INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Team MExICo

Modeling and Exploitation of Interaction and Concurrency

Saclay - Île-de-France

Theme : Programs, Verification and Proofs

Table of contents

1. Team

Research Scientists

Benedikt Bollig [Research Associate (CR) CNRS]
Stefan Haar [Team leader, Research Director (DR) Inria since October 2010, HdR]
Marc Zeitoun [Professor on leave from Bordeaux, HdR]

Faculty Members

Thomas Chatain [Maître de Conférence ENS Cachan]
Paul Gastin [Professor ENS Cachan, HdR]
Serge Haddad [Professor ENS Cachan, HdR]
Stefan Schwoon [Maître de Conférence ENS Cachan, Chaire INRIA]

PhD Students

Sandie Balaguer [since September 2010]
Aiswarya Cyriac [since September 2010]
Hilal Djafri [since October 2008]
Dorsaf El Hog [until October 2010]
Benjamin Monmege [since September 2010]
Cesar Rodriguez [since September 2010]
Akshay Sundararaman [co-supervised with CMI, India; defended July 2010]

Administrative Assistant

Isabelle Biercewicz [Secretary (SAR) Inria]

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.

2.2. Highlights

- The MExICo team co-organized (jointly with LIAFA) the 21st International Conference on Concurrency Theory (CONCUR'10), the leading theoretical conference on concurrency-related research, in Paris, from August 31 to September 3, 2010.
- Recently, a logical characterization of weighted automata was established by Droste and Gastin, in terms of a (restricted) weighted MSO logic capturing the recognizable formal power series. The key idea is to interpret existential and universal quantifications as the operations sum and product from a semiring. To make this definition work, however, one has to restrict the universal first-order quantification, which, otherwise, appears to be too powerful and goes beyond the class of recognizable series.

  In the paper "Pebble weighted automata and transitive closure logics" [24] by Benedikt Bollig, Paul Gastin, Benjamin Monmege, and Marc Zeitoun at well-known conference *ICALP'10*, we followed a different approach, which is inspired by the theory of pebble automata on words and trees: Instead

of restricting the logic, we introduce new classes of weighted automata on words. Equipped with pebbles and a two-way mechanism, they go beyond the class of recognizable formal power series, but capture a weighted version of first-order logic with bounded transitive closure.

More precisely, we introduce pebble weighted automata on words and establish expressive equivalence to weighted first-order logic with bounded positive transitive closure and unrestricted use of quantification, extending the classical Boolean case for words. Our equivalence proof makes a detour via another natural concept, named nested weighted automata. The transitive closure logic also yields alternative characterizations of the (classical) recognizable formal power series.

These results are not only of theoretical interest. They also lay the basis for quantitative extensions of database query languages such as XPath, and may provide tracks to evaluate quantitative aspects of XML documents. The framework of weighted automata is natural for answering questions such as "How many nodes are selected by a request?", or "How difficult is it to answer a query?". The navigational mechanism of pebble automata is also well-suited in this context. For these reasons, our work is a first step before tackling similar questions on trees.

3. Scientific Foundations

3.1. Concurrency

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

Glossary

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, Cesar Rodriguez, Stefan Schwoon, Marc Zeitoun.

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 [63], [64], [68], 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 [45].

3.1.3. 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*.

3.1.4. 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 [64], [46], 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 [62], [76]). 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 [44], [48]). 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 [81], [82]. Automatizing such procedures for the supervision and management of distributed and locally monitored asynchronous systems is a mid-term goal of *MExICo*. The participants above have established a working group on the subject of distributed diagnosability.

3.1.5. Verification of Concurrent Recursive Programs

Participants: Benedikt Bollig, Aiswarya Cyriac, Paul Gastin, Marc Zeitoun.

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.

3.1.6. Testing

Participants: Benedikt Bollig, Paul Gastin, Stefan Haar.

3.1.6.1. 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.

3.1.6.2. 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 [70], [52] 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.

3.1.6.3. Local Testing

Message-Sequence-Charts (MSCs) provide models of behaviors of distributed systems with communicating processes. An important problem is to test whether an implementation conforms to a specification given for instance by an HMSC. In *local testing*, one proceeds by injecting messages to the local processes and observing the responses: for each process p, a local observer records the sequence of events at p. If each local observation is consistent with some MSC defined by the specification, the implementation passes the test. If local testing on individual processes suffices to check conformance, the given specification (an HMSC language) is called locally testable. Local testability turns out to be undecidable even for regular HMSC languages [44]; the main difficulty lies in the existence of implied scenarios, i.e., global behaviors which are locally consistent with different specification scenarios. There are two approaches to attack the problem of local testing in light of this bottleneck. One is to allow joint observations of tuples of processes. This gives rise to the problem of k-testability where one allows joint observations of up to k processes [48]. We will look for structural conditions on the model or the specification ensuring k-testability. Another tactic would be to recognize that practical implementations always work with bounded buffers and impose an upper bound B on the buffer size. The set of B-bounded MSCs in the k-closure of a regular MSC language is again regular, so the B-bounded k-testability problem is decidable for all regular HMSC-definable specifications. The focus could now be on efficiently identifying the smallest k for which an HMSC specification is k-testable. Another interesting problem is to identify a minimal set of tests to validate a k-testable specification.

The first step that should be reached in the near future is the completion of asynchronous testing in the setting without any big-step synchronization. In parallel, work on the problems in local testing should progress sufficiently to allow, in a mid-term perspective, to understand the relations and possible interconnections between local (i.e. distributed) and asynchronous (centralized) testing. The mid-to long term goal (perhaps not

yet to achieve in a four-year term) 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, Marc Zeitoun.

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 [61] 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 [79], [74], [78], [65], [67]. 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 [66],[7]. 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: Benedikt Bollig, 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.

3.3. Management of Quantitative Behavior

Participants: Sandie Balaguer, 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.

3.3.2.1. Non-sequential probabilistic processes.

Practical fault diagnosis requires to select explanations of *maximal likelihood*; this leads therefore to the question what the probability of a given partially ordered execution is. In Benveniste et al. [47], [42], 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 [69], 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.

Recently, we started a new line of research with Anne Bouillard, Sidney Rosario, and Albert Benveniste in the DistribCom team at INRIA Rennes, studying the likelihood of occurrence of non-sequential runs under random durations in a stochastic Petri net setting.

Once the properties of the probability measures thus obtained are understood, it will be interesting to relate them with the two above models in logical time, and understand their differences. Another mid-term goal, in parallel, is the transfer to diagnosis with possible cooperation with René Boel's group in Ghent/Belgium.

3.3.2.2. Distributed Markov Decision Processes

Participant: Serge Haddad.

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 cooperate on this subject with Eric Fabre in the DistribCom team at INRIA Rennes, in the context of the DISC project.

3.3.3. 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 [77] [55]. 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.

3.3.3.1. 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 [43]. 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.

3.3.3.2. Equivalences between Models with Time and Concurrency (EMoTiCon)

Participants: Sandie Balaguer, Thomas Chatain, Stefan Haar, Serge Haddad.

This was the subject of a project of the Farman institute of ENS Cachan in collaboration with the LURPA (laboratory for automated production at ENS Cachan).

Due to the dramatic development of the techniques that aim at improving the security of automated systems (synthesis, verification, test...), several classes of models are often needed to study a complex system, either to give several views of the system or to study the same aspect of the system using several techniques. Thus one often needs to transform a model from one formalism to another or to compare models written in different formalisms (time Petri nets, networks of timed automata...), that have common features: they allow one to model both (dense) time and concurrency. These transformations are usually done by hand and rely on natural equivalences between the basic components of the models. For instance, a state of an automaton corresponds intuitively to a place of a Petri net; a transition of a Petri net corresponds to a tuple of synchronized transitions in a network of automata; the interval of possible delays associated with a transition of a time Petri net corresponds to a pair invariant/guard in a timed automaton. But these natural equivalences do not apply easily to general models. And since the transformations are usually built on case studies and for ad-hoc reasons, no effort is made to generalize them and most often the relations between the initial model and the transformed one are not formalized.

Nevertheless we see clearly that the transformations on case studies tend naturally to preserve concurrency. Moreover this property is appreciated because it improves the readability of the transformation and makes the transformed model faithful to the initial one and to the modeled system. But these ad hoc transformations are difficult generalize. Thus, not surprisingly, the first works about formal comparison of the expressiveness of different models [57], [56], [84] [58], [59], [60], [53], [54] did not take preservation of concurrency into account. These works make extensive use of tricks that destroy concurrency and focus only on the preservation of sequential (interleaving) timed semantics.

In contrast, 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.

3.3.4. Weighted Automata and Weighted Logics

Participants: Benedikt Bollig, Paul Gastin, Benjamin Monmege, Marc Zeitoun.

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 [41]. 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 [4]. That paper, however does not give final solutions to all the aforementioned problems. It identifies directions for future research that we will be tackling.

4. Application Domains

4.1. Panorama

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.
- Several recent Intelligent Transport Systems projects aim at providing assistance to drivers, in the way of (partially) automated motorways. We will focus on the modeling and analysis of the collision avoidance problems in critical short sections of motorways.

This list is likely to grow over the next years as we continue our research.

4.2. Autonomous Telecommunications Systems: In-Band Supervision

Participants: Stefan Haar, Serge Haddad.

In the context of traditional hard-wired communication networks, supervision structures for managing faults, configuration, provisioning etc could be developed with a fixed infrastructure, and perform the communication between sensors, supervisors, policy enforcement points etc over a separate network using separate hardware. This rigid, **out-of-band** technology does not survive passing to today's and tomorrow's services and networks. In fact, the dynamic mobility of services combined across sites and domains cannot be captured unless the network used for supervision evolves in the same way and simultaneously, which rules out static solutions; but providing out-of-band infrastructure that grows with the networks to be supervised would be prohibitively expensive, if at all technically feasible. *Heterogeneity* is the other feature of modern networks that forces a change, since different domains are not likely to agree on a pervasive third-party supervision. Rather, the providers will keep control over the internal state and evolution of their domain, and accept only exchange through standardized outward interfaces.

Supervision has thus to be re-invented on an *in-band*, *autonomous* base: monitoring probes deployed on the web, dysfunctions on one peer node diagnosed by another peer in a network with changing configuration, enhanced supervisor and actor capacities of services, etc. *MExICo* will work on improving the interoperability of service components through continued application of e.g. distributed techniques for control and diagnosis.

4.3. Traffic Safety Control

Participant: Serge Haddad.

The *Intelligent Transport Systems* **(ITS)** community tries to deal with the numerous challenges that arise when designing secure and reliable software dedicated to automatic transport systems.

Several recent ITS projects aim at providing assistance to drivers and deal with partially automated motorways. The community investigated first a fully automated infrastructure and vehicles approach (as in the PATH project [72]) in the 1990's. That approach was then abandoned in favor of a new line of research and development activities, more centered on safety strategies to ensure properties such as Collision Avoidance or Safety Margin for Assistance Vehicles [73].

This vision relies on cooperative systems where *"road operators, infrastructure, vehicles, their drivers and other road users will cooperate to deliver the most efficient, safe, secure and comfortable journeys"* [51]. Implementing such a system then follows a peer-to-peer organization where each vehicle must fully cooperate in a time-constrained and safety-critical environment.

In that context, many projects are dealing with safety-oriented applications based on sensors, communication devices and protocols as well as distributed traffic management systems involving cooperation between the infrastructure and vehicles [50], [49], [83]. Thus, reliability, flexibility in the design as well as safety are primary issues. Such systems are even more complex to analyze than previous distributed systems. Consequently, there is a need for a specific methodology and tools to design and analyze them.

We will focus on an approach for the modeling and analysis of the collision avoidance problems in critical short sections of motorways with the aim to check whether a control strategy exists depending on the parameters (speed, safety distances, etc.). We intend to cope with the undecidability of such problems by appropriate discretizations and with high complexity of the obtained systems by using elaborated data structures based on decision diagrams.

4.4. Web Services

Participants: Stefan Haar, Serge Haddad.

Specific applications targeted by *MExICo* include the problem of adaptation in Service-Oriented Computing (SOC). The challenge is here twofold, stemming both from the distributed nature of services (scattered over the entire web) and their heterogeneous origins.

4.4.1. Context

Web services have become the most frequently used model of design and programming based on components for business applications. Web service languages like BPEL have useful constructors that manage for instance exceptions, (timed guarded) waiting of messages, parallel execution of processes, distant service invocations, etc. Interoperability of components is based on interaction protocols associated with them and often published on public or private registers. In the framework of Web services, these protocols are called abstract processes by contrast with business processes (i.e. services). Composition of components must be analyzed for several reasons and at least to avoid deadlocks during execution. This has led to numerous works that focus on compositional verification, substitution of a component by another one, synthesis of adaptators, etc., and triggered a push towards a unifying theoretical framework (see e.g. [80], [85])

4.4.2. Problems

Interoperability requires that when a user or a program wants to interact with the component, the knowledge of the interaction protocol is enough. Our previous works have shown that the interaction protocols can be inherently ambiguous: no client can conduct a correct interaction with the component in every scenario. This problem is even more complex when the protocol can evolve during execution due to adaptation requirements. The composition of components also raises interesting problems. When composing optimal components (w.r.t. the number of states for instance) the global component can be non optimal. So one aims at reducing a posteriori or better on the fly the global component. At last, the dynamical insertion of a component in a business process requires to check whether this insertion is behaviorally consistent [86], [75]

We do not intend to check global properties based on a modular verification technique. Rather, given an interaction protocol per component and a global property to ensure, we want to synthesize an adaptator per component such that this property is fulfilled or to detect that there cannot exist such adaptators [71]. In another research direction, one can introduce the concept of utility of a service and then optimize a system i.e. keeping the same utility value while reducing the resources (states, transitions, clocks, etc.).

5. Software

5.1. Software

5.1.1. 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/

5.1.2. Mole: an unfolder for Petri Nets

Participants: Stefan Schwoon [correspondant], Cesar Rodriguez.

`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. In the context of MExiCo, we are extending it to handle contextual Petri nets. A preliminary implementation with reasonable performance has been achieved, which we intend to improve further into a viable, efficient tool.

Details on Mole can be found at http://www.lsv.ens-cachan.fr/~schwoon/tools/mole/

6. New Results

6.1. Quantitative systems

6.1.1. Real-time systems

(Serge Haddad, joint work with B. Bérard, M. Sassolas, supported by DIGITEO COCHAT)

Interrupt Timed Automata (ITA) have been introduced to model multi-task systems with interruptions. They form a subclass of stopwatch automata, where the real valued variables (with rate 0 or 1) are organized along priority levels. While reachability is undecidable with usual stopwatches, the problem was proved decidable for ITA. We have extended this work. First we have answered some questions left open about expressiveness, closure, and complexity for ITA. Then we have investigated the verification of real time properties over ITA. While we have proved that model checking a variant of the timed logic TCTL is undecidable, we have nevertheless given model checking procedures for two relevant fragments of this logic: one where formulas contain only model clocks and another one where formulas have a single external clock [28].

6.1.2. Quantitative analysis of web services

(Serge Haddad, joint work with L. Mokdad and S. Youcef, supported by ANR CheckBound)

We have pursued this research in two complementary directions. On the one hand, we have refined the analysis of response time of a Web service which is generally considered as a fixed composition of elementary services with exponential response time. This hypothesis is often unrealistic and thus we have proposed analytical formulas for the response time of web services where the composition involves a variable number of elementary services whose distribution is no more necessarily exponential [33]. On the other hand, we have enlarged the standard UDDI registry of services with a service selection based on a multi-criteria approach. This work has led to a prototype based on the open source jUDDI registry [33], [34].

6.1.3. Weighted Automata

Weighted automata provide a very general framework for the modeliing of quantitative systems, but a strong theory for specifying and verifying quantitative properties still needs to be developed. In 2005, Droste and Gastin introduced a quantitative (weighted) extension of MSO logic and established its relationship with weighted automata. However, this logical characterization requires a considerable restriction of the first-order fragment, which makes the logic inconvenient for specification purposes. In [24], we introduced new classes of weighted automata that generalize classical weighted automata and capture unrestricted first-order logic. We gave precise logical characterizations of these new models and, at the same time, provide new characterizations of the classical recognizable formal power series. As future work, motivated by applications in database theory and XML, we plan to extend our results to trees, which will require substantially more work and different techniques. This will be part of Benjamin Monmege's PhD program, which is aiming at quantitative extensions of automata and logics.

6.2. Concurrency

6.2.1. Dynamic Distributed Systems

(Benedikt Bollig, joint work with Loic Helouet, IRISA/INRIA, Rennes)

In [25], we have tackled the problem of modeling, analyzing, and synthesizing distributed systems with an evolving communication topology and an unbounded number of processes. We introduced dynamic communicating automata (DCA), an extension of communicating finite-state machines that allows for dynamic creation of processes. A DCA comes with three types of actions: (1) a new process can be created, (2) a message can be sent to an already existing process, and (3) a message can be received from an existing process. Processes are identified by means of local process variables, whose values can change dynamically during an execution of an automaton and be updated when a message is received. Messages are stored in bidirectional unbounded FIFO channels, which exist between any two created processes. The behavior of a DCA can be described as a set of message sequence charts (MSCs). The realizability problem for DCA is: given a dynamic MSC grammar (a variant of the fork-and-join grammars introduced by Leucker, Madhusudan, and Mukhopadhyay), is there a DCA defining the same set of MSCs ? We have shown that this problem is decidable in exponential time, and identified a class of realizable grammars that can be implemented by finite DCA.

6.2.2. Design of compositional distributed systems

(Serge Haddad, joint work with D. El Hog and R. Hennicker)

Modal specification have been introduced in order to allow a "continuous" design process from the specification to the implementation of compositional systems. Since their introduction, they have been the topic of numerous research extending their application area (for instance to timed or probabilistic systems). However for decidability reasons, composition mode is synchronous which is incompatible with a real distributed system. So we have proposed a framework for the specification of infinite state systems based on Petri nets with distinguished may- and must-transitions (called modalities) which specify the allowed and the required behavior of refinements and hence of implementations. Formally, refinements are defined by relating the modal language specifications generated by two modal Petri nets according to the refinement relation for modal language specifications. We have showed that this refinement relation is decidable if the underlying modal Petri nets are weakly deterministic. We have also showed that the membership problem for the class of weakly deterministic modal Petri nets is decidable. As an important application of our approach we consider I/O-Petri nets which are obtained by asynchronous composition and thus exhibit inherently an infinite behavior [31].

6.2.3. Diagnosis

Refinements were obtained in the analysis of unfoldings for the purposes of diagnosis. The *reveals* relation in occurrence nets has led to novel criteria on observability in labeled Petri nets, presented in [17]. The study of reveals-relations in more general settings promises also more insight into other asynchronous models, such as time-guarded and contextual systems. Furthermore, the particularities of diagnosis for distributed systems with partial order semantics appear more clearly now; a framework based on metric topologies [32] for event structures allows us to distinguish and compare *weak* and *strong* observability and diagnosability properties.

Moreover, progress was made on the question of computing the reveals relation for a given occurrence net. Together with Christian Kern, Stefan Haar and Stefan Schwoon developed an efficient algorithm for computing the relation for a given, finite occurrence net. A publication is being prepared.

6.2.4. Contextual Net Unfoldings

Following the initial publication on contextual net unfoldings [2], which established the necessary theoretical background for obtaining a finite complete prefix, César Rodríguez and Stefan Schwoon worked on the algorithmics and an efficient implementation. First results were published in [22], [40]; they concern the establishment of a concurrency relation that allows to more efficiently compute the events that make up the unfolding. An implementation, which promises to provide comparable performance to existing tools for (conventional) Petri nets, is in preparation.

6.2.5. Timed concurrent Systems

Sandie Balaguer, Thomas Chatain and Stefan Haar have presented in [21] a translation of 1-bounded Time Petri Nets into a Network of Timed Automata and introduced an equivalence which takes the distribution of actions into account. This translation is extensible to bounded TPNs. We first use S-invariants to decompose the net into components that give the structure of the automata network, then we add clocks to provide the timing information. Although we have to use an extended syntax in the timed automata, this is a novel approach towards preservation of concurrency: the localisation of actions to one process, i.e. S-invariant or automaton component, is preserved.

6.3. Interaction

6.3.1. Distributed Synthesis with application to covert channels

(Serge Haddad, joint work with B. Bérard, M. Sassolas and M. Zeitoun, supported by DIGITEO COCHAT)

Given (1) an architecture defined by processes and communication channels between them or with the environment, and (2) a specification on the messages transmitted over the channels, distributed synthesis aims at deciding existence of local programs, one for each process, that together meet the specification, whatever the environment does. Recent work shows that this problem can be solved when a linear preorder sorts the agents w.r.t. the information received from the environment. We have showed a new decidability result in the case where this preorder is broken by the addition of noisy agents embedded in a pipeline architecture. This case cannot be captured by the classical framework. Besides, this architecture makes it possible to model particular security threats, known as covert channels, where two users (the sender and the receiver) manage to communicate via a noisy protocol, and despite incomparable views over the environment [37].

6.3.2. Factorization Properties of Symbolic Unfoldings of Colored Petri Nets

The unfolding technique is an efficient tool to explore the runs of a Petri net in a true concurrency semantics, i.e. without constructing all the interleavings of concurrent actions. But even small real systems are never modeled directly as ordinary Petri nets: they use many high-level features that were designed as extensions of Petri nets. Thomas Chatain and Eric Fabre have shown in [29] that the symbolic unfolding of a product of colored Petri nets can be expressed as the product of the symbolic unfoldings of these nets. This is a necessary result in view of distributed computations based on symbolic unfoldings, as they have been developed already for standard unfoldings, to design modular verification techniques, or modular diagnosis procedures, for example.

7. Contracts and Grants with Industry

7.1. Contracts and Grants with Industry

So far, several contacts with industry have been established, but no bilateral contracts have materialized yet. Cooperations with France Télécom , Alcatel-Lucent and NEC are currently being developped within the EU IP UNIVERSELF, which has started in October of 2010.

8. Other Grants and Activities

8.1. Regional Initiatives

8.1.1. Farman Project EMoTiCon

Participants: Sandie Balaguer, Thomas Chatain, Stefan Haar, Serge Haddad.

The EMoTiCon project was a collaboration between researchers from *MExICo* and from the laboratory of automated production at ENS Cachan. As such, this collaboration is funded by the Farman institute of ENS Cachan.

The scentific context of the EMoTiCon project is the variety of formalisms used to model distributed real-time systems. We aim at formalizing and automating translations between these models, that preserve both the timed semantics and the concurrent semantics. This problem is described in Section 3.3.3.2.

The project ended in September 2010.

8.2. National Actions

8.2.1. ANR DOTS

Participants: Benedikt Bollig, Thomas Chatain, Paul Gastin, Serge Haddad, Marc Zeitoun.

The DOTS project is a collaboration with researchers from IRCCyN, IRISA, LAMSADE, LaBRI and LSV.

The scientific context of the DOTS project is specification, verification and design of information systems. Complex systems, such as embedded systems that are widely used nowadays (telecommunication, transport, automation), are often distributed –composed of several components that communicate together–, timed –contain timing constraints–, and open –interact with their environment. Each of these aspects considered separately is now relatively well understood and corresponds to an active research area. The big challenge is to deal with systems which present several of these features.

The aim of the DOTS project is to associate researchers specialized in verification of different aspects mentioned above in order to tackle problems that emerge when considering several features simultaneously. In this way we plan to significantly advance both theory as well as algorithmics of design and verification of distributed, open and timed systems.

The research of *MExICo* about distributed control (Section 3.2.2) and real time distributed systems (Section 3.3.3) take place in the DOTS project.

8.2.2. ANR CHECKBOUND ANR-06-SETI-002

Participants: Hilal Djafri, Serge Haddad.

The increasing use of computerised systems in all aspects of our lives gives an increasing importance on the need for them to function correctly. The presence of such systems in safety-critical applications, coupled with their increasing complexity, makes indispensable their verification to see if they behaves as required . Thus the model checking which is the automated manner of formal verification techniques is of particular interest. Since verification techniques have becomemore efficient and more prevalent, the natural extension is to extend the range of models and specification formalisms to which model checking can be applied. Indeed the behaviour of many real-life processes is inherently stochastic, thus the formalism has been extended to probabilistic model checking. Therefore, different formalisms in which the underlying system has been modelled by Markovian models have been proposed.

Stochastic model checking can be performed by numerical or statistical methods. In model checking formalism, models are checked to see if the considered measures are guaranteed or not, bounding techniques become useful.We propose to apply Stochastic Comparison technique for numerical stochastic model checking. The main advantage of this approach is the possibility to derive transient and steady-state bounding distributions as well as the possibility to avoid the state space explosion problem. For the statistical model checking we propose to study the application of perfect simulation by coupling in the past. This method has been shown that to be efficient when the underlying system is monotonous for the exact steady-state distribution sampling. We consider to extend this approach for transient analysis and to model checking by means of bounding models and the stochastic monotonicity. One of difficult problems for model checking formalism, we envisage to study is when the state space is infinite. In some cases, it would be possible to consider bounding models defined in finite state space.

Indeed, formal verification using model checking and performance and dependability evaluation have a lot of things in common. We think that it would be interesting to apply the methods that we have a large experience in quantitative evaluation in the context of stochastic model checking.

8.2.3. DIGITEO PhD Grant (LoCoReP)

Participants: Benedikt Bollig, Aiswarya Cyriac, Paul Gastin, Marc Zeitoun.

Benedikt Bollig and Paul Gastin obtained a DIGITEO PhD grant for their student Aiswarya Cyriac. The aim of the PhD will be to design linear-time temporal logics for concurrent recursive programs.

8.2.4. DIGITEO 2009-27HD CoChaT: Covert Channels in Timed Systems

Participant: Serge Haddad.

Attacks with timing channels have been described and simulated for instance on TCP/IP protocols, Web communications or cryptographic operations. The scientific objective of the CoChaT project is to study the conditions underwhich such attacks can occur in timed systems, with two main directions. a. The first step consists in defining a theoretical framework, in which timing channels can be formally described. b. A second part of thework concerns the design of detection and verification algorithms, for which decidability issues are involved. Progress in both steps will have to take into account practical examples like the case studies mentionned above, in order to validate the formal approach.

8.2.5. DIGITEO 2010-LoCoRep

Benedikt Bollig and Paul Gastin obtained a PhD DIGITEO grant for their student Aiswarya Cyriac. The project is entitled "Temporal Logic for Concurrent Recursive Programs (LoCoReP)". The aim of the PhD will be to design linear-time temporal logics 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.

8.2.6. INRIA Associated teams

Participants: Serge Haddad, Stefan Haar.

Following Dorsaf El Hog's decision to stop her thesis research on adaptation of , Serge Haddad and Stefan Haar have withdrawn from their participation in the associated team FOSSA led by the DistribCom team at INRIA Rennes, with the University of Texas, Austin. The objective is of the AT *Formalizing Orchestration and Secure Services Analysis*; see http://www.irisa.fr/distribcom/FOSSA2010/Fossa10.html.

8.3. European Initiatives

8.3.1. DISC: Grant Agreement 224498

Serge Haddad and Stefan Haar are participating, as associate members of INRIA Rennes, in the Project on *Distributed Supervisory Control of Large Plants - DISC*. The European Commission supports the project financially by the EU.ICT program, Challenge ICT-2007.3.3 (Information and Communication Technologies (ICT)). 1 September 2008 - 1 Septemer 2011. Project partners:

- University of Cagliari (coordinator),
- CWI - Amsterdam, Ghent University,
- Technical University of Berlin,
- University of Zaragoza,
- INRIA,
- Akhela s.r.l. Italy,
- Czech Academy of Sciences,
- Ministry of the Flemish Government,
- CyBio AG.

8.3.2. IP Univerself, Grant Agreement 257513

Serge Haddad and Stefan Haar are among the INRIA participants of the IP Univerself on autonomous Management in telecommunications, along with members of the Distribcom group at INRIA Rennes and the MADYNES group at INRIA Nancy. The project consortium is :

- Alcatel Lucent France (coordinator),
- Universiteit Twente,
- Alcatel Lucent Ireland,
- Alcatel Lucent Deutschland,
- Valtion Teknillinen Tutkimuskeskus (Finland),
- University of Piraeus,
- France Telecom,
- Telecom Italia,
- National University of Athens,
- Fraunhofer-Gesellschaft zur Förderung der Angewandten Forschung,
- Interdisciplinary Institute for Broadband Technology,
- Telefonica Investigacion y Desarrollo,
- Thales Communications,
- INRIA,
- Nec Europe,
- University of Surrey,
- University College London
- IBBT (Belgium).

Univerself has been launched in October 2010 and is scheduled for four years. We anticipate that a major part of our research activity on diagnosis, distributed optimization, service interaction and testing will be linked to the project.

8.3.3. NoE HYCON2

Serge Haddad and Stefan Haar participate in the european project FP7-ICT, Network of Excellence: *Highly-Complex and Networked Control Systems, Hycon 2*, coordinateur: LLS (Supelec), 23 participants.

in this project, we anticipate to intensify contact with researchers in control, to explore a field adjacent to our work on discrete event and quantitative systems and open future cooperations and topics.

8.4. International Initiatives

8.4.1. ARCUS Inde

Most participants of the team participate in the sub-project 4, Formal approaches for computer systems, of the Ile-de-France/Inde project of the ARCUS program (Region Ile-de-France and Foreign Affairs Ministry, France), initially funded for 3 years (2008 – 2010) and extended until August 2011.

9. Dissemination

9.1. Scientific animation

9.1.1. Benedikt Bollig

was a member of the programme committee for DOTS'10 and YR-CONCUR'10, co-located with CONCUR'10. He was a member of the commission scientifique INRIA Saclay. He gave an invited talk at the LaBRI, Bordeaux, on "Distributed Timed Automata with Independently Evolving Clocks" (16/09/2010). Moreover, he was on the defense committee for the PhD thesis of Akshay Sundararaman (as co-supervisor), and served as a reviewer for many international conferences and journals.

9.1.2. Thomas Chatain

supervises Sandie Balaguer's PhD thesis on concurrency in real-time distributed systems. He was a member of the program committee of the *International Conference on Application of Concurrency to System Design (ACSD) 2010*.

9.1.3. Paul Gastin

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

He organized (co-chair with François Laroussinie) the *International Conference on Concurrency Theory (CONCUR'10)* in Paris, September 2010.

He co-organized the international workshops on *Quantitative Models: Expressiveness and Analysis* in Dagstuhl, January 2010; on *Automata, Concurrency and Timed Systems (ACTS)* in Chennai, February 2010; on *Weighted Automata: Theory and Applications (WATA'10)* in Leipzig, May 2010.

He was/is a member of the program committees of CONCUR'10 and CONCUR'11 (*International Conference on Concurrency Theory*), LATA'10 (*International Conference on Language and Automata Theory and Applications*), DLT'10 and DLT'11 (*International Conference on Development in Language Theory*).

He gave a tutorial (4 hours) on weighted logics at the workshop *Weighted Automata: Theory and Applications (WATA'10)* in StLeipzig, May 2010.

He was on the scientific committee of the international workshop on *Automata, Concurrency and Timed Systems (ACTS)* in Chennai, February 2010.

He was on the defense committee for the PhD thesis of Akshay Sundararaman (as supervisor).

He is the head of the computer science department of ENS Cachan. He was the head of the Parisian Master of Research in Computer Science (until August 2009).

9.1.4. Stefan Haar

has started a term as an associate editor for *Discrete event dynamic systems: theory and application* in January 2010. He participated in the publicity of the conference *CONCUR 2010* in Paris, and was a member of the program committee of *Workshop On Discrete Event Systems (WODES) 2010* and of *PNSE 2010*. He is the correspondent of the *DRI* (international relations service) of INRIA for the Saclay center, and represents INRIA in the steering commitee for the French consortium of IIT Rajasthan, Jodhpur, India.

9.1.5. Serge Haddad

has been a member of the editorial board of the journal *Technique et Science Informatiques* since 2007, and a member of the steering committee of the *International Conference on Applications and Theory of Petri Nets (ICATPN)* since 2001.

In 2010, he was

- member of the PC of *7th International Conference on Quantitative Evaluation of Systems (QEST)*, Williamsburg, USA (september 2010),
- member of the PC of *4th International Workshop on Verification and Evaluation of Computer and Communication Systems (VECOS)*, Paris, France, July 2010,
- member of the PC of *31th International Conference on Application and Theory of Petri nets (ATPN)*, Braga, Portugal, june 2010,
- and member of the PC of workshops APNOC and SUMO, both associated with *31th International Conference on Application and Theory of Petri nets (ATPN)*, Paris, France, june 2010,

and has participated in the following PhD and HdR committees:

- **9 december 2010** Y. Li, *Diagnosis of Large Software Systems Based on Colored Petri Nets*, ThÃ¨se de l'Université d'Orsay, Jury : S. Haddad (président), A. Benveniste, L. Concole *(rapporteurs)*, P. Dague, T. Melliti, F. Zaidi;
- **22 october 2010** M. Yeddes, *Approche formelle de vérification dans le cadre de systèmes à événements discrets et des systèmes hybrides*, Habilitation à diriger des recherches de l'Université de Paris 6, Jury : F. Kordon (président), D. Buchs, S. Haddad, J-J. Lesage (rapporteurs), K. Barkaoui, B. Bérard;
- **21 october 2010** D-T. Nguyên, *Vérification symbolique de modèles à l'aide de systèmes de réécritures dédiés*, Thèse de l'Université d'Orléans, Jury : S. Haddad (president), D. Buchs, F. Kordon (rapporteurs), Y. Boichut, A. Bouajjani, J-M. Couvreur, G. Sutre.

Cooperation with professor R. Hennicker University Ludwig-Maximilians of Munich , and with associate professor F. Rosa Velardo University of Madrid. Serge Haddad is responsible of years L3 and M1 of the computer science department of ENS Cachan.

9.1.6. Stefan Schwoon

supervises César Rodríguez' PhD thesis on contextual Petri net unfoldings. He participated in the organization of the CONCUR 2010 conference in Paris. He gave invited presentations at the summer school MOVEP (Modelling and Verifying Parallel Processes) in Aachen, the ACTS II workshop in Chennai, and the seminar of the Software Security Lab of CEA LIST Saclay this year.

9.2. Visits and Visitors

9.2.1. Visits received

9.2.1.1. K. Narayan KUMAR

Professor at Chennai Mathematical Institute (CMI); visited from May 24 to June 12 and from August 26/8 to September 6.

9.2.1.2. Madhavan MUKUND

Professor at Chennai Mathematical Institute (CMI); visited from April 21 to 30 and from July 1 to 11.

9.2.1.3. Akshay SUNDARARAMAN

PhD student co-supervised between CMI and MExICo; visited from March 9 to April 14 and June 25 to July 4.

9.2.1.4. Rolf HENNICKER

Professor at Ludwig-Maximilians-Universität of Munich ; visited the team on invitation by ENS Cachan from May 11 through 21.

9.2.1.5. Martin Leucker

currently full professor at University of Lübeck, Germany; visited the team on invation by ENS Cachan from June 28 through July 8.

9.2.1.6. Madhavan Mukund

professor at CMI, Chennai, India; visited the team from April 21 to 30 and from July 1 to 11.

9.2.1.7. K. Narayan Kumar

professor at CMI, Chennai, India; visited the team from May 24 to June 12 and from August 26 to September 6.

9.2.1.8. S. Akshay

Post-Doc at Singapore; visited the team from March 9 to April 14 and from June 25 to July 4.

9.2.1.9. Fernando Rosa VELARDO

Associate professor at University of Madrid. visited *MExICo* on two occasions, from January 5 to February 12 and from July 19 through 23.

9.2.1.10. Marc ZEITOUN

PR University Bordeaux 1; Marc has extended his stay with MExIcO for a second year with INRIA (*délégation INRIA*), from Sept 1st 2010 to August 31, 2011). His main research interests concern the verification of distributed systems, with a focus on the control and synthesis problems. He will also be involved in the diagnosis for concurrent systems. Finally, he is interested in developing tools and specification languages adapted to checking quantitative aspects of systems.

9.2.1.11. Christian KERN

PhD student under supervision of Javier Esparza at the Technical University of Munich; visited the group for two months (May/June). He cooperated with Stefan Schwoon and Stefan Haar on making the reveals relation for Petri nets, first presented in [17], efficiently computable.

9.2.1.12. César Rodríguez

Enrolled in the MPRI program; did his Master thesis with Stefan Schwoon on the subject of algorithmics and implementation of a contextual Petri net unfolder [40]. He continues as a PhD student on this subject area. For the time being, he is funded by a grant from Caja Madrid, limited to one year.

9.2.1.13. Naman AGARWAL

Second-year student at the IIT Bombay; he came through the INRIA internship program to do an *ARCUS*-financed research internship with Stefan Schwoon from May to July 2010. The work focused on improving the efficiency of reachability algorithms for pushdown systems.

9.2.1.14. Anoopam AGARWAL

Second-year student at the IIT Delhi; he came through the INRIA internship program to do an *ARCUS*-financed research internship with Stefan Haar from May to July 2010. The subject of his work was the verification of weak diagnosability using Petri net unfoldings.

9.2.1.15. Hernàn PONCE DE LEON

Master's student of University of Rosario, Argentina; he performed an INRIA internship with Stefan Haar from April to July 2010, on asynchronous conformance testing in concurrent systems modeled by safe Petri nets.

9.2.2. Visits to other laboratories

Stefan Haar visited IIT Rajasthan, Jodhpur, for a one-week workshop (Sept 13-17) on ICT in preparation of the creation of a Center of Excellence at IITR, and Hamburg University in April 2010 for a talk in the *Informatik-Kolloquium*. *Stefan Schwoon* visited the group of Theoretical Computer Science (Barbara König, June 30–July 2) at Duisburg-Essen in the course of cooperation on contextual unfoldings.

9.3. Teaching

- **Sandie Balaguer** teaches OS programming at Paris 7 university.
- **Thomas Chatain** is maître de conférences at ENS Cachan. He teaches complexity, logic, graph theory, network programming, algorithms and C++ programming. He also prepares for the modélisation option in the *agrégation* program at ENS Cachan.
- **Aiswarya Cyriac** is a teaching assistant at the Department of Computer Science of ENS Cachan. She currently teaches the tutorial sessions of the course Algorithms and the lab sessions of Programming Lab.
- **Paul Gastin** is a professor and head of the computer science department at ENS Cachan. He was also head of the Parisian Master of Research in Computer Science until August 2009. He teaches courses on algorithms and on formal languages in the *Licence d'informatique* (L3), on basics of verification in the *Parisian Master of Research in Computer Science* (M1), and several topics in the *agrégation* program at ENS Cachan.
- **Stefan Haar** teaches - with Serge Haddad and Benjamin Monmege - course on Algorithms
- **Serge Haddad** is a professor at ENS Cachan. He currently teaches courses on Complexity and Advanced Algorithmics in the *Licence d'informatique* (L3), a course in the program on preparation for *agrégation* (with **Stefan Haar** and **Benjamin Monmege**) a course on Algorithms, and, in MPRI-M2 a course on probabilistic timed systems. a teaching assistant at department of Computer Science in ENS Cachan.
- **Benjamin Monmege** is a teaching assistant at ENS Cachan. He teaches in Preparation to agregation of Mathematics, option Computer Science : Courses/Practical Sessions Algorithmics and Courses/Practical Sessions Formal Languages.
- **Stefan Schwoon** is maître de conférences at ENS Cachan. Stefan Schwoon is maître de conférences at ENS Cachan. He currently teaches a course and tutorial on operating systems at the L3 level of the ENS Cachan, and a course on verification at the M1 level of the MPRI program.
- **Marc Zeitoun** has participated in the entrance examination *(concours d'entrÃ©e en 3Ã¨ annÃ©e, informatique 2)* at ENS Cachan 2010, and has taught a cours on weighted automata in the master's program *MPRI* 2010-11 (C 2-8).

10. 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, p. 82-97 [*DOI :* 10.1007/978-3-540-85361-9_10], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/ABGMN-concur08.pdf.

[2] 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, p. 199–220, Volume 5100 of Lecture Notes in Computer Science.

[3] 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, p. 76-87 [*DOI :* 10.1007/978-3-540-74240-1_8], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BGMN-fct07.pdf.

[4] 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, p. 18-38 [*DOI :* 10.1007/978-3-642-02737-6_2], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BG-dlt09.pdf.

[5] 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, p. 73-107 [*DOI :* 10.1016/J.IC.2007.10.004], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BHR-ic07.pdf.

[6] 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, p. 202-220 [*DOI :* 10.1016/J.TCS.2008.03.030], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHLR-tcs08.pdf.

[7] 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, p. 141-152 [*DOI :* 10.1007/978-3-540-95891-8_16], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/CGS-sofsem09.pdf.

[8] M. DROSTE, P. GASTIN. *Weighted automata and weighted logics*, in "Theoretical Computer Science", June 2007, vol. 380, n° 1-2, p. 69-86 [*DOI :* 10.1016/J.TCS.2007.02.055], http://www.lsv.ens-cachan.fr/Publis/RAPPORTS_LSV/PDF/rr-lsv-2005-13.pdf.

[9] E. 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), p. 33-84, Preliminary version: Proc. CONCUR 2003, LNCS 2761, pp.1–28, Springer..

[10] 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, p. 321-332 [*DOI :* 10.1007/11944836_30], http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/GSZ-fsttcs2006.pdf.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[11] S. AKSHAY. *Spécification et vérification pour des systèmes distribués et temporisés*, Laboratoire Spécification et Vérification, ENS Cachan, France, July 2010.

Articles in International Peer-Reviewed Journal

[12] 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, p. 1169-1192, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHK-icomp10.pdf.

[13] B. BOLLIG, J.-P. KATOEN, C. KERN, M. LEUCKER. *Learning Communicating Automata from MSCs*, in "IEEE Transactions on Software Engineering", May/June 2010, vol. 36, n° 3, p. 390-408, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BKKL-tse09.pdf.

[14] B. BOLLIG, J.-P. KATOEN, C. KERN, M. LEUCKER. *SMA—The Smyle Modeling Approach*, in "Computing and Informatics", 2010, vol. 29, n° 1, p. 45-72, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BKKL-cai09.pdf.

[15] 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.

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

[17] 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, p. 2310-2320, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/haar-tac10.pdf.

[18] S. LI, S. BALAGUER, A. DAVID, K. LARSEN, B. NIELSEN, S. PUSINSKAS. *Scenario-based verification of real-time systems using Uppaal*, in "Formal Methods in System Design", 2010, p. 1-65, 10.1007/s10703-010-0103-z, http://dx.doi.org/10.1007/s10703-010-0103-z.

[19] 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, p. 235-256.

International Peer-Reviewed Conference/Proceedings

[20] S. AKSHAY, P. GASTIN, M. MUKUND, K. NARAYAN KUMAR. *Model checking time-constrained scenario-based specifications*, in "Proceedings of the 30th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'10)", Chennai, India, K. LODAYA, M. MAHAJAN (editors), Leibniz International Proceedings in Informatics, Leibniz-Zentrum für Informatik, December 2010, To appear, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/AGMN-fsttcs10.pdf.

[21] S. BALAGUER, TH. CHATAIN, S. HAAR. *A Concurrency-Preserving Translation from Time Petri Nets to Networks of Timed Automata*, in "Proceedings of the 17th International Symposium on Temporal Representation and Reasoning (TIME'10)", Paris, France, N. MARKEY, J. WIJSEN (editors), IEEE Computer Society Press, September 2010, p. 77-84, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCH-time10.pdf.

[22] P. BALDAN, A. BRUNI, A. CORRADINI, B. KÖNIG, S. SCHWOON. *On the Computation of McMillan's Prefix for Contextual Nets and Graph Grammars*, in "Proceedings of the 5th International Conference on Graph Transformation (ICGT'10)", Enschede, The Netherlands, H. EHRIG, A. RENSINK, G. ROZENBERG, A. SCHÜRR (editors), Lecture Notes in Computer Science, Springer, 2010, vol. 6372, p. 91–106.

[23] M. BEN HMIDA, S. HADDAD. *Client Synthesis for Aspect Oriented Web Services*, in "Revised Selected Papers of the 15th Monterey Workshop on Foundations of Computer Software (MONTEREY'08)", Budapest, Hungary, CH. CHOPPY, O. SOKOLSKY (editors), Lecture Notes in Computer Science, Springer, April 2010, vol. 6028, p. 24-42, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BH-monterey08.pdf.

[24] 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, p. 587-598, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BGMZ-icalp10.pdf.

[25] B. BOLLIG, L. HÉLOUËT. *Realizability of Dynamic MSC Languages*, in "Proceedings of the 5th International Computer Science Symposium in Russia (CSR'10)", Kazan, Russia, E. W. MAYR (editor), Lecture Notes in Computer Science, Springer, June 2010, vol. 6072, p. 48-59, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BH-csr10.pdf.

[26] B. BOLLIG, J.-P. KATOEN, C. KERN, M. LEUCKER, D. NEIDER, D. R. PIEGDON. *libalf: the Automata Learning Framework*, in "Proceedings of the 22nd International Conference on Computer Aided Verification (CAV'10)", Edinburgh, Scotland, UK, B. COOK, P. JACKSON, T. TOUILI (editors), Lecture Notes in Computer Science, Springer, July 2010, vol. 6174, p. 360-364, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BKKLNP-cav10.pdf.

[27] A. BUŠIĆ, H. DJAFRI, J.-M. FOURNEAU. *Stochastic Bounds for Censored Markov Chains*, in "Proceedings of the 6th International Meeting on the Numerical Solution of Markov Chains (NSMC'10)", Williamsburg, Virginia, USA, M. BENZI, T. DAYAR (editors), sep 2010.

[28] B. BÉRARD, S. HADDAD, M. SASSOLAS. *Real Time Properties for Interrupt Timed Automata*, in "Proceedings of the 17th International Symposium on Temporal Representation and Reasoning (TIME'10)", Paris, France, N. MARKEY, J. WIJSEN (editors), IEEE Computer Society Press, September 2010, p. 69-76, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BHS-time10.pdf.

[29] T. CHATAIN, E. FABRE. *Factorization Properties of Symbolic Unfoldings of Colored Petri Nets*, in "Proceedings of the 31st International Conference on Applications and Theory of Petri Nets (ICATPN'10)", Braga, Portugal, J. LILIUS, W. PENCZEK (editors), Lecture Notes in Computer Science, Springer, June 2010, vol. 6128, p. 165-184, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/CF-pn10.pdf.

[30] TH. CHATAIN, C. JARD. *Sémantique concurrente symbolique des réseaux de Petri saufs et dépliages finis des réseaux temporels*, in "Actes de la 10ème Conférence Internationale sur les NOuvelles TEchnologies de la RÉpartition (NOTERE'10)", Tozeur, Tunisia, IEEE Computer Society Press, May-June 2010, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/CJ-notere10.pdf.

[31] 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.

[32] S. HAAR. *What Topology Tells us about Diagnosability in Partial Order Semantics*, in "Proceedings of the 10th Workshop on Discrete Event Systems (WODES'10)", Berlin, Germany, August-September 2010, p. 221-226, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/SH-wodes10.pdf.

[33] S. HADDAD, L. MOKDAD, S. YOUCEF. *Response time of BPEL4WS constructors*, in "Proceedings of the 15th IEEE Symposium on Computers and Communications (ISCC'10)", Riccione, Italy, IEEE Computer Society Press, June 2010, p. 695-700, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/HMY-iscc10.pdf.

[34] S. HADDAD, L. MOKDAD, S. YOUCEF. *Selection of the Best composite Web Service Based on Quality of Service*, in "Proceedings of the 2nd International Symposium on Services Science and 3rd International Conference on Business Process and Services Computing (ISSS/BPSC'10)", Leipzig, Germany, W. ABRAMOWICZ, R. ALT, K.-P. FÄHNRICH, B. FRANCZYK, L. A. MACIASZEK (editors), Lecture Notes in Informatics, Gesellschaft für Informatik, September-October 2010, vol. 177, p. 255-266, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/hmy-bpsc10.pdf.

Scientific Books (or Scientific Book chapters)

[35] S. DEMRI, P. GASTIN. *Specification and Verification using Temporal Logics*, in "Modern applications of automata theory", D. D'SOUZA, P. SHANKAR (editors), IISc Research Monographs, World Scientific, 2010, vol. 2, To appear, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/DG-iis09.pdf.

Books or Proceedings Editing

[36] P. GASTIN, F. LAROUSSINIE (editors). *Proceedings of the 21st International Conference on Concurrency Theory (CONCUR'10)*, Lecture Notes in Computer Science, Springer, August-September 2010, vol. 6269, http://www.springerlink.com/content/978-3-642-15374-7.

Research Reports

[37] B. BÉRARD, S. HADDAD, M. SASSOLAS, M. ZEITOUN. *Distributed Synthesis with Incomparable Information*, Laboratoire Spécification et Vérification, ENS Cachan, France, October 2010, n^{o} LSV-10-17, 20 pages, http://www.lsv.ens-cachan.fr/Publis/RAPPORTS_LSV/PDF/rr-lsv-2010-17.pdf.

[38] P. GASTIN, N. SZNAJDER. *Decidability of well-connectedness for distributed synthesis*, Laboratoire Spécification et Vérification, ENS Cachan, France, February 2010, n^{o} LSV-10-02, 15 pages, http://www.lsv.ens-cachan.fr/Publis/RAPPORTS_LSV/PDF/rr-lsv-2010-02.pdf.

Other Publications

[39] B. MONMEGE. *Propriétés quantitatives des mots et des arbres – Applications aux langages XML*, Master Parisien de Recherche en Informatique, Paris, France, September 2010, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/monmege-m2.pdf.

[40] C. RODRÍGUEZ. *Implementation of a complete prefix unfolder for contextual nets*, Master Parisien de Recherche en Informatique, Paris, France, September 2010, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/cr-m2.pdf.

References in notes

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

[42] 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.

[43] 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, p. 82-97, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/ABGMN-concur08.pdf.

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

[45] P. BALDAN, TH. CHATAIN, S. HAAR, B. KOENIG. *Unfolding-based Diagnosis of Systems with an Evolving Topology*, 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, p. 203-217, Extended version accepted for publication in: Information and Computation, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHK-concur08.pdf.

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

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

[48] 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, p. 76-87, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BGMN-fct07.pdf.

[49] R. BISHOP. *Intelligent Vehicle R&D: a review and contrast of programs worldwide and emerging trends.*, in "Annals of Telecommunications - Intelligent Transportation Systems", J. EHRLICH (editor), GET-Lavoisier, March-April 2005, vol. 60(3–4), p. 228–263.

[50] J.-M. BLOSSEVILLE. *Driving assistance systems and road safety: State-of-the-art and outlook*, in "Annals of Telecommunications - Intelligent Transportation Systems", J. EHRLICH (editor), GET-Lavoisier, March-April 2005, vol. 60(3–4), p. 281–298.

[51] P. BLY. *e-Safety - Co-operative Systems for Road Transport (IST Work Progamme 2005-2006)*, European Commission, 2004.

[52] 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, p. 169-183.

[53] P. BOUYER, S. HADDAD, P.-A. REYNIER. *Extended Timed Automata and Time Petri Nets*, in "Proceedings of the 6th International Conference on Application of Concurrency to System Design (ACSD'06)", Turku, Finland, K. GOOSSENS, L. PETRUCCI (editors), IEEE Computer Society Press, June 2006, p. 91-100, http://www.lsv.ens-cachan.fr/Publis/RAPPORTS_LSV/PDF/rr-lsv-2006-01.pdf.

[54] P. BOUYER, S. HADDAD, P.-A. REYNIER. *Timed Petri Nets and Timed Automata: On the Discriminating Power of Zeno Sequences*, in "Proceedings of the 33rd International Colloquium on Automata, Languages and

Programming (ICALP'06) — Part II", Venice, Italy, M. BUGLESI, B. PRENEEL, V. SASSONE, I. WEGENER (editors), Lecture Notes in Computer Science, Springer, July 2006, vol. 4052, p. 420-431, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BHR-icalp06.pdf.

[55] 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, p. 292-306, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BHR-atva06.pdf.

[56] M. BOYER, M. DIAZ. *Non equivalence between time Petri nets and time stream Petri nets*, in "PNPM", IEEE Computer Society, 1999, p. 198-207.

[57] M. BOYER, O. H. ROUX. *Comparison of the Expressiveness of Arc, Place and Transition Time Petri Nets*, in "ICATPN", J. KLEIJN, A. YAKOVLEV (editors), Lecture Notes in Computer Science, Springer, 2007, vol. 4546, p. 63-82.

[58] B. BÉRARD, F. CASSEZ, S. HADDAD, D. LIME, O. H. ROUX. *Comparison of Different Semantics for Time Petri Nets*, in "Proceedings of the 3rd International Symposium on Automated Technology for Verification and Analysis (ATVA'05)", Taipei, Taiwan, ROC, D. A. PELED, Y.-K. TSAY (editors), Lecture Notes in Computer Science, Springer, October 2005, vol. 3707, p. 293-307, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHLR05-atva.pdf.

[59] B. BÉRARD, F. CASSEZ, S. HADDAD, D. LIME, O. H. ROUX. *Comparison of the Expressiveness of Timed Automata and Time Petri Nets*, in "Proceedings of the 3rd International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS'05)", Uppsala, Sweden, P. PETTERSSON, W. YI (editors), Lecture Notes in Computer Science, Springer, November 2005, vol. 3829, p. 211-225, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHLR-formats2005.pdf.

[60] B. BÉRARD, F. CASSEZ, S. HADDAD, D. LIME, O. H. ROUX. *When are Timed Automata Weakly Timed Bisimilar to Time Petri Nets?*, in "Proceedings of the 25th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'05)", Hyderabad, India, R. RAMANUJAM, S. SEN (editors), Lecture Notes in Computer Science, Springer, December 2005, vol. 3821, p. 273-284, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BCHLR-fsttcs05.pdf.

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

[62] 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, p. 33–86.

[63] E. 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), p. 714-727.

[64] E. 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), p. 33-84, Preliminary version: Proc. CONCUR 2003, LNCS 2761, pp.1–28, Springer..

[65] 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, p. 321–330.

[66] 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, p. 275-286, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/GLZ-fsttcs04.pdf.

[67] 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, p. 321-332, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/GSZ-fsttcs2006.pdf.

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

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

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

[71] S. HADDAD, P. POIZAT. *Transactional Reduction of Component Compositions*, in "Proceedings of 27th IFIP WG6.1 International Conference on Formal Techniques for Networked and Distributed Systems (FORTE'07)", Tallinn, Estonia, J. DERRICK, J. VAIN (editors), Lecture Notes in Computer Science, Springer, June 2007, vol. 4574, p. 341-357, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/HP-forte07.pdf.

[72] R. HOROWITZ, P. VARAIYA. *Control Design of an Automated Highway System*, in "Proc. of the IEEE 88(7)", 2000.

[73] INTELLIGENT VEHICLE INITIATIVE. *Saving Lives through advanced vehicle safety technology*, September 2005.

[74] 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.

[75] D. KÖNIG, N. LOHMANN, S. MOSER, C. STAHL, K. WOLF. *Extending the compatibility notion for abstract WS-BPEL processes*, in "WWW", ACM, 2008, p. 785-794.

[76] S. LAFORTUNE, Y. 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, p. 95–110.

[77] 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, p. 76-89.

[78] 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, p. 338–351.

[79] 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, p. 746–757.

[80] W. REISIG. *Towards a Theory of Services*, in "UNISCON", Lecture Notes in Business Information Processing, Springer, 2008, vol. 5, p. 271-281.

[81] 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, p. 1656–1668.

[82] 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, p. 428–441.

[83] Y. ROBIN-JOUAN, J. EHRLICH, B. GUILLAUMIN, M. DELARCHE, M. DUTECH. *Transport-specific communication services: Safety-based or critical applications for mobiles and cooperation with infrastructure networks*, in "Annals of Telecommunications - Intelligent Transportation Systems", J. EHRLICH (editor), GET-Lavoisier, March-April 2005, vol. 60(3–4), p. 405–440.

[84] J. SRBA. *Timed-Arc Petri Nets vs. Networks of Timed Automata*, in "ICATPN", Lecture Notes in Computer Science, Springer, 2005, vol. 3536, p. 385-402.

[85] S. STAAB, W. VAN DER AALST, V. BENJAMINS, A. SHETH, J. MILLER, C. BUSSLER, A. MAEDCHE, D. FENSEL, D. GANNON. *Web Services: Been There, Done That?*, in "IEEE Intelligent Systems", 2003, vol. 18, p. 72-85.

[86] M. B. VAN RIEMSDIJK, R. HENNICKER, M. WIRSING, A. SCHROEDER. *Service Specification and Matchmaking Using Description Logic*, in "AMAST", Lecture Notes in Computer Science, Springer, 2008, vol. 5140, p. 392-406.