1.2.1. Observability and Diagnosability

Diagnosis algorithms have to operate in contexts with low observability, i.e., in systems where many events are invisible to the supervisor. Checking *observability* and *diagnosability* for the supervised systems is therefore a crucial and non-trivial task in its own right. Analysis of the relational structure of occurrence nets allows us to check whether the system exhibits sufficient visibility to allow diagnosis. Developing efficient methods for both verification of *diagnosability checking* under concurrency, and the *diagnosis* itself for distributed, composite and asynchronous systems, is an important field for *MExICo*.

3.1.2.2. Distribution

Distributed computation of unfoldings allows one to factor the unfolding of the global system into smaller *local* unfoldings, by local supervisors associated with sub-networks and communicating among each other. In [72], [58], elements of a methodology for distributed computation of unfoldings between several supervisors, underwritten by algebraic properties of the category of Petri nets have been developed. Generalizations, in particular to Graph Grammars, are still do be done.

Computing diagnosis in a distributed way is only one aspect of a much vaster topic, that of *distributed diagnosis* (see [68], [80]). In fact, it involves a more abstract and often indirect reasoning to conclude whether or not some given invisible fault has occurred. Combination of local scenarios is in general not sufficient: the global system may have behaviors that do not reveal themselves as faulty (or, dually, non-faulty) on any local supervisor's domain (compare [56], [61]). Rather, the local diagnosers have to join all *information* that is available to them locally, and then deduce collectively further information from the combination of their views. In particular, even the *absence* of fault evidence on all peers may allow to deduce fault occurrence jointly, see [84], [85]. Automatizing such procedures for the supervision and management of distributed and locally monitored asynchronous systems is a long-term goal to which *MExICo* hopes to contribute.

3.1.3. Verification of Concurrent Recursive Programs

Participants: Benedikt Bollig, Aiswarya Cyriac, Paul Gastin, César Rodríguez, Stefan Schwoon.

(How about Thomas and Stefan H ?)

3.1.3.1. Contextual nets

Assuring the correctness of concurrent systems is notoriously difficult due to the many unforeseeable ways in which the components may interact and the resulting state-space explosion. A well-established approach to alleviate this problem is to model concurrent systems as Petri nets and analyse their unfoldings, essentially an acyclic version of the Petri net whose simpler structure permits easier analysis [70].

However, Petri nets are inadequate to model concurrent read accesses to the same resource. Such situations often arise naturally, for instance in concurrent databases or in asynchronous circuits. The encoding tricks typically used to model these cases in Petri nets make the unfolding technique inefficient. Contextual nets, which explicitly do model concurrent read accesses, address this problem. Their accurate representation of concurrency makes contextual unfoldings up to exponentially smaller in certain situations. An abstract algorithm for contextual unfoldings was first given in [57]. In recent work, we further studied this subject from a theoretical and practical perspective, allowing us to develop concrete, efficient data structures and algorithms and a tool (Cunf) that improves upon existing state of the art. This work led to the PhD thesis of César Rodríguez [15]

Contexutal unfoldings deal well with two sources of state-space explosion: concurrency and shared resources. Recently, we proposed an improved data structure, called *contextual merged processes* (CMP) to deal with a third source of state-space explosion, i.e. sequences of choices. The work on CMP [45] is currently at an abstract level. In the short term, we want to put this work into practice, requiring some theoretical groundwork, as well as programming and experimentation.

Another well-known approach to verifying concurrent systems is *partial-order reduction*, exemplified by the tool SPIN. Although it is known that both partial-order reduction and unfoldings have their respective strengths and weaknesses, we are not aware of any conclusive comparison between the two techniques. Spin comes with a high-level modeling language having an explicit notion of processes, communication channels, and variables. Indeed, the reduction techniques implemented in Spin exploit the specific properties of these features. On the other side, while there exist highly efficient tools for unfoldings, Petri nets are a relatively general low-level formalism, so these techniques do not exploit properties of higher language features. Our work on contextual unfoldings and CMPs represents a first step to make unfoldings exploit richer models. In the long run, we wish raise the unfolding technique to a suitable high-level modelling language and develop appropriate tool support.

3.1.3.2. Concurrent Recursive Programs

In a DIGITEO PhD project, we will study logical specification formalisms for concurrent recursive programs. With the advent of multi-core processors, the analysis and synthesis of such programs is becoming more and more important. However, it cannot be achieved without more comprehensive formal mathematical models of concurrency and parallelization. Most existing approaches have in common that they restrict to the analysis of an over- or underapproximation of the actual program executions and do not focus on a behavioral semantics. In particular, temporal logics have not been considered. Their design and study will require the combination of prior works on logics for sequential recursive programs and concurrent finite-state programs.

3.1.4. Testing

Participants: Benedikt Bollig, Paul Gastin, Stefan Haar, Hernán Ponce de Léon.

3.1.4.1. Introduction

The gap between specification and implementation is at the heart of research on formal testing. The general *conformance testing problem* can be defined as follows: Does an implementation $\mathcal{M}'$ conform a given specification $\mathcal{M}$? Here, both $\mathcal{M}$ and $\mathcal{M}'$ are assumed to have input and output channels. The formal model $\mathcal{M}$ of the specification is entirely known and can be used for analysis. On the other hand, the implementation $\mathcal{M}'$ is unknown but interacts with the environment through observable input and output channels. So the behavior of $\mathcal{M}'$ is partially controlled by input streams, and partially observable via output streams. The Testing problem consists in computing, from the knowledge of $\mathcal{M}$, *input streams* for $\mathcal{M}'$ such that observation of the resulting output streams from $\mathcal{M}'$ allows to determine whether $\mathcal{M}'$ conforms to $\mathcal{M}$ as intended.

In this project, we focus on distributed or asynchronous versions of the conformance testing problem. There are two main difficulties. First, due to the distributed nature of the system, it may not be possible to have a unique global observer for the outcome of a test. Hence, we may need to use *local* observers which will record only *partial views* of the execution. Due to this, it is difficult or even impossible to reconstruct a coherent global execution. The second difficulty is the lack of global synchronization in distributed asynchronous systems. Up to now, models were described with I/O automata having a centralized control, hence inducing global synchronizations.

3.1.4.2. Asynchronous Testing

Since 2006 and in particular during his sabbatical stay at the University of Ottawa, Stefan Haar has been working with Guy-Vincent Jourdan and Gregor v. Bochmann of UOttawa and Claude Jard of IRISA on asynchronous testing. In the synchronous (sequential) approach, the model is described by an I/O automaton with a centralized control and transitions labeled with individual input or output actions. This approach has known limitations when inputs and outputs are distributed over remote sites, a feature that is characteristic of , e.g., web computing. To account for concurrency in the system, they have developed in [78], [62] asynchronous conformance testing for automata with transitions labeled with (finite) partial orders of I/O. Intuitively, this is a “big step” semantics where each step allows concurrency but the system is synchronized before the next big step. This is already an important improvement on the synchronous setting. The non-trivial challenge is now to cope with fully asynchronous specifications using models with decentralized control such as Petri nets.

3.1.4.3. Near Future

Completion of asynchronous testing in the setting without any big-step synchronization, and an improved understanding of the relations and possible interconnections between local (i.e. distributed) and asynchronous (centralized) testing. This is the objective of the *TECSTES* project (2011-2014), funded by a DIGITEO *DIM/LSC* grant, and which involves Hernán Ponce de Léon and Stefan Haar of *MExICo*, and Delphine Longuet at LRI, University Paris-Sud/Orsay. We have extended several well known conformance (ioco style) relations for sequential models to models that can handle concurrency (labeled event structures). Two semantics (interleaving and partial order) were presented for every relation. With the interleaving semantics, the relations we obtained boil down to the same relations defined for labeled transition systems, since they focus on sequences of actions. The only advantage of using labeled event structures as a specification formalism for testing remains in the conciseness of the concurrent model with respect to a sequential one. As far as testing is concerned, the benefit is low since every interleaving has to be tested. By contrast, under the partial order semantics, the relations we obtain allow to distinguish explicitly implementations where concurrent actions are implemented concurrently, from those where they are interleaved, i.e. implemented sequentially. Therefore, these relations will be of interest when designing distributed systems, since the natural concurrency between actions that are performed in parallel by different processes can be taken into account. In particular, the fact of being unable to control or observe the order between actions taking place on different processes will not be considered as an impediment for testing. We have developped a complete testing framework for concurrent systems, which included the notions of test suites and test cases. We studied what kind of systems are testable in such a framework, and we have proposed sufficient conditions for obtaining a complete test suite as well as an algorithm to construct a test suite with such properties.

A mid-to long term goal (not yet to achieve in this four-year term, and which may or may not be addressed by *MExICo* depending on the availability of staff for this subject) is the comprehensive formalization of testing and testability in asynchronous systems with distributed architecture and test protocols.

3.2. Interaction

Participants: Benedikt Bollig, Thomas Chatain, Paul Gastin, Stefan Haar, Serge Haddad.

3.2.1. Introduction

Systems and services exhibit non-trivial *interaction* between specialized and heterogeneous components. This interplay is challenging for several reasons. On one hand, a coordinated interplay of several components is required, though each has only a limited, partial view of the system's configuration. We refer to this problem as *distributed synthesis* or *distributed control*. An aggravating factor is that the structure of a component might be semi-transparent, which requires a form of *grey box management*.

Interaction, one of the main characteristics of systems under consideration, often involves an environment that is not under the control of cooperating services. To achieve a common goal, the services need to agree upon a strategy that allows them to react appropriately regardless of the interactions with the environment. Clearly, the notions of opponents and strategies fall within *game theory*, which is naturally one of our main tools in exploring interaction. We will apply to our problems techniques and results developed in the domains of distributed games and of games with partial information. We will consider also new problems on games that arise from our applications.

3.2.2. Distributed Control

Participants: Benedikt Bollig, Thomas Chatain, Paul Gastin, Stefan Haar.

Program synthesis, as introduced by Church [67] aims at deriving directly an implementation from a specification, allowing the implementation to be correct by design. When the implementation is already at hand but choices remain to be resolved at run time then the problem becomes controller synthesis. Both program and controller synthesis have been extensively studied for sequential systems. In a distributed setting, we need to synthesize a distributed program or distributed controllers that interact locally with the system components. The main difficulty comes from the fact that the local controllers/programs have only a partial

view of the entire system. This is also an old problem largely considered undecidable in most settings [83], [79], [82], [73], [75].

Actually, the main undecidability sources come from the fact that this problem was addressed in a synchronous setting using global runs viewed as sequences. In a truly distributed system where interactions are asynchronous we have recently obtained encouraging decidability results [74],[8]. This is a clear witness where concurrency may be exploited to obtain positive results. It is essential to specify expected properties directly in terms of causality revealed by partial order models of executions (MSCs or Mazurkiewicz traces). We intend to develop this line of research with the ambitious aim to obtain decidability for all natural systems and specifications. More precisely, we will identify natural hypotheses both on the architecture of our distributed system and on the specifications under which the distributed program/controller synthesis problem is decidable. This should open the way to important applications, e.g., for distributed control of embedded systems.

3.2.3. Adaptation and Grey box management

Participants: Stefan Haar, Serge Haddad.

Contrary to mainframe systems or monolithic applications of the past, we are experiencing and using an increasing number of services that are performed not by one provider but rather by the interaction and cooperation of many specialized components. As these components come from different providers, one can no longer assume all of their internal technologies to be known (as it is the case with proprietary technology). Thus, in order to compose e.g. orchestrated services over the web, to determine violations of specifications or contracts, to adapt existing services to new situations etc, one needs to analyze the interaction behavior of *boxes* that are known only through their public interfaces. For their semi-transparent-semi-opaque nature, we shall refer to them as **grey boxes**. While the concrete nature of these boxes can range from vehicles in a highway section to hotel reservation systems, the tasks of *grey box management* have universal features allowing for generalized approaches with formal methods. Two central issues emerge:

- Abstraction: From the designer point of view, there is a need for a trade-off between transparency (no abstraction) in order to integrate the box in different contexts and opacity (full abstraction) for security reasons.
- Adaptation: Since a grey box gives a partial view about the behavior of the component, even if it is not immediately useable in some context, the design of an adaptator is possible. Thus the goal is the synthesis of such an adaptator from a formal specification of the component and the environment.

Our work on direct modeling and handling of "grey boxes" via modal models (see [69]) was halted when Dorsaf El-Hog stopped her PhD work to leave academia, and has not resumed for lack of staff. However, it should be noted that semi-transparent system management in a larger sense remains an active field for the team, witness in particular our work on diagnosis and testing.

3.3. Management of Quantitative Behavior

Participants: Benedikt Bollig, Thomas Chatain, Paul Gastin, Stefan Haar, Serge Haddad, Benjamin Monmege.

3.3.1. Introduction

Besides the logical functionalities of programs, the *quantitative* aspects of component behavior and interaction play an increasingly important role.

- *Real-time* properties cannot be neglected even if time is not an explicit functional issue, since transmission delays, parallelism, etc, can lead to time-outs striking, and thus change even the logical course of processes. Again, this phenomenon arises in telecommunications and web services, but also in transport systems.
- In the same contexts, *probabilities* need to be taken into account, for many diverse reasons such as unpredictable functionalities, or because the outcome of a computation may be governed by race conditions.
- Last but not least, constraints on *cost* cannot be ignored, be it in terms of money or any other limited resource, such as memory space or available CPU time.

Traditional mainframe systems were proprietary and (essentially) localized; therefore, impact of delays, unforeseen failures, etc. could be considered under the control of the system manager. It was therefore natural, in verification and control of systems, to focus on *functional* behavior entirely.

With the increase in size of computing system and the growing degree of compositionality and distribution, quantitative factors enter the stage:

- calling remote services and transmitting data over the web creates *delays*;
- remote or non-proprietary components are not "deterministic", in the sense that their behavior is uncertain.

Time and *probability* are thus parameters that management of distributed systems must be able to handle; along with both, the *cost* of operations is often subject to restrictions, or its minimization is at least desired. The mathematical treatment of these features in distributed systems is an important challenge, which *MExICo* is addressing; the following describes our activities concerning probabilistic and timed systems. Note that cost optimization is not a current activity but enters the picture in several intended activities.

3.3.2. Probabilistic distributed Systems

Participants: Stefan Haar, Serge Haddad, Claudine Picaronny.

3.3.2.1. Non-sequential probabilistic processes

Practical fault diagnosis requires to select explanations of *maximal likelihood*. For partial-order based diagnosis, this leads therefore to the question what the probability of a given partially ordered execution is. In Benveniste et al. [60], [54], we presented a model of stochastic processes, whose trajectories are partially ordered, based on local branching in Petri net unfoldings; an alternative and complementary model based on Markov fields is developed in [77], which takes a different view on the semantics and overcomes the first model's restrictions on applicability.

Both approaches abstract away from real time progress and randomize choices in *logical* time. On the other hand, the relative speed - and thus, indirectly, the real-time behavior of the system's local processes - are crucial factors determining the outcome of probabilistic choices, even if non-determinism is absent from the system.

In another line of research [64] we have studied the likelihood of occurrence of non-sequential runs under random durations in a stochastic Petri net setting. It remains to better understand the properties of the probability measures thus obtained, to relate them with the models in logical time, and exploit them e.g. in *diagnosis*.

3.3.2.2. Distributed Markov Decision Processes

Distributed systems featuring non-deterministic and probabilistic aspects are usually hard to analyze and, more specifically, to optimize. Furthermore, high complexity theoretical lower bounds have been established for models like partially observed Markovian decision processes and distributed partially observed Markovian decision processes. We believe that these negative results are consequences of the choice of the models rather than the intrinsic complexity of problems to be solved. Thus we plan to introduce new models in which the associated optimization problems can be solved in a more efficient way. More precisely, we start by studying connection protocols weighted by costs and we look for online and offline strategies for optimizing the mean cost to achieve the protocol. We have been cooperating on this subject with the SUMO team at Inria Rennes; in the joint work [26]; there, we strive to synthesize for a given MDP a control so as to guarantee a specific stationary behavior, rather than - as is usually done - so as to maximize some reward.

3.3.3. Large scale probabilistic systems

Addressing large-scale probabilistic systems requires to face state explosion, due to both the discrete part and the probabilistic part of the model. In order to deal with such systems, different approaches have been proposed:

- Restricting the synchronization between the components as in queuing networks allows to express the steady-state distribution of the model by an analytical formula called a product-form [59].

- Some methods that tackle with the combinatory explosion for discrete-event systems can be generalized to stochastic systems using an appropriate theory. For instance symmetry based methods have been generalized to stochastic systems with the help of aggregation theory [66].
- At last simulation, which works as soon as a stochastic operational semantic is defined, has been adapted to perform statistical model checking. Roughly speaking, it consists to produce a confidence interval for the probability that a random path fulfills a formula of some temporal logic [86] .

We want to contribute to these three axes: (1) we are looking for product-forms related to systems where synchronization are more involved (like in Petri nets), see [24]; (2) we want to adapt methods for discrete-event systems that require some theoretical developments in the stochastic framework and, (3) we plan to address some important limitations of statistical model checking like the expressiveness of the associated logic and the handling of rare events.

3.3.4. Real time distributed systems

Nowadays, software systems largely depend on complex timing constraints and usually consist of many interacting local components. Among them, railway crossings, traffic control units, mobile phones, computer servers, and many more safety-critical systems are subject to particular quality standards. It is therefore becoming increasingly important to look at networks of timed systems, which allow real-time systems to operate in a distributed manner.

Timed automata are a well-studied formalism to describe reactive systems that come with timing constraints. For modeling distributed real-time systems, networks of timed automata have been considered, where the local clocks of the processes usually evolve at the same rate [81] [65]. It is, however, not always adequate to assume that distributed components of a system obey a global time. Actually, there is generally no reason to assume that different timed systems in the networks refer to the same time or evolve at the same rate. Any component is rather determined by local influences such as temperature and workload.

3.3.4.1. Distributed timed systems with independently evolving clocks

Participants: Benedikt Bollig, Paul Gastin.

A first step towards formal models of distributed timed systems with independently evolving clocks was done in [55]. As the precise evolution of local clock rates is often too complex or even unknown, the authors study different semantics of a given system: The *existential semantics* exhibits all those behaviors that are possible under *some* time evolution. The *universal semantics* captures only those behaviors that are possible under *all* time evolutions. While emptiness and universality of the universal semantics are in general undecidable, the existential semantics is always regular and offers a way to check a given system against safety properties. A decidable under-approximation of the universal semantics, called *reactive semantics*, is introduced to check a system for liveness properties. It assumes the existence of a *global* controller that allows the system to react upon local time evolutions. A short term goal is to investigate a *distributed* reactive semantics where controllers are located at processes and only have local views of the system behaviors.

Several questions, however, have not yet been tackled in this previous work or remain open. In particular, we plan to exploit the power of synchronization via local clocks and to investigate the *synthesis problem*: For which (global) specifications $\mathcal{S}$ can we generate a distributed timed system with independently evolving clocks $\mathcal{A}$ (over some given system architecture) such that both the reactive and the existential semantics of $\mathcal{A}$ are precisely (the semantics of) $\mathcal{S}$? In this context, it will be favorable to have partial-order based specification languages and a partial-order semantics for distributed timed systems. The fact that clocks are not shared may allow us to apply partial-order–reduction techniques.

If, on the other hand, a system is already given and complemented with a specification, then one is usually interested in controlling the system in such a way that it meets its specification. The interaction between the actual *system* and the *environment* (i.e., the local time evolution) can now be understood as a 2-player game: the system's goal is to guarantee a behavior that conforms with the specification, while the environment aims at violating the specification. Thus, building a controller of a system actually amounts to computing winning strategies in imperfect-information games with infinitely many states where the unknown or unpredictable

evolution of time reflects an imperfect information of the environment. Only few efforts have been made to tackle those kinds of games. One reason might be that, in the presence of imperfect information and infinitely many states, one is quickly confronted with undecidability of basic decision problems.

3.3.4.2. Implementation of Real-Time Concurrent Systems

Participants: Thomas Chatain, Stefan Haar, Serge Haddad.

This is one of the tasks of the ANR ImpRo.

Formal models for real-time systems, like timed automata and time Petri nets, have been extensively studied and have proved their interest for the verification of real-time systems. On the other hand, the question of using these models as specifications for designing real-time systems raises some difficulties. One of those comes from the fact that the real-time constraints introduce some artifacts and because of them some syntactically correct models have a formal semantics that is clearly unrealistic. One famous situation is the case of Zeno executions, where the formal semantics allows the system to do infinitely many actions in finite time. But there are other problems, and some of them are related to the distributed nature of the system. These are the ones we address here.

One approach to implementability problems is to formalize either syntactical or behavioral requirements about what should be considered as a reasonable model, and reject other models. Another approach is to adapt the formal semantics such that only realistic behaviors are considered.

These techniques are preliminaries for dealing with the problem of implementability of models. Indeed implementing a model may be possible at the cost of some transformation, which make it suitable for the target device. By the way these transformations may be of interest for the designer who can now use high-level features in a model of a system or protocol, and rely on the transformation to make it implementable.

We aim at formalizing and automating translations that preserve both the timed semantics and the concurrent semantics. This effort is crucial for extending concurrency-oriented methods for logical time, in particular for exploiting partial order properties. In fact, validation and management - in a broad sense - of distributed systems is not realistic *in general* without understanding and control of their real-time dependent features; the link between real-time and logical-time behaviors is thus crucial for many aspects of *MExICo*'s work.

3.3.5. Weighted Automata and Weighted Logics

Participants: Benedikt Bollig, Paul Gastin, Benjamin Monmege.

Time and probability are only two facets of quantitative phenomena. A generic concept of adding weights to qualitative systems is provided by the theory of weighted automata [53]. They allow one to treat probabilistic or also reward models in a unified framework. Unlike finite automata, which are based on the Boolean semiring, weighted automata build on more general structures such as the natural or real numbers (equipped with the usual addition and multiplication) or the probabilistic semiring. Hence, a weighted automaton associates with any possible behavior a weight beyond the usual Boolean classification of "acceptance" or "non-acceptance". Automata with weights have produced a well-established theory and come, e.g., with a characterization in terms of rational expressions, which generalizes the famous theorem of Kleene in the unweighted setting. Equipped with a solid theoretical basis, weighted automata finally found their way into numerous application areas such as natural language processing and speech recognition, or digital image compression.

What is still missing in the theory of weighted automata are satisfactory connections with verification-related issues such as (temporal) logic and bisimulation that could lead to a general approach to corresponding satisfiability and model-checking problems. A first step towards a more satisfactory theory of weighted systems was done in [63]. That paper, however, does not give definite answers to all the aforementioned problems. It identifies directions for future research that we will be tackling.

4. Application Domains

4.1. Telecommunications

Participants: Stefan Haar, Serge Haddad.

MExICo's research is motivated by problems on system management in several domains:

- In the domain of service oriented computing, it is often necessary to insert some Web service into an existing orchestrated business process, e.g. to replace another component after failures. This requires to ensure, often actively, conformance to the interaction protocol. One therefore needs to synthesize *adaptators* for every component in order to steer its interaction with the surrounding processes.
- Still in the domain of telecommunications, the supervision of a network tends to move from out-of-band technology, with a fixed dedicated supervision infrastructure, to in-band supervision where the supervision process uses the supervised network itself. This new setting requires to revisit the existing supervision techniques using control and diagnosis tools.

We have participated in the Univerself Project (see below) on self-aware networks, and will be searching new cooperations.

4.2. Transport Systems

Participants: Stefan Haar, Serge Haddad, Simon Theissing.

We participate in the project MIC on multi-modal transport systems with in the IRT *System X*, with academic partners UPMC, IFSTTAR and CEA, and several industrial partners including Alstom (project leader), COSMO and Renault. Transportation operators in an urban area need to plan, supervise and steer different means of transportation with respect to several criteria:

- Maximize capacity;
- guarantee punctuality and robustness of service;
- minimize energy consumption.

The systems must achieve these objectives not only under ideal conditions, but also be robust to perturbations (such as a major cultural or sport event creating additional traffic), modifications of routes (roadwork, accidents, demonstrations, ...) and tolerant to technical failures. Therefore, systems must be enabled to raise appropriate alarms upon detection of anomalies, diagnose the type of anomaly and select the appropriate response.

While the above challenges belong already to the tasks of individual operators in the unimodal setting, the rise of and increasing demand for *multi-modal* transports forces to achieve these planning, optimization and control goals not in isolation, but in a cooperative manner, across several operators. The research task here is first to analyze the transportation system regarding the available means, capacities and structures, and so as to identify the impacting factors and interdependencies of the system variables. Based on this analysis, the task is to derive and implement robust planning, with tolerance to technical faults; diagnosis and control strategies that are optimal under several, possibly different, criteria (average case vs worst case performance, energy efficiency, etc.) and allow to adapt to changes e.g. from nominal mode to reduced mode, sensor failures, etc.

5. Software and Platforms

5.1. Software and Platform

5.1.1. Software

5.1.1.1. libalf: the Automata Learning Framework

Participant: Benedikt Bollig [correspondant].

`libalf` is a comprehensive, open-source library for learning finite-state automata covering various well-known learning techniques (such as, Angluin s L^*, Biermann, and RPNI, as well as a novel learning algorithm for NFA. libalf is highly flexible and allows for facilely interchanging learning algorithms and combining domain-specific features in a plug-and-play fashion. Its modular design and its implementation in C++ make it a flexible platform for adding and engineering further, efficient learning algorithms for new target models (e.g., Büchi automata).

Details on `libalf` can be found at http://libalf.informatik.rwth-aachen.de/

5.1.1.2. Mole/Cunf: unfolders for Petri Nets

Participants: Stefan Schwoon [correspondant], César Rodríguez.

Mole computes, given a safe Petri net, a finite prefix of its unfolding. It is designed to be compatible with other tools, such as PEP and the Model-Checking Kit, which are using the resulting unfolding for reachability checking and other analyses. The tool Mole arose out of earlier work on Petri nets. Details on Mole can be found at http://www.lsv.ens-cachan.fr/~schwoon/tools/mole/. Mole served as an experimentation platform for several of our papers this year, notably [38] and [46].

In the context of MExICo, we have created a new tool called Cunf [47], which is able to handle contextual nets, i.e. Petri nets with read arcs [12].While in principle every contextual net can be transformed into an equivalent Petri net and then unfolded using Mole, Cunf can take advantage of their special features to do the job faster and produce a smaller unfolding. Cunf has recently been extended with a verification component that takes advantage of these features; More details can be found at http://www.lsv.ens-cachan.fr/~rodrigue/tools/cunf/. Moreover, Cunf has been integrated into the CosyVerif environment (see section 5.1.2.1). Cunf has also participated in the Model Checking Contest held at the Petri Nets conference in 2013.

5.1.1.3. COSMOS : a Statistical Model Checker for the Hybrid Automata Stochastic Logic

Participant: Benoît Barbot [correspondant].

COSMOS is a statistical model checker for the Hybrid Automata Stochastic Logic (HASL). HASL employs Linear Hybrid Automata (LHA), a generalization of Deterministic Timed Automata (DTA), to describe accepting execution paths of a Discrete Event Stochastic Process (DESP), a class of stochastic models which includes, but is not limited to, Markov chains. As a result HASL verification turns out to be a unifying framework where sophisticated temporal reasoning is naturally blended with elaborate reward-based analysis. COSMOS takes as input a DESP (described in terms of a Generalized Stochastic Petri Net), an LHA and an expression Z representing the quantity to be estimated. It returns a confidence interval estimation of Z; recently, it has been equipped with functionalities for rare event analysis. COSMOS is written in C++ and is freely available to the research community.

Details on COSMOS can be found at http://www.lsv.ens-cachan.fr/~barbot/cosmos/

5.1.2. Platforms

5.1.2.1. CosyVerif

Participants: Serge Haddad, Alban Linard [correspondant], Benoît Barbot.

CosyVerif (http://www.cosyverif.org/) is a platform dedicated to the formal specification and verification of dynamic systems. It allows to specify systems in a graphical editor, using several formalisms (such as automata and Petri nets) and to run verification tools on these models in a dedicated execution server. These tools are mainly developed by researchers of the MeFoSyLoMa group (a Parisian verification group, http://www.mefosyloma.fr/).

The platform is available as installable bundles, that contain both the client, the server, and the tools. It is also usable through two public servers: one with the latest release, one with the development version.

CosyVerif does not only handle several formalisms, but also allows to easily define new ones and integrate them within the platform. To the best of our knowledge, no other verification framework presents such a feature.

It has different kinds of users:

- Tool developers, that are usually researchers, can use the platform to distribute their tools, and have a demonstration version easily available.
- Students can use this platform in modeling and verification courses.
- Industrial case studies are also a target of the CosyVerif platform, in order to promote the practice of formal verification in industry.

The platform is managed by a steering committee consisting of researchers and engineers. This committee decides strategic orientations as well as technical choices.

This year, we have improved the platform in several ways.

- Tools: the platform handles two families of formalisms: automata and Petri nets, both with extensions. It currently integrates 10 tools with 4 new ones this year. Some of them perform structural analyses like invariant computations, while other tools perform behavioural analyses: symbolic reachability graph building, unfolding, stochastic simulations, etc.
- Server: the execution server has been enhanced with asynchronous executions, that allow to disconnect and reconnect the client in long executions. It has also been improved by the ability to communicate between servers to share their available tools.
- Client: a new command line client has been developed for scripting the executions.
- Usability: the client and server are new distributed as one bundle that can be installed easily on all platforms. The server and its tools are embedded within a virtual machine to achieve this portability.

All the developed software are open source and free software tools.

Two engineers have worked this year on CosyVerif:

- Francis Hulin-Hubard, part-time (CNRS engineer);
- Alban Linard, full-time (Inria engineer).

CosyVerif has been the subject of two international communications [28], [29]. It has been presented at the french-speaking PhD school ETR'2013 in Toulouse. and used for teaching in the master SAR of University Pierre et Marie Curie.

6. New Results

6.1. Diagnosis

- For non-diagnosable discrete event systems, *active* diagnosis aims at synthesizing a partial-observabion based control for the system in order to make it diagnosable. While some solutions had already been proposed for the active diagnosis problem, their complexity remained to be improved. In [40], we solved both the active diagnosability decision problem and the active diagnoser synthesis problem, proving that (1) our procedures are optimal w.r.t. to computational complexity, and (2) the memory required for the active diagnoser produced by the synthesis is minimal. Furthermore, focusing on the minimal delay before detection, we establish that the memory required for any active diagnoser achieving this delay may be highly greater than the previous one. So we refine our construction to build with the same complexity and memory requirement an active diagnoser that realizes a delay bounded by twice the minimal delay. An extension to *probabilistic* systems has been accepted to *FoSSaCS 2014*.
- In [41], we present a methodology for fault diagnosis in concurrent, partially observable systems with additional fairness constraints. In this *weak* diagnosis, one asks whether a concurrent chronicle of observed events allows to determine that a non-observable fault will inevitably occur, sooner or later, on any maximal system run compatible with the observation. The approach builds on strengths and techniques of unfoldings of safe Petri nets, striving to compute a compact prefix of the unfolding that carries sufficient information for the diagnosis algorithm. Our work extends and generalizes the unfolding-based diagnosis approaches by Benveniste et al. as well as Esparza and Kern. Both of these focused mostly on the use of sequential observations, in particular did not exploit the capacity of unfoldings to reveal inevitable occurrences of concurrent or future events studied by Balaguer et al. [19]. Our diagnosis method captures such indirect, revealed dependencies. We develop theoretical foundations and an algorithmic solution to the diagnosis problem, and present a SAT solving method for practical diagnosis with our approach. The algorithms to check diagnosability of concurrent

systems are usually performed by local diagnoses of twin plant communicating with each other, directly or through a co- ordinator, and by that means pooling together the observations. Parallel analysis of diagnosability [43] takes advantage of the distribution of the system allowing to decide the diagnosability of the whole system in terms of the diagnosability of smaller systems.

6.2. Testing for Concurrent Systems

6.2.1. Model Based Testing with Labeled Event Structures

In [52], we have developped a complete testing framework for concurrent systems, which included the notions of test suites and test cases. We studied what kind of systems are testable in such a framework, and we have proposed sufficient conditions for obtaining a complete test suite as well as an algorithm to construct a test suite with such properties. However complete test suites are usually infinite in practice. In [44] (and a submitted journal version), we have proposed several testing criteria based on dedicated notions of complete prefixes that selects a manageable test suite together with a coverable criterion that allows to compare them.

6.3. Petri Nets

6.3.1. A Modular Approach for Reusing Formalisms in Verification Tools of Concurrent Systems

Over the past two decades, numerous verification tools have been successfully used for verifying complex concurrent systems, modelled using various formalisms. However, it is still hard to coordinate these tools since they rely on such a large number of formalisms. Having a proper syntactical mechanism to interrelate them through variability would increase the capability of effective integrated formal methods. In [28], we propose a modular approach for defining new formalisms by reusing existing ones and adding new features and/or constraints. Our approach relies on standard XML technologies; their use provides the capability of rapidly and automatically obtaining tools for representing and validating models. It thus enables fast iterations in developing and testing complex formalisms. As a case study, we applied our modular definition approach on families of Petri nets and timed automata.

6.3.2. Computation of summaries using net unfoldings

In [38], we study the following summarization problem: given a parallel composition A = A1 || ... || An of labelled transition systems communicating with the environment through a distinguished component Ai, efficiently compute a summary Si such that E || A and E || Si are trace-equivalent for every environment E. While Si can be computed using elementary automata theory, the resulting algorithm suffers from the state-explosion problem. We present a new, simple but subtle algorithm based on net unfoldings, a partial-order semantics, give experimental results. Our algorithm can also handle divergences and compute weighted summaries with minor modifications.

6.3.3. Complexity Analysis of Continuous Petri Nets

At the end of the eighties, continuous Petri nets were introduced for: (1) alleviating the combinatory explosion triggered by discrete Petri nets and, (2) modelling the behaviour of physical systems whose state is composed of continuous variables. Since then several works have established that the computational complexity of deciding some standard behavioural properties of Petri nets is reduced in this framework. In [39], we first establish the decidability of additional properties like boundedness and reachability set inclusion. We also design new decision procedures for the reachability and lim-reachability problems with a better computational complexity. Finally we provide lower bounds characterising the exact complexity class of the boundedness, the reachability, the deadlock freeness and the liveness problems.

6.3.4. Contextual Merged Processes

In [45], we integrate two compact data structures for representing state spaces of Petri nets: merged processes and contextual prefixes. The resulting data structure, called contextual merged processes (CMP), combines the advantages of the original ones and copes with several important sources of state space explosion: concurrency, sequences of choices, and concurrent read accesses to shared resources. In particular, we demonstrate on a number of benchmarks that CMPs are more compact than either of the original data structures. Moreover, we sketch a polynomial (in the CMP size) encoding into SAT of the model-checking problem for reachability properties.

6.3.5. A Canonical Contraction for Safe Petri Nets

Under maximal semantics, the occurrence of an event a in a concurrent run of an occurrence net may imply the occurrence of other events, not causally related to a, in the same run. In recent works, we have formalized this phenomenon as the *reveals* relation, and used it to obtain a contraction of sets of events called *facets* in the context of occurrence nets. In [36], we extend this idea to propose a canonical contraction of general safe Petri nets into pieces of partial-order behaviour which can be seen as “macro-transitions” since all their events must occur together in maximal semantics. On occurrence nets, our construction coincides with the facets abstraction. Our contraction preserves the maximal semantics in the sense that the maximal processes of the contracted net are in bijection with those of the original net.

6.4. Composition

6.4.1. Specification of Asynchronous Component Systems with Modal I/O-Petri Nets

In collaboration with Professor Rolf Hennicker from LMU and M.H. Møller, a PhD student from Aalborg University, we have studied the asynchronous composition of systems where the internal channels remain observable. In [42], we have modelled such systems by Petri nets enlarged with communication channels, we have defined several channel properties and shown these properties are compositional, and proved their decidability. In TGC 2013 (not yet in HAL), we have extended the previous models with modalities *must* and *may* "à la Larsen" and generalized most of the results in this framework.

6.4.2. Bounding models families for performance evaluation in composite Web services

One challenge of composite Web service architectures is the guarantee of the Quality of Service (QoS). Performance evaluation of these architectures is essential but complex due to synchronizations inside the orchestration of services. In (ADD WHEN IN HAL), we propose methods to automatically derive from the original model a family of bounding models for the composite Web response time. These models allow to find the appropriate trade-off between accuracy of the bounds and the computational complexity. The numerical results show the interest of our approach w.r.t. complexity and accuracy of the response time bounds.

6.5. Stochastic Systems

6.5.1. Simulation-based Verification of HASL (Hybrid Automata Stochastic Logic) Formulas for Stochastic Symmetric Nets

The Hybrid Automata Stochastic Logic (HASL) has been recently defined as a flexible way to express classical performance measures as well as more complex, path-based ones (generically called "HASL formulas"). The considered paths are executions of Generalized Stochastic Petri Nets (GSPN), which are an extension of the basic Petri net formalism to define discrete event stochastic processes. The computation of the HASL formulas for a GSPN model is demanded to the COSMOS tool, that applies simulation techniques to the formula computation. Stochastic Symmetric Nets (SSN) are an high level Petri net formalism, of the colored type, in which tokens can have an identity, and it is well known that colored Petri nets allow one to describe systems in a more compact and parametric form than basic (uncolored) Petri nets. In [27], we propose to extend HASL and COSMOS to support colors, so that performance formulas for SSN can be easily defined and evaluated. This requires a new definition of the logic, to ensure that colors are taken into account in a correct and useful manner, and a significant extension of the COSMOS tool.

6.5.2. Steady-state control problem for Markov decision processes

We address in (ADD CITATION WHEN IN HAL) a control problem for probabilistic models in the setting of Markov decision processes (MDP). We are interested in the steady-state control problem which asks, given an ergodic MDP M and a distribution δ, whether there exists a (history-dependent randomized) policy π ensuring that the steady-state distribution of M under $\imath$ is exactly δ. We first show that stationary randomized policies suffice to achieve a given steady-state distribution. Then we infer that the steady-state control problem is decidable for MDP, and can be represented as a linear program which is solvable in PTIME. This decidability result extends to labeled MDP (LMDP) where the objective is a steady-state distribution on labels carried by the states, and we provide a PSPACE algorithm. We also show that a related steady-state language inclusion problem is decidable in EXPTIME for LMDP. Finally, we prove that if we consider MDP under partial observation (POMDP), the steady-state control problem becomes undecidable.

6.6. Timed Systems

6.6.1. Back in Time Petri Nets

The time progress assumption is at the core of the semantics of real-time formalisms. It is also the major obstacle to the development of partial-order techniques for real-time distributed systems since the events are ordered both by causality and by their occurrence in time. Anyway, extended free choice safe time Petri nets (TPNs) were already identified as a class where partial order semantics behaves well. In [37], we show that, for this class, the time progress assumption can even be dropped (time may go back in case of concurrency), which establishes a nice relation between partial-order semantics and time progress assumption.

6.6.2. Expressiveness of Timed Models

In coopération with Nantes and UPMC, an in-depth study of the expressiveness of time Petri nets was completed [20]. With roughly the same partners, we have extended th ITA (Interrupt Timed Automata) by parametrizing both guards and clock rates while preserving the decidability results (RP 2013, not yet in HAL).

6.7. Weighted Systems

6.7.1. Specification and Verification of Quantitative Properties via Expressions, Logics, and Automata

Alongside boolean properties, automatic verification of *quantitative* properties such as lifespan of an equipment, energy consumption of an application or reliability of a program is gaining importance rapidly. In the thesis [14] and the articles [32], [14] , several weight-enabled formalisms for specification of such properties were examined, including denotational ones such as regular expressions, first-order logic with transitive closure, or temporal logics, as well as more operational ones such as navigating automata, possibly extended with pebbles. A unified framework of graph structures allows to compare these formalisms with respect to expressiveness, using efficient translations from denotational to operational formalisms. Several decidability and complexity results for the algorithmic questions that arise were obtained, depending on the underlying semiring from which weights are chosen, and on the structures (words, trees, ...) considered.

6.8. Dynamic Communicating Systems

6.8.1. Specification and Verification of Dynamic Message-Passing Systems

In [31], we study dynamic communicating automata (DCA), an extension of classical communicating finite-state machines that allows for dynamic creation of processes. The behavior of a DCA can be described as a set of message sequence charts (MSCs). While DCA serve as a model of an implementation, we propose branching high-level MSCs (bHMSCs) on the specification side. Our focus is on the implementability problem: given a bHMSC, can one construct an equivalent DCA? As this problem is undecidable, we introduce the notion of executability, a decidable necessary criterion for implementability. We show that executability of bHMSCs is EXPTIME-complete. We then identify a class of bHMSCs for which executability effectively implies implementability.

6.9. Concurrent Recursive Programs

6.9.1. The Complexity of Model Checking Concurrent Recursive Programs

In [34], we consider the linear-time model checking problem for boolean concurrent programs with recursive procedure calls. While sequential recursive programs are usually modeled as pushdown automata, concurrent recursive programs involve several processes and can be naturally abstracted as pushdown automata with multiple stacks. Their behavior can be understood as words with multiple nesting relations, each relation connecting a procedure call with its corresponding return. To reason about multiply nested words, we consider the class of all temporal logics as defined in the book by Gabbay, Hodkinson, and Reynolds (1994). The unifying feature of these temporal logics is that their modalities are defined in monadic second-order (MSO) logic. In particular, this captures numerous temporal logics over concurrent and/or recursive programs that have been defined so far. Since the general model checking problem is undecidable, we restrict attention to phase bounded executions as proposed by La Torre, Madhusudan, and Parlato (LICS 2007). While the MSO model checking problem in this case is non-elementary, our main result states that the model checking (and satisfiability) problem for all MSO-definable temporal logics is decidable in elementary time. More precisely, it is solvable in (n + 2)-EXPTIME where n is the maximal level of the MSO modalities in the monadic quantifier alternation hierarchy. We complement this result and provide, for each level n, a temporal logic whose model checking problem is n-EXPSPACE-hard.

6.9.2. Model Checking Concurrent Recursive and Communicating Programs via Split-Width

The work described in the following was done by Aiswarya Cyriac in collaboration with Paul Gastin and K. Narayan Kumar, and it is part of Aiswarya Cyriac's PhD thesis, which has recently been defended. It is a generalisation of our CONCUR'12 paper where split-width is introduced to address the decidability of MSO specifications for multi-pushdown systems.

We consider generic systems which incorporate shared-variable communication and communication via channels. We are considering physically distributed machines which communicate via (possibly several) reliable first-in-first-out queues. Each of these machines are capable of running potentially recursive multi-threaded programs. These programs within a machine use shared variable for communication. Such a machine consisting of a set of threads communicating by shared memory can be formally modelled as a multi-pushdown system. Thus we have a network of multi-pushdown systems communicating via FIFO queues. Moreover, these programs may use stacks and queues as data-structures to aid their local computation. We call such a system a system of concurrent processes with data-structures (CPDS).

We introduce and study a new technique called split-width for the under-approximate verification of CPDS. This parameter is based on simple shuffle and merge operations and gives us a divide-conquer-way to prove the bound of languages. When parametrised by a bound on split-width, we obtain decidability for various verification problems. We provide a uniform decision procedure for various verification problems with optimal complexities.

We expose the power of split-width in several ways. We show that our simple algebra is powerful enough to capture any class of CPDS which admits decidability for MSO model checking, and yardstick graph metrics such as tree-width and clique-width. We also show that various restrictions well-studied in the literature for obtaining decidability of reachability for the particular cases of multi-pushdown systems and message passing systems admit a bound on split-width. In fact, we propose generic controllers which subsume many of these cases.

Distributed controller design amounts to designing a controller (which is another CPDS) which, when run sychronously with a system ensures bounded split-width. These controllers are distributed in nature and are independent of the system it is controlling. Thus such a controller respects the privacy of the system (by not reading their states, for instance). Moreover, thanks to split-width such a controlled system offers efficient (in most cases optimal) decision procedures for the verification of the controlled system. We propose a generic approach to define controllable classes of CPDS in terms of quotient graphs, which admit a "suitable" acyclicity restriction. We also give a generic controller for several of the classes definable in this framework.

The controllers we propose are sound and complete for the respective class, meaning that they allow all and only the behaviours of this class. Moreover, our technique for proving the bound on split-width of the controlled systems is also generic and systematic, hence may easily extend to generalisations and other classes as well.

The decidability results for the controllable classes proposed in the thesis are new while they capture, as special cases, several restrictions studied in the literature like bounded phase, bounded scope, poly-forest topology etc.

7. Partnerships and Cooperations

7.1. Regional Initiatives

7.1.1. DIM/LSC TECSTES - 2011-052D

In this DIGITEO project (No. 6024), Hernán Ponce de Léon, Delphine Longuet (ParisSud) and Stefan Haar cooperate on the subject of conformance testing for concurrent systems, using Event Structures. The project started on September 1, 2011 and is scheduled to end on August 31, 2014.

7.1.2. LOCOREP

In the DIGITEO project LoCoReP (No. 2010-043D), Aiswarya Cyriac, Paul Gastin, and Benedikt Bollig worked on temporal logics for the specification and verification of concurrent recursive programs. The project started on September 1, 2010 and ended on August 31, 2013.

7.2. IRT

7.2.1. SystemX

Participants: Simon Theissing, Stefan Haar.

We participate in the project MIC on multi-modal transport systems with in the IRT *System X*, with academic partners UPMC, IFSTTAR and CEA, and several industrial partners including Alstom (project leader), COSMO and Renault.

7.3. National Initiatives

7.3.1. ANR project IMPRO

Participants: Thomas Chatain, Stefan Haar, Serge Haddad.

The Project ANR **ImpRo** ANR-2010-BLAN-0317 involves *IRCCyN* (Nantes), *IRISA* (Rennes), *LIP6*(Paris), *LSV* (Cachan), *LIAFA* (Paris) and *LIF* (Marseille). It addresses issues related to the practical implementation of formal models for the design of communication-enabled systems: such models abstract away from many complex features or limitations of the execution environment. The modeling of *time*, in particular, is usually idealized, with infinitely precise clocks, instantaneous tests or mode communications, etc. Our objective is thus to study to what extent the practical implementation of these models preserves their good properties. We aim at a generic mathematical framework to reason about and measure implementability, and then study the possibility to integrate implementability constraints in the models. A particular focus is on the combination of several sources of perturbation such as resource allocation, the distributed architecture of applications, etc. We also study implementability through control and diagnosis techniques, and apply the developed methods to a case study based on the AUTOSAR architecture, a standard in the automotive industry.

7.4. European Initiatives

7.4.1. FP7 Projects

7.4.1.1. Hycon2

Type: COOPERATION

Defi: Engineering of Networked Monitoring and Control Systems

Instrument: Network of Excellence

Objectif: Engineering of Networked Monitoring and Control systems

Duration: September 2010 - August 2014

Coordinator: CNRS

Partner: ETH Zürich, TU Berlin, TU Delft and many others.

Inria contact: C. Canudas de Wit

Abstract: Hycon2 aims at stimulating and establishing a long-term integration in the strategic field of control of complex, large-scale, and networked dynamical systems. It focuses in particular on the domains of ground and aerospace transportation, electrical power networks, process industries, and biological and medical systems.

7.4.1.2. Univerself: realizing autonomics for Future Networks

Type: COOPERATION

Defi: The Network of the Future

Instrument: Integrated Project

Objectif: The Network of the Future

Duration: September 2010 - August 2013

Coordinator: Alcatel Lucent (France)

Partner: UTwente, AL Ireland, AL Germany, VTT (Finland), U. of Piraeus, FT, Telecom Italia, NU of Athens, Fraunhofer, Interdic. Institue for Broadband Technology, Telefonica, Thales, Nec Europe, U. of Surrey, UCL, IBBT (Belgium)

Inria contact: E. Fabre

Abstract: UniverSelf unites 17 partners with the aim of overcoming the growing management complexity of future networking systems, and to reduce the barriers that complexity and ossification pose to further growth.

7.5. International Initiatives

7.5.1. Inria International Partners

7.5.1.1. Informal International Partners

1. The CMI (Chennai Mathematical Institute) is a long-standing partner of our team. The project *Île de France/Inde* in the *ARCUS* program from 2008 to 2011 has allowed several exchange visits between Cachan and Chennai, organizations of ACTS workshops with french and indian researchers in Chennai, internships in Cachan, and two theses in *co-tutelle* (Akshay Sundararaman, defended in 2010) and Aiswarya Cyriac (thesis in progress).

   Currently, Paul Gastin is co-head (with Madhavan Mukund) of the CNRS International Associated Laboratory (LIA) INFORMEL (INdo-French FORmal Methods Lab, http://projects.lsv.ens-cachan.fr/informel/) . This LIA was created in January 2012 by an agreement between CNRS, ENS Cachan, University Bordeaux 1 on the french side and the Chennai Mathematical Institute, the Institute of Mathematical Sciences of Chennai, and the Indian Institute of Science of Bangalore on the Indian side.

2. We have been exchanging visits for several years between *MExICo* and the DISCO team (Lucia Pomello and Luca Bernardinello) at University Milano-Bicocca, Italy.
3. Exchanges are frequent with Rolf Hennicker from LMU and Javier Esparza at TUM, both in Munich, Germany.
4. With the computer science and electrical engineering departments at Newcastle University, UK

7.5.2. Participation In Other International Programs (non-Inria)

Benedikt Bollig, Aiswarya Cyriac, and Benjamin Monmege are participating in LeMon, a joint Procope project with LIAFA, (Paris) and the University of Lübeck, supported by EGIDE/DAAD. The aim of the project is to develop techniques for the inference of systems that deal with infinite data domains.

7.6. International Research Visitors

7.6.1. Visits of International Scientists

- Monika Heiner, Professor at University of Cottbus/Germany, visited *MExICo*from September 15 through October 15, 2013.
- Estibaliz Fraca, PhD student from Zaragossa, visited *MExICo*from november 2012 trough February 2013.
- From 7 to 19 January 2013, Paul Gastin and Aiswarya Cyriac (LSV) visit K. Narayan Kumar and Madhavan Mukund at CMI Chennai. They studied verification problems for concurrent and recursive multi-threaded programs.
- 13 May to 1 June 2013: Madhavan Mukund (CMI) visits LSV, IRISA.
- 8 to 29 June K. Narayan Kumar (CMI) visits LSV, LaBRI. The study verification problems for concurrent and recursive multi-threaded programs was pursued.
- 16 June to 30 June 2013: Saivasan Prakash (CMI) visits LSV and LIAFA. Discussions with Ahmed Bouajjani on Verification of networks of Communicating Recursive Processes. Joint work with M.F.Atig (Uppsala) and manuscript based on this work is under preparation.
- 25 May to 20 July 2013: Bharat Adsul (IIT Bombay) visits LSV and LaBRI to work on cascade products of asynchronous automata.

7.6.1.1. Internships

Gonzalo Amadio

Subject: Diagnosis of Stochastic Systems

Date: from Apr 2013 until Jul 2013

Institution: Universidad National de Rosario (Argentina)

Siddharth Krishna

Subject: Multiple Context Free Grammars

Date: from May 19, 2013 until June 15, 2013

Institution: Chennai Mathematical Institute, India

7.6.2. Visits to International Teams

- Thomas Chatain visited
  - Lucia Pomello and Luca Bernardinello at University of Milano-Bicocca for one week in February 2013,
  - Humboldt Universität Berlin for the KOSMOS-Workshop (November 28-30, 2013)

- 4 to 19 December 2013: Paul Gastin and Aiswarya Cyriac (LSV) visit CMI. With K. Narayan Kumar, they completed the study of verification problems via split-width for concurrent recursive multi-threaded programs (a paper is in preparation). With Madhavan Mukund, they started working on statistical analysis of asynchronous systems.
- Stefan Haar visited
    1. Technische Universitä Berlin in for five days in March 2013 and three days in November 2013 for seminar talks and technical cooperation,
    2. Humboldt Universität Berlin for the KOSMOS-Workshop (Nov. 28-30)
    3. University of Newcastle (UK) June 10-12 and Sep.16-20,
    4. Bucarest Polytechnic (RO) May 29 to June 1, giving a course on verification within the *CAN'TI* summer school, and
    5. University of Cordoba (Argentina) as an invited professor, from Oct 27 to Nov 1.
- Serge Haddad
- Hernán Ponce de Léon visited University of Cordoba (Argentina) for two weeks in October/November.
- César Rodríguez visited Victor Khomenko at the University of Newcastle for one week in May.
- Stefan Schwoon visited the group of Javier Esparza at the Technical University of Munich for two weeks in February.

8. Dissemination

8.1. Scientific Animation

8.1.1. Benedikt Bollig

was on the program committee of *YR-CONCUR 2013*. He also was a member of the commission scientifique Inria Saclay.

8.1.2. Thomas Chatain

was on the program committee *ACSD 2013* and of *FORMATS 2013*. He also participated in the organization of the latter.

8.1.3. Paul Gastin

is co-head (with Madhavan Mukund) of the new International Associated Laboratory (LIA) INFORMEL (INdo-French FORmal Methods Lab). This LIA was created in January 2012 by an agreement between CNRS, ENS Cachan, University Bordeaux 1 on the french side and the Chennai Mathematical Institute, the Institute of Mathematical Sciences of Chennai, and the Indian Institute of Science of Bangalore on the Indian side.

He is the head of the computer science department of ENS Cachan.

Paul Gastin is an associate editor of the Journal of Automata, Languages and Combinatorics.

He is on the Advisory Board of the EATCS-Springer book series

- Monographs in Theoretical Computer Science,
- Texts in Theoretical Computer Science.

8.1.4. Stefan Haar

is an associated editor for the journal *Discrete Event Dynamic Systems: Theory and Application*, and was on the program committee of *PNSE 2013*. At the Inria Center Saclay, Stefan was the correspondent for international relations until September 2013 (his successor is Benjamin Smith), and has since become the correspondent for European Partnerships; he continues as a member of the GTRI (working group on international relations) of Inria's *COST*. In 2013 he also joined the DIGITEO program committee.

8.1.5. Serge Haddad

has been a member of the steering committee of the international conference Applications and Theory of Petri Nets (ATPN) since 2001. In 2013, he was a member of the following program committees of international conferences:

- *7th International Workshop on Verification and Evaluation of Computer and Communication systems*, Florence, Italy;
- *21st International Conference on Real Time Networks and Systems (RTNS 2013)*, Sophia-Antipolis, France;
- PC Co-Chaire of *SMC*, associated workshop at *Run-Time Verification 2013*, Rennes, France;
- *33rd International Conference on Application and Theory of Petri Nets (ATPN)*, Milan, Italy;
- *Petri Nets in Software Engineering (PNSE)*, associated Workshop at ATPN 2013.

Serge Haddad served on the program committees of the following national conferences:

- *9ème Colloque Francophone sur la Modélisation des Systèmes Réactifs (MSR) 2013*, Rennes
- *Ecole d'été Temps Réel (ETR) 2013*, Toulouse.

8.1.6. Claudine Picaronny

is a member of the Program committee of the *SIMUL* conference. She is a *Maître de Conférence* at ENS Cachan, and in charge of the Master M2 FESUP and the *préparation à l'agrégation en mathématiques* at ENSC. Moreover, Claudine Picaronny is a member of the jury for the second ENSC entrance examination in mathematics, coordinator of the entrance examinations MP and PC of the groupe E3A, and of the jury of the national olympics in mathematics and computer science.

8.1.7. Stefan Schwoon

was on the program committee of *SPIN 2013*.

8.2. Teaching - Supervision - Juries

8.2.1. Teaching

Here we present the teaching activities of *researchers*; note that in addition to the classes here, five team members (Th. Chatain, P. Gastin, S. Haddad, C. Picaronny and S. Schwoon) are full-time professors (*professeurs* ou *maître de conférences*) of ENS Cachan and fulfill their teaching obligations there.

Master : Stefan Haar, *Analyse structurelle de Réseaux de Petri*, 15 TDEQ, M2 SAR, UPMC

Agrégation: Stefan Haar, Algorithmique, ca. 20 TDEQ, *Préparation Agrégation option informatiqe*, ENS Cachan

Thomas Chatain and Stefan Schwoon gave lectures on unfoldings of Petri nets at the Petri Nets 2013 conference.

8.2.2. Supervision

8.2.2.1. HdR

Thomas Chatain, *Concurrency in Real-Time Distributed Systems, from Unfoldings to Implementability*, ENS Cachan, defended Dec. 13, 2013; *garant*: Stefan Haar

HdR : Stefan Schwoon, *Efficient verification of sequential and concurrent systems*, ENS Cachan, defended Dec. 6, 2013; *garant*: Stefan Haar

8.2.2.2. PhD

Benjamin Monmege, *Specification and Verification of Quantitative Properties: Expressions, Logics, and Automata* , ENS Cachan, defended Oct 24, 2013; Supervisors: Paul Gastin and Benedikt Bollig.

César Rodrîguez, *Verification Based on Unfoldings of Petri Nets with Read Arcs*, ENS Cachan, defended Dec 12, 2013; Supervisor: Stefan Schwoon

8.2.2.3. PhD in progress

Benoît Barbot, *Rare event handling in statistical model checking*, since September 2011; Supervisors: Serge Haddad and Claudine Picaronny.

Aiswarya Cyriac, *Verification of Communicating Recursive Programs via Split-width*, since September 2010; Supervisors: Paul Gastin and Benedikt Bollig.

Hernán Ponce de Léon, Testing concurrent systems using event structures, ENS Cachan, since September 2011; Supervisors: Stefan Haar and Delphine Longuet

Simon Theissing, Supervision of Multi-Modal Transport Systems, since September 2013, ENS Cachan/Inria/IRT SystemX project MIC; Supervisor: Stefan Haar

Salim Perchy, *D-Spaces*, Ecole Polytechnique, since November 2013; main supervisor : Supervisors: Frank D. Valencia (COMETE Team) and Stefan Haar

8.2.3. Juries

8.2.3.1. Paul Gastin

was a reviewer

- of the PhD thesis of Amélie Stainer, Rennes , defended on Nov. 25;
- and of the HdR thesis of Loic Helouet, Rennes, defended on May 17.

As supervisor of Benjamin Monmege, he also served as *rapporteur* on the jury of his defense, on Oct 24.

8.2.3.2. Stefan Haar

was a reviewer of the PhD theses of

- Elisabeta Mangioni, Univ. Milan-Bicocca, Italy,
- Florent Avellaneda, Univ. Aix-Marseille, defended Dec 10, and
- Sébastien Chédor, Université Rennes 1, defended on January 7, 2014.

Except for the defence of E. Mangioni, he also participated in the respective juries. Moreover, he was jury president for the defence of Aurore Junier, ENS Cachan, defended on December 16 in Rennes, and *examinateur* in the PhD jury of César Rodríguez, defended December 12. He also was a member of the HdR juries (as *garant*) of Stefan Schwoon (December 6) and Thomas Chatain (December 13), both at ENS Cachan.

8.2.3.3. Serge Haddad

was a reviewer for the PhD thesis of Henri Debrat in Nancy and member of the Jury on December 6, 2013. He also served on the jury for the HdR of Kais Klai at Université Paris 13/ Nord on December 9, 2013.

8.2.3.4. Stefan Schwoon

was *examinateur* for the PhD thesis of Yan Zhang at Université Paris 6. [18], [22], [23], [17], [19], [21]

8.3. Popularization

Benedikt Bollig gave an invited talk at the German Workshop “Automaten und Logik”, held on September 25, 2013, in Ilmenau.

Paul Gastin has given a talk on *"Automates: Applications et Algorithmique"* during the *Rencontres Algorithmiques et Programmation*, addressing secondary teachers of the *classes préparatoires* level, at CIRM Marseille, 6 to 10 May. He also gave an invited talk on *Evaluation of Weighted Specifications over Nested Words* for the opening of the research training group *Quantitative Logics and Automata (QuantLa)*, Leipzig, on April 30.

9. Bibliography

Major publications by the team in recent years

[1] S. AKSHAY, B. BOLLIG, P. GASTIN, M. MUKUND, K. NARAYAN KUMAR. *Distributed Timed Automata with Independently Evolving Clocks*, in "Proceedings of the 19th International Conference on Concurrency Theory (CONCUR'08)", Toronto, Canada, F. VAN BREUGEL, M. CHECHIK (editors), Lecture Notes in Computer Science, Springer, August 2008, vol. 5201, pp. 82-97 [*DOI :* 10.1007/978-3-540-85361-9_10], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/ABGMN-concur08.pdf

[2] S. BALAGUER, TH. CHATAIN, S. HAAR. *A Concurrency-Preserving Translation from Time Petri Nets to Networks of Timed Automata*, in "Formal Methods in System Design", June 2012, vol. 40, n° 3, pp. 330-355 [*DOI :* 10.1007/s10703-012-0146-4], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCH-fmsd12.pdf

[3] P. BALDAN, TH. CHATAIN, S. HAAR, B. KÖNIG. *Unfolding-based Diagnosis of Systems with an Evolving Topology*, in "Information and Computation", October 2010, vol. 208, n° 10, pp. 1169-1192, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHK-icomp10.pdf

[4] B. BOLLIG, P. GASTIN, B. MONMEGE, M. ZEITOUN. *Pebble weighted automata and transitive closure logics*, in "Proceedings of the 37th International Colloquium on Automata, Languages and Programming (ICALP'10) – Part II", Bordeaux, France, S. ABRAMSKY, F. MEYER AUF DER HEIDE, P. SPIRAKIS (editors), Lecture Notes in Computer Science, Springer, July 2010, vol. 6199, pp. 587-598, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BGMZ-icalp10.pdf

[5] B. BOLLIG, D. KUSKE, I. MEINECKE. *Propositional Dynamic Logic for Message-Passing Systems*, in "Logical Methods in Computer Science", September 2010, vol. 6, n° 3:16, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BKM-lmcs10.pdf

[6] P. BOUYER, S. HADDAD, P.-A. REYNIER. *Timed Petri Nets and Timed Automata: On the Discriminating Power of Zeno Sequences*, in "Information and Computation", January 2008, vol. 206, n° 1, pp. 73-107 [*DOI :* 10.1016/j.ic.2007.10.004], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BHR-ic07.pdf

[7] B. BÉRARD, F. CASSEZ, S. HADDAD, D. LIME, O. H. ROUX. *When are Timed Automata Weakly Timed Bisimilar to Time Petri Nets ?*, in "Theoretical Computer Science", September 2008, vol. 403, n° 2-3, pp. 202-220 [*DOI :* 10.1016/j.tcs.2008.03.030], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHLR-tcs08.pdf

[8] TH. CHATAIN, P. GASTIN, N. SZNAJDER. *Natural Specifications Yield Decidability for Distributed Synthesis of Asynchronous Systems*, in "Proceedings of the 35th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM'09)", Špindlerův Mlýn, Czech Republic, M. NIELSEN, A. KUČERA, P. BRO MILTERSEN, C. PALAMIDESSI, P. TŮMA, F. VALENCIA (editors), Lecture Notes in Computer Science, Springer, January 2009, vol. 5404, pp. 141-152 [*DOI :* 10.1007/978-3-540-95891-8_16], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/CGS-sofsem09.pdf

[9] P. GASTIN, D. KUSKE. *Uniform satisfiability problem for local temporal logics over Mazurkiewicz traces*, in "Information and Computation", July 2010, vol. 208, n° 7, pp. 797-816, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/GK-icomp10.pdf

[10] S. HAAR. *Types of Asynchronous Diagnosability and the Reveals-Relation in Occurrence Nets*, in "IEEE Transactions on Automatic Control", October 2010, vol. 55, n° 10, pp. 2310-2320, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/haar-tac10.pdf

[11] L. RECALDE, S. HADDAD, M. SILVA. *Continuous Petri Nets: Expressive Power and Decidability Issues*, in "International Journal of Foundations of Computer Science", April 2010, vol. 21, n° 2, pp. 235-256

[12] C. RODRÍGUEZ, S. SCHWOON, P. BALDAN. *Efficient contextual unfolding*, in "Proceedings of the 22nd International Conference on Concurrency Theory (CONCUR'11)", Aachen, Germany, J.-P. KATOEN, B. KÖNIG (editors), Lecture Notes in Computer Science, Springer, September 2011, vol. 6901, pp. 342-357 [*DOI :* 10.1007/978-3-642-23217-6_23], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/RSB-concur11.pdf

Publications of the year

Doctoral Dissertations and Habilitation Theses

[13] T. CHATAIN. , *Concurrency in Real-Time Distributed Systems, from Unfoldings to Implementability*, École normale supérieure de Cachan - ENS Cachan, December 2013, Habilitation à Diriger des Recherches, http://hal.inria.fr/tel-00926306

[14] B. MONMEGE. , *Spécification et vérification de propriétés quantitatives : expressions, logiques et automates*, École normale supérieure de Cachan - ENS Cachan, October 2013, http://hal.inria.fr/tel-00908990

[15] C. RODRÍGUEZ. , *Verification Based on Unfoldings of Petri Nets with Read Arcs*, École normale supérieure de Cachan - ENS Cachan, December 2013, http://hal.inria.fr/tel-00927064

[16] S. SCHWOON. , *Efficient verification of sequential and concurrent systems*, École normale supérieure de Cachan - ENS Cachan, December 2013, Habilitation à Diriger des Recherches, http://hal.inria.fr/tel-00927066

Articles in International Peer-Reviewed Journals

[17] S. AKSHAY, B. BOLLIG, P. GASTIN. *Event-clock Message Passing Automata: A logical characterization and an emptiness checking algorithm*, in "Formal Methods in System Design", 2013, vol. 42, n° 3, pp. 262-300, http://hal.inria.fr/hal-00925745

[18] S. BALAGUER, T. CHATAIN. *Avoiding Shared Clocks in Networks of Timed Automata*, in "Logical Methods in Computer Science", 2013, vol. 9, n° 4:13, pp. 1-26, http://hal.inria.fr/hal-00925723

[19] S. BALAGUER, T. CHATAIN, S. HAAR. *Building Occurrence Nets from Reveals Relations*, in "Fundamenta Informaticae", 2013, vol. 123, n° 3, pp. 245-272, http://hal.inria.fr/hal-00925754

[20] B. BERARD, F. CASSEZ, S. HADDAD, D. LIME, O. H. ROUX. *The Expressive Power of Time Petri Nets*, in "Theoretical Computer Science", 2013, vol. 474, pp. 1-20, http://hal.inria.fr/hal-00925765

[21] R. BONNET, A. FINKEL, S. HADDAD, F. ROSA-VELARDO. *Ordinal Theory for Expressiveness of Well-Structured Transition Systems*, in "Information and Computation", 2013, vol. 224, pp. 1-22, http://hal.inria.fr/hal-00925762

[22] P. GASTIN, N. SZNAJDER. *Fair Synthesis for Asynchronous Distributed Systems*, in "ACM Transactions on Computational Logic", 2013, vol. 14, n^{o} 2:9, pp. 1-31, http://hal.inria.fr/hal-00925735

[23] S. HAAR, C. KERN, S. SCHWOON. *Computing the Reveals Relation in Occurrence Nets*, in "Theoretical Computer Science", 2013, vol. 493, pp. 66-79, http://hal.inria.fr/hal-00925742

[24] S. HADDAD, J. MAIRESSE, H.-T. NGUYEN. *Synthesis and Analysis of Product-form Petri Nets*, in "Fundamenta Informaticae", 2013, vol. 122, n^{o} 1-2, pp. 147-172, http://hal.inria.fr/hal-00925774

[25] S. HADDAD, L. MOKDAD, S. YOUCEF. *Bounding models families for performance evaluation in composite Web services*, in "Journal of Computational Science", March 2013, vol. 4, n^{o} 4, pp. 232-241 [*DOI :* 10.1016/J.JOCS.2011.11.003], http://hal.inria.fr/hal-00920332

International Conferences with Proceedings

[26] S. AKSHAY, N. BERTRAND, S. HADDAD, L. HELOUET. *The steady-state control problem for Markov decision processes*, in "Qest 2013", Buenos Aires, Argentina, K. R. JOSHI, M. SIEGLE, M. STOELINGA, P. R. D'ARGENIO (editors), LNCS, Springer, September 2013, vol. 8054, pp. 290-304, http://hal.inria.fr/hal-00879355

[27] E. G. AMPARORE, B. BARBOT, M. BECCUTI, S. DONATELLI, G. FRANCESCHINIS. *Simulation-based Verification of Hybrid Automata Stochastic Logic Formulas for Stochastic Symmetric Nets*, in "1st ACM SIGSIM Conference on Principles of Advanced Discrete Simulation (PADS'13)", Canada, ACM Press, 2013, pp. 253-264 [*DOI :* 10.1145/2486092.2486124], http://hal.inria.fr/hal-00926210

[28] É. ANDRÉ, B. BARBOT, C. DÉMOULINS, L. MESSAN HILLAH, F. HULIN-HUBARD, F. KORDON, A. LINARD, L. PETRUCCI. *A Modular Approach for Reusing Formalisms in Verification Tools of Concurrent Systems*, in "15th International Conference on Formal Engineering Methods (ICFEM'13)", New Zealand, Lecture Notes in Computer Science, Springer, 2013, pp. 199-214, http://hal.inria.fr/hal-00926126

[29] É. ANDRÉ, L. HILLAH, F. HULIN-HUBARD, F. KORDON, Y. LEMBACHAR, A. LINARD, L. PETRUCCI. *CosyVerif: An Open Source Extensible Verification Environment*, in "18th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'13)", Singapore, IEEE Computer Society Press, 2013, pp. 33-36, http://hal.inria.fr/hal-00926165

[30] N. BERTRAND, E. FABRE, S. HAAR, S. HADDAD, L. HELOUET. *Active diagnosis for probabilistic systems*, in "FOSSACS'2014", Grenoble, France, A. MUSCHOLL (editor), Springer, April 2014, http://hal.inria.fr/hal-00930919

[31] B. BOLLIG, A. CYRIAC, L. HELOUET, A. KARA, T. SCHWENTICK. *Dynamic Communicating Automata and Branching High-Level MSCs*, in "LATA 2013", bilbao, Spain, A. HORIA DEDIU, C. MARTÍN-VIDE, B. TRUTHE (editors), LNCS, Springer, April 2013, vol. 7810, pp. 177-189, http://hal.inria.fr/hal-00879353

[32] B. BOLLIG, P. GASTIN, B. MONMEGE. *Weighted Specifications over Nested Words*, in "Foundations of Software Science and Computation Structures (FoSSaCS'13)", Rome, Italy, Lecture Notes in Computer Science, March 2013, vol. 7794, pp. 385-400 [*DOI :* 10.1007/978-3-642-37075-5_25], http://hal.inria.fr/hal-00909035

[33] B. BOLLIG, P. HABERMEHL, M. LEUCKER, B. MONMEGE. *A Fresh Approach to Learning Register Automata*, in "Developments in Language Theory", France, Lecture Notes in Computer Science, June 2013, vol. 7907, pp. 118-130 [*DOI :* 10.1007/978-3-642-38771-5_12], http://hal.inria.fr/hal-00908998

[34] B. BOLLIG, D. KUSKE, R. MENNICKE. *The Complexity of Model Checking Multi-Stack Systems*, in "28th Annual IEEE Symposium on Logic in Computer Science (LICS'13)", United States, IEEE Computer Society Press, 2013, pp. 163-170 [*DOI :* 10.1109/LICS.2013.22], http://hal.inria.fr/hal-00926182

[35] B. BÉRARD, S. HADDAD, A. JOVANOVIC, D. LIME. *Parametric Interrupt Timed Automata*, in "7th Workshop on Reachability Problems in Computational Models (RP'13)", Sweden, 2013, pp. 59-69, http://hal.inria.fr/hal-00936961

[36] T. CHATAIN, S. HAAR. *A Canonical Contraction for Safe Petri Nets*, in "7th International Workshop on Petri Nets and Software Engineering (PNSE'13)", Germany, D. M. ;. H. RÖLKE (editor), CEUR Workshop Proceedings, 2013, vol. 969, pp. 25-39, http://hal.inria.fr/hal-00926190

[37] T. CHATAIN, C. JARD. *Back in Time Petri Nets*, in "FORMATS'13", Argentina, Lecture Notes in Computer Science, Springer, 2013, vol. 8053, pp. 91-105, http://hal.inria.fr/hal-00925467

[38] J. ESPARZA, L. JEZEQUEL, S. SCHWOON. *Computation of summaries using net unfoldings*, in "33rd Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'13)", India, Leibniz International Proceedings in Informatics, 2013, vol. 24, pp. 225-236, http://hal.inria.fr/hal-00925457

[39] *Best Paper*
E. FRACA, S. HADDAD. *Complexity Analysis of Continuous Petri Nets*, in "34th International Conference on Applications and Theory of Petri Nets (ICATPN'13)", Italy, J.-M. COLOM, J. DESEL (editors), Lecture Notes in Computer Science, Springer, 2013, vol. 7927, pp. 170-189 [*DOI :* 10.1007/978-3-642-38697-8_10], http://hal.inria.fr/hal-00926196.

[40] S. HAAR, S. HADDAD, T. MELLITI, S. SCHWOON. *Optimal Constructions for Active Diagnosis*, in "33rd Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'13)", Guwahati, India, Leibniz International Proceedings in Informatics, December 2013, pp. 527-539, http://hal.inria.fr/hal-00926098

[41] S. HAAR, C. RODRÍGUEZ, S. SCHWOON. *Reveal Your Faults: It's Only Fair!*, in "13th International Conference on Application of Concurrency to System Design (ACSD'13)", Spain, IEEE Computer Society Press, 2013, pp. 120-129, http://hal.inria.fr/hal-00926171

[42] S. HADDAD, R. HENNICKER, M. H. MØLLER. *Channel Properties of Asynchronously Composed Petri Nets*, in "34th International Conference on Applications and Theory of Petri Nets (ICATPN'13)", Italy, Lecture Notes in Computer Science, Springer, 2013, vol. 7927, pp. 369-388 [*DOI :* 10.1007/978-3-642-38697-8_20], http://hal.inria.fr/hal-00926200

[43] H. PONCE DE LEÓN, G. BONIGO, L. BRANDÁN BRIONES. *Distributed Analysis of Diagnosability in Concurrent Systems*, in "24th International Workshop on Principles of Diagnosis (DX'13)", Israel, 2013, pp. 142-147, http://hal.inria.fr/hal-00926145

[44] H. PONCE DE LEÓN, S. HAAR, D. LONGUET. *Unfolding-based Test Selection for Concurrent Conformance*, in "25th IFIP International Conference on Testing Software and Systems (ICTSS'13)", Turkey, Lecture Notes in Computer Science, Springer, 2013, vol. 8254, pp. 98-113, http://hal.inria.fr/hal-00926103

[45] C. RODRÍGUEZ, S. SCHWOON, V. KHOMENKO. *Contextual Merged Processes*, in "34th International Conference on Applications and Theory of Petri Nets (ICATPN'13)", Italy, Lecture Notes in Computer Science, Springer, 2013, vol. 7927, pp. 29-48 [*DOI :* 10.1007/978-3-642-38697-8_3], http://hal.inria.fr/hal-00926202

[46] C. RODRÍGUEZ, S. SCHWOON. *An Improved Construction of Petri Net Unfoldings*, in "1st French-Singaporean Workshop on Formal Methods and Applications (FSFMA'13)", Singapore, Open Access Series in Informatics, Leibniz-Zentrum für Informatik, 2013, pp. 47-52, http://hal.inria.fr/hal-00926177

[47] C. RODRÍGUEZ, S. SCHWOON. *Cunf: A Tool for Unfolding and Verifying Petri Nets with Read arcs*, in "11th International Symposium on Automated Technology for Verification and Analysis (ATVA'13)", Viet Nam, Lecture Notes in Computer Science, Springer, 2013, vol. 8172, pp. 492-495, http://hal.inria.fr/hal-00925448

Scientific Books (or Scientific Book chapters)

[48] S. HAAR, E. FABRE. *Diagnosis with Petri Net Unfoldings*, in "Control of Discrete-Event Systems - Automata and Petri Net Perspectives", Lecture Notes in Control and Information Sciences, Springer, 2013, pp. 301-318, http://hal.inria.fr/hal-00926087

[49] S. HAAR, T. MASOPUST. *Languages, Decidability, and Complexity*, in "Control of Discrete-Event Systems - Automata and Petri Net Perspectives", Lecture Notes in Control and Information Sciences, Springer, 2013, pp. 23-43, http://hal.inria.fr/hal-00926094

Other Publications

[50] B. BOLLIG, P. HABERMEHL, M. LEUCKER, B. MONMEGE. , *A Robust Class of Data Languages and an Application to Learning*, 2013, http://hal.inria.fr/hal-00920945

[51] F. KORDON, A. LINARD, M. BECUTTI, D. BUCHS, L. FRONC, F. HULIN-HUBARD, F. LEGOND-AUBRY, N. LOHMANN, A. MARECHAL, E. PAVIOT-ADET, F. POMMEREAU, C. RODRÍGUEZ, C. ROHR, Y. THIERRY-MIEG, H. WIMMEL, C. WOLF. , *Web Report on the Model Checking Contest @ Petri Net 2013*, 2013, http://mcc.lip6.fr, http://hal.inria.fr/hal-00926989

[52] H. PONCE DE LEÓN, S. HAAR, D. LONGUET. , *Model Based Testing for Concurrent Systems with Labeled Event Structures*, March 2013, Submitted to a journal, http://hal.inria.fr/hal-00796006

References in notes

[53] W. KUICH, H. VOGLER, M. DROSTE (editors). , *Handbook of Weighted Automata*, EATCS Monographs in Theoretical Computer Science, Springer, 2009

[54] S. ABBES, A. BENVENISTE, S. HAAR. *A Petri net model for distributed estimation*, in "Proc. MTNS 2004, Sixteenth International Symposium on Mathematical Theory of Networks and Systems, Louvain (Belgium), ISBN 90-5682-517-8", 2004

[55] S. AKSHAY, B. BOLLIG, P. GASTIN, M. MUKUND, K. NARAYAN KUMAR. *Distributed Timed Automata with Independently Evolving Clocks*, in "Proceedings of the 19th International Conference on Concurrency Theory (CONCUR'08)", Toronto, Canada, F. VAN BREUGEL, M. CHECHIK (editors), Lecture Notes in Computer Science, Springer, August 2008, vol. 5201, pp. 82-97, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/ABGMN-concur08.pdf

[56] R. ALUR, K. ETESSAMI, M. YANNAKAKIS. *Realizability and Verification of MSC Graphs*, in "Theor. Comput. Sci.", 2005, vol. 331, n^{o} 1, pp. 97–114

[57] P. BALDAN, A. CORRADINI, B. KÖNIG, S. SCHWOON. *McMillan's complete prefix for contextual nets*, in "Transactions on Petri Nets and Other Models of Concurrency", November 2008, vol. 1, pp. 199–220, Volume 5100 of Lecture Notes in Computer Science

[58] P. BALDAN, S. HAAR, B. KOENIG. *Distributed Unfolding of Petri Nets*, in "Proc.FOSSACS 2006", LNCS, Springer, 2006, vol. 3921, pp. 126-141, Extended version: Technical Report CS-2006-1. Department of Computer Science, University Ca' Foscari of Venice

[59] F. BASKETT, K. M. CHANDY, R. R. MUNTZ, F. G. PALACIOS. *Open, Closed, and Mixed Networks of Queues with Different Classes of Customers*, in "J. ACM", April 1975, vol. 22, pp. 248–260, http://doi.acm.org/10.1145/321879.321887

[60] A. BENVENISTE, É. FABRE, S. HAAR. *Markov Nets: Probabilistic Models for distributed and concurrent Systems*, in "IEEE Transactions on Automatic Control", 2003, vol. 48 (11), pp. 1936-1950, Extended version: IRISA Research Report 1538

[61] P. BHATEJA, P. GASTIN, M. MUKUND, K. NARAYAN KUMAR. *Local testing of message sequence charts is difficult*, in "Proceedings of the 16th International Symposium on Fundamentals of Computation Theory (FCT'07)", Budapest, Hungary, E. CSUHAJ-VARJÚ, Z. ÉSIK (editors), Lecture Notes in Computer Science, Springer, August 2007, vol. 4639, pp. 76-87 [*DOI :* 10.1007/978-3-540-74240-1_8], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BGMN-fct07.pdf

[62] G. V. BOCHMANN, S. HAAR, C. JARD, G.-V. JOURDAN. *Testing Systems Specified as Partial Order Input/Output Automata*, in "Proc. TESTCOM/Fates 08, 20th IFIP International Conference on Testing of Communicating Systems and 8th International Workshop on Formal Approaches to Testing of Software", LNCS, Springer, 2008, vol. 5047, pp. 169-183

[63] B. BOLLIG, P. GASTIN. *Weighted versus Probabilistic Logics*, in "Proceedings of the 13th International Conference on Developments in Language Theory (DLT'09)", Stuttgart, Germany, V. DIEKERT, D. NOWOTKA (editors), Lecture Notes in Computer Science, Springer, June-July 2009, vol. 5583, pp. 18-38 [*DOI :* 10.1007/978-3-642-02737-6_2], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BG-dlt09.pdf

[64] A. BOUILLARD, S. HAAR, S. ROSARIO. *Critical paths in the Partial Order Unfolding of a Stochastic Petri Net*, in "Proceedings of the 7th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS'09)", Budapest, Hungary, J. OUAKNINE, F. VAANDRAGER (editors), Lecture Notes in Computer Science, Springer, September 2009, vol. 5813, pp. 43-57 [*DOI :* 10.1007/978-3-642-04368-0_6], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BHR-formats09.pdf

[65] P. BOUYER, S. HADDAD, P.-A. REYNIER. *Timed Unfoldings for Networks of Timed Automata*, in "Proceedings of the 4th International Symposium on Automated Technology for Verification and Analysis (ATVA'06)",

Beijing, ROC, S. GRAF, W. ZHANG (editors), Lecture Notes in Computer Science, Springer, October 2006, vol. 4218, pp. 292-306, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BHR-atva06.pdf

[66] G. CHIOLA, C. DUTHEILLET, G. FRANCESCHINIS, S. HADDAD. *Stochastic Well-Formed Colored Nets and Symmetric Modeling Applications*, in "IEEE Transactions on Computers", November 1993, vol. 42, n^{o} 11, pp. 1343-1360, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PS/CDFH-toc93.ps

[67] A. CHURCH. *Logic, arithmetics, and automata*, in "Proc. of Int. Congr. of Mathematicians", 1962, pp. 23–35

[68] R. DEBOUK, D. TENEKETZIS. *Coordinated decentralized protocols for failure diagnosis of discrete-event systems*, in "Journal of Discrete Event Dynamical Systems: Theory and Application", 2000, vol. 10, pp. 33–86

[69] D. EL HOG-BENZINA, S. HADDAD, R. HENNICKER. *Process Refinement and Asynchronous Composition with Modalities*, in "Proceedings of the 2nd International Workshop on Abstractions for Petri Nets and Other Models of Concurrency (APNOC'10)", Braga, Portugal, N. SIDOROVA, A. SEREBRENIK (editors), June 2010, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/EHH-apnoc10.pdf

[70] J. ESPARZA, K. HELJANKO. , *Unfoldings - A Partial-Order Approach to Model Checking*, EATCS Monographs in Theoretical Computer Science, Springer, 2008

[71] É. FABRE, A. BENVENISTE, C. JARD, S. HAAR. *Diagnosis of Asynchronous Discrete Event Systems, a Net Unfolding Approach*, in "IEEE Trans. Aut. Control", 2003, vol. 48 (5), pp. 714-727

[72] É. FABRE, A. BENVENISTE, C. JARD, S. HAAR. *Distributed monitoring of concurrent and asynchronous systems*, in "Discrete Event Dynamic Systems: theory and application", 2005, vol. 15 (1), pp. 33-84, Preliminary version: Proc. CONCUR 2003, LNCS 2761, pp.1–28, Springer

[73] B. FINKBEINER, S. SCHEWE. *Uniform distributed synthesis*, in "Proc. of the 20th IEEE Annual Symposium on Logic in Computer Science (LICS'05)", IEEE Computer Society Press, 2005, pp. 321–330

[74] P. GASTIN, B. LERMAN, M. ZEITOUN. *Distributed games with causal memory are decidable for series-parallel systems*, in "Proceedings of the 24th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'04)", Chennai, India, K. LODAYA, M. MAHAJAN (editors), Lecture Notes in Computer Science, Springer, December 2004, vol. 3328, pp. 275-286, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/GLZ-fsttcs04.pdf

[75] P. GASTIN, N. SZNAJDER, M. ZEITOUN. *Distributed synthesis for well-connected architectures*, in "Proceedings of the 26th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'06)", Kolkata, India, N. GARG, S. ARUN-KUMAR (editors), Lecture Notes in Computer Science, Springer, December 2006, vol. 4337, pp. 321-332 [*DOI :* 10.1007/11944836_30], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/GSZ-fsttcs2006.pdf

[76] S. HAAR, A. BENVENISTE, É. FABRE, C. JARD. *Partial Order Diagnosability Of Discrete Event Systems Using Petri Net Unfoldings*, in "42nd IEEE Conference on Decision and Control (CDC)", 2003

[77] S. HAAR. *Probabilistic Cluster Unfoldings*, in "Fundamenta Informaticae", 2003, vol. 53 (3-4), pp. 281-314

[78] S. HAAR, C. JARD, G.-V. JOURDAN. *Testing Input/Output Partial Order Automata*, in "Proc. TESTCOM/FATES", LNCS, Springer, 2007, vol. 4581, pp. 171-185, LNCS 4581

[79] O. KUPFERMAN, M. Y. VARDI. *Synthesizing Distributed Systems*, in "Proc. of the 16th IEEE Annual Symposium on Logic in Computer Science (LICS'01)", IEEE Computer Society Press, 2001

[80] S. LAFORTUNE, Y. WANG, T.-S. YOO. *Diagnostic Decentralisé Des Systèmes A Evénements Discrets*, in "Journal Europeen des Systèmes Automatisés (RS-JESA)", August 2005, vol. 99, n^{o} 99, pp. 95–110

[81] K. G. LARSEN, P. PETTERSSON, W. YI. *Compositional and symbolic model-checking of real-time systems*, in "Proc. of RTSS 1995", IEEE Computer Society, 1995, pp. 76-89

[82] S. MOHALIK, I. WALUKIEWICZ. *Distributed Games*, in "Proc. of the 23rd Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'03)", LNCS, Springer, 2003, vol. 2914, pp. 338–351

[83] A. PNUELI, R. ROSNER. *Distributed reactive systems are hard to synthesize*, in "Proc. of the 31st Annual IEEE Symposium on Foundations of Computer Science (FOCS'90)", IEEE Computer Society Press, 1990, vol. II, pp. 746–757

[84] L. RICKER, K. RUDIE. *Know Means No: Incorporating Knowledge into Discrete-Event Control Systems*, in "IEEE Transactions on Automatic Control", September 2000, vol. 45, n^{o} 9, pp. 1656–1668

[85] L. RICKER, K. RUDIE. *Knowledge Is a Terrible Thing to Waste: Using Inference in Discrete-Event Control Problems*, in "IEEE Transactions on Automatic Control", MarchSeptember 2007, vol. 52, n^{o} 3, pp. 428–441

[86] H. L. S. YOUNES, R. G. SIMMONS. *Statistical probabilistic model checking with a focus on time-bounded properties*, in "Inf. Comput.", September 2006, vol. 204, pp. 1368–1409 [*DOI :* 10.1016/J.IC.2006.05.002], http://dl.acm.org/citation.cfm?id=1182767.1182770