Activity Report 2012

Project-Team DISTRIBCOM

Distributed and Iterative Algorithms for the Management of Telecommunications Systems

IN COLLABORATION WITH: Institut de recherche en informatique et systèmes aléatoires (IRISA)

Table of contents

Project-Team DISTRIBCOM

Keywords: Quality Of Service, Distributed Algorithms, Monitoring, Optical Networks, Service Oriented Architecture, Workflow

Creation of the Project-Team: May 03, 2004 .

1. Members

Research Scientists
  Albert Benveniste [Team Leader, DR INRIA, part–time, HdR]
  Eric Fabre [DR INRIA , HdR]
  Loïc Hélouët [CR INRIA]
  Blaise Genest [CR CNRS 15% with DistribCom, 85% with UMI IPAL, Singapore]
  Axel Legay [CR INRIA]

Faculty Members
  Claude Jard [PROF. ENS CACHAN, until August 31st, HdR]
  Guillaume Aucher [CHAIRE MAÎTRE DE CONF. UNIV. RENNES I – INRIA]
  François Schwarzentruber [MAÎTRE DE CONF. ENS CACHAN]

Engineers
  Benoit Boyer [Inria grant]
  Kevin Corre [Inria grant]

PhD Students
  Ajay Kattepur [Inria grant]
  Loig Jézéquel [ENS Cachan grant]
  Rouwaida Abdallah [ENS Cachan grant]
  Cyrille Jégourel [Univ. Rennes I grant]
  Carole Hounkonnou [Inria grant]
  Aurore Junier [Inria grant]

Post-Doctoral Fellows
  Sundararaman Akshay [ENS Cachan grant]
  Benoit Delahaye [Inria grant]
  Sean Sedwards [Inria grant]
  Ulrich Fahrenberg [Inria grant]
  Louis-Marie Traonouez [Inria grant]

Administrative Assistant
  Laurence Dinh

2. Overall Objectives

2.1. Objectives of the team

telecommunications, self-management, distributed algorithms, fault management, distributed testing, web services, orchestrations, quality of service

The DistribCom team is jointly headed by Albert Benveniste (official head for Inria) and Claude Jard. It addresses models and algorithms for the distributed management of networks, services, Web services and business processes.

Today, research on network and service management as well as Web Services mainly focuses on issues of software architecture and infrastructure deployment. However, these areas also involve algorithmic problems such as fault diagnosis and alarm correlation, testing, QoS evaluation, negotiation, and monitoring. The DistribCom team develops the foundations supporting such algorithms. Our algorithms are model-based. Our research topics are therefore structured as follows:

1. *Fundamentals of distributed observation and supervision of concurrent systems*: this provides the foundations for deriving models and algorithms for the above mentioned tasks.
2. *Self-modeling*: for obvious reasons of complexity, our models cannot be built by hand. We thus address the new topic of self-modeling, i.e., the automatic construction of models, both structural and behavioral.
3. *Algorithms for distributed management of telecommunications systems and services.*
4. *Web Services orchestrations, functional and QoS aspects.*
5. *Active XML peers for Web scale data and workflow management.*

Our main industrial ties are with Alcatel-Lucent, and France-Telecom, on the topic of networks and service management.

Inria, Centre of Rennes-Bretagne-Atlantique, decided that Axel Legay and his group of post-docs and PhDs would remain hosted by DistribCom. The activities of Axel's group are specifically reported in Sections 3.4, 6.5, and 6.6.

3. Scientific Foundations

3.1. Overview of the needed paradigms

Management of telecommunications networks and services, and Web services, involves the following algorithmic tasks:

Observing, monitoring, and testing large distributed systems: Alarm or message correlation is one of the five basic tasks in network and service management. It consists in causally relating the various alarms collected throughout the considered infrastructure—be it a network or a service sitting on top of a transport infrastructure. Fault management requires in particular reconstructing the set of all state histories that can explain a given log of observations. Testing amounts to understanding and analyzing the responses of a network or service to a given set of stimuli; stimuli are generally selected according to given test purposes. All these are variants of the general problem of *observing* a network or service. Networks and services are large distributed systems, and we aim at observing them in a distributed way as well, namely: logs are collected in a distributed way and observation is performed by a distributed set of supervising peers.

Quality of Service (QoS) evaluation, negotiation, and monitoring: QoS issues are a well established topic for single domain networks or services, for various protocols — e.g., Diffserv for IP. Performance evaluation techniques are used that follow a "closed world" point of view: the modeling involves the overall traffic, and resource characteristics are assumed known. These approaches extend to some telecommunication services as well, e.g., when considering (G)MPLS over an IP network layer.

However, for higher level applications, including composite Web services (also called *orchestrations*), this approach to QoS is no longer valid. For instance, an orchestration using other Web services has no knowledge of how many users are calling the same Web services. In addition, it has no knowledge of the transport resources it is using. Therefore, the well developed "closed world" approach can no longer be used. *Contract-based* approaches are considered instead, in which a given orchestration offers promises to its users on the basis of promises it has from its subcontracting services. In this context, contract composition becomes a central issue. Monitoring is needed to check for possible breaching of the contract. Countermeasures would consist in reconfigurating the orchestration by replacing the failed subcontracted services by alternative ones.

The DistribCom team focuses on the algorithms supporting the above tasks. Therefore models providing an adequate framework are fundamental. We focus on models of discrete systems, not models of streams or fluid types of models. And we address the distributed and asynchronous nature of the underlying systems by using models involving only local, not global, states, and local, not global, time. These models are reviewed in section 3.2. We use these mathematical models to support our algorithms and we use them also to study and develop formalisms of Web services orchestrations and workflow management in a more general setting.

3.2. Models of concurrency: nets, scenarios, event structures, and their variants

For Finite State Machines (FSM), a large body of theory has been developed to address problems such as: observation (the inference of hidden state trajectories from incomplete observations), control, diagnosis, and learning. These are difficult problems, even for simple models such as FSM's. One of the research tracks of DistribCom consists in extending such theories to distributed systems involving concurrency, i.e., systems in which both time and states are local, not global. For such systems, even very basic concepts such as "trajectories" or "executions" need to be deeply revisited. Computer scientists have for a long time recognized this topic of concurrent and distributed systems as a central one. In this section, we briefly introduce the reader to the models of scenarios, event structures, nets, languages of scenarios, graph grammars, and their variants.

3.2.1. *Scenarios.*

The simplest concept related to concurrency is that of a finite execution of a distributed machine. To this end, scenarios have been informally used by telecom engineers for a long time. In scenarios, so-called "instances" exchange asynchronous messages, thus creating events that are totally ordered on a given instance, and only partially ordered by causality on different instances (emission and reception of a message are causally related). The formalization of scenarios was introduced by the work done in the framework of ITU and OMG on High-level Message Sequence Charts and on UML Sequence Diagrams in the last ten years, see [52], [57]. This allowed in particular to formally define infinite scenarios, and to enhance them with variables, guards, etc [61], [59], [60]. Today, scenarios are routinely offered by UML and related software modeling tools.

3.2.2. *Event structures.*

Event structures were invented by Glynn Winskel and co-authors in 1980 [56], [62]. Executions are sets of events that are partially ordered by a *causality* relation. Event structures collect all the executions by superimposing shared prefixes. Events not belonging to a same execution are said in *conflict*. Events that are neither causally related nor in conflict are called *concurrent*. Concurrent processes model the "parallel progress" of components.

Categories of event structures have been defined, with associated morphisms, products, and co-products, see [63]. Products and co-products formalize the concepts of parallel composition and "union" of event structures, respectively. This provides the needed apparatus for composing and projecting (or abstracting) systems. Event structures have been mostly used to give the semantics of various formalisms or languages, such as Petri nets, CCS, CSP, etc [56], [62]. We in DistribCom make a nonstandard use of these, e.g., we use them as a structure to compute and express the solutions of observation or diagnosis problems, for concurrent systems.

3.2.3. *Nets and languages of scenarios.*

The next step is to have finite representations of systems having possibly infinite executions. In DistribCom, we use two such formalisms: *Petri nets* [58], [45] and *languages of scenarios* such as High-level Message Sequence Charts (HMSC) [52], [60]. Petri nets are well known, at least in their basic form, we do not introduce them here. We use so-called *safe* Petri Nets, in which markings are boolean (tokens can be either 0 or 1); and we use also variants, see below.

3.2.4. Extensions and variants.

Two extensions of the basic concepts of nets or scenario languages are useful for us. Nets or scenario languages enriched with variables, actions, and guards, are useful to model general concurrent and distributed dynamical systems in which a certain discrete abstraction of the control is represented by means of a net or a scenario language. Manipulating such *symbolic nets* requires using abstraction techniques. Time Petri nets and network of timed automata are particular cases of symbolic nets. Probabilistic Nets or event structures: Whereas a huge literature exists on stochastic Petri nets or stochastic process algebras (in computer science), randomizing *concurrent models,* i.e., with ω's being concurrent trajectories, not sequential ones, has been addressed only since the 21st century. We have contributed to this new area of research.

3.2.5. Handling dynamic changes in the systems.

The last and perhaps most important issue, for our applications, is the handling of dynamic changes in the systems model. This is motivated by the constant use of dynamic reconfigurations in management systems. Extensions of net models have been proposed to capture this, for example the *dynamic nets* of Vladimiro Sassone [44] and *net systems* [46]. For the moment, such models lack a suitable theory of unfoldings.

3.3. Modal logics for distributed systems

Modal logics are a family of logics that were developed originally to reason about different modalities occurring in natural language, such as for example the modality of knowledge (epistemic logic), the modalities of obligation and permission (deontic logic) and the modality of time (temporal logic). Temporal logics (CTL, LTL, μ-calculus...) are the most prominent (modal) logics used in computer science nowadays, especially in the field of verification.

3.3.1. Epistemic logic and distributed systems.

In the 1980's, epistemic logic was propounded by computer scientists such as Fagin, Halpern, Moses and Vardi to address problems in distributed systems, resulting in the TARK conference series (Theoretical Aspects of Rationality and Knowledge) and the books [48], [54]. This interest in epistemic logic was due to their observation that the notion of knowledge plays a central role in the informal reasoning used in the design of distributed protocols. This lead these authors to "hope that a theory of knowledge, communication and action will prove rich enough to provide general foundations for a unified theoretical treatment of distributed systems" [50]. The research pursued in DistribCom follows this line of thought, although we also strive to feed and confront our theoretical developments with actual problems stemming from diverse areas of application of distributed systems.

In [48], the behavior of a distributed system is represented by a set of *runs*, each run being a possible execution of the distributed system, determined by a given protocol. Processors are called *agents* and their partial observation of the system is represented at any point in the run by indistinguishability relations between local states of different runs (the local state of a processor represents the state of this processor at a moment of time). This model was used to show for example that the specific notion of common knowledge of epistemic logic is necessary to reach agreement and to coordinate actions [50]. Dynamic Epistemic Logic (DEL) is another logical framework that can be used to represent and reason about distributed systems (connections between these two logical frameworks were made in [64]). DEL deals with the representation of global states of synchronous distributed systems. The global state of the system at a moment in time is represented directly by means of an *epistemic model*. Events occurring in this distributed system are represented by means of *event models* and their effects on the local states of agents (processors) are represented by means of a *product update*.

The contributions in this sub-module are described in Section 6.4.

3.3.2. Deontic logic and privacy in distributed systems.

We also use deontic logic in combination with epistemic logic for the formalization of privacy regulations. We intend to use this formalization to reason about privacy in the composition of web-services. The combination of these two modal logics can be used to express statements such as "it is *forbidden* for agent 1 to *know*

that agent 2 sent message m" or "if agent 1 is an administrator of the system, then it is *permitted* for him to *know* information i". This provides a formal language very close to the natural language used in actual privacy regulations by law legislators. In the long run, we expect this formal language to be used at the level of interfaces of the web-service in order to:

1. check that the privacy policy declared by the web-service on its interface is indeed compliant (coherent) with respect to the privacy regulations expressed by law makers;
2. check that the web-service does enforce and apply the privacy policy it has declared on its interface.

The contributions in this sub-module are described in Section 6.8.

3.4. Statistical Model Checking

Complex systems pose two particular challenges to formal verification: (i) the non-determinism caused by concurrency and unpredictable environmental conditions and (ii) the size of the state space. Our interest is probabilistic model checking, that can verify intricate details of a system's dynamical behavior and where non-determinism is handled by assigning probabilistic distributions to unknowns and quantifying results with a probability. Exact probabilistic model checking quantifies these probabilities to the limit of numerical precision by an exhaustive exploration of the state space, but is restricted by what can be conveniently stored in memory. Our focus is therefore statistical model checking (SMC), that avoids an explicit representation of the state space by building a statistical model of the executions of a system and giving results within confidence bounds. The key challenges of this approach are to reduce the length (simulation steps and cpu time) and number of simulation traces necessary to achieve a result with given confidence. Rare properties pose a particular problem in this respect, since they are not only difficult to observe but their probability is difficult to bound. A further goal is to make a tool where the choice of modeling language and logic are flexible.

4. Application Domains

4.1. Telecommunication network management

The management of telecommunication networks is traditionally a human performed activity that covers the five FCAPS functions: Fault management, network Configuration, Accounting, Performances and Security. This simple classification has exploded in the last decade, under the pressure of several phenomena. The first one concerns the growth in size and complexity of networks, with the emergence of new (possibly virtual) operators, the multiplication of vendors, new core and (wireless) access technologies, the variety of terminal devices, the convergence of phone/computer/radio/TV networks, the multiplication of services over the top, the necessity to provide QoS for a wide variety of traffic demands, etc. As a consequence, the management task is reaching the limits of human operators and demands automation. It is estimated that telecommunication companies spend over 50% of their manpower on management tasks. They naturally want to reduce it and dedicate their effort to the design and offer of innovative services, where the added value is more important (as witnessed by the success of some over-the-top companies). The result of these trends is that network management now covers a much wider variety of problems, for which automatic solutions are requested. This takes the name of self-management, or autonomic management: one wishes to manage networks by high-level objectives, and networks should be able to adapt themselves automatically to fulfill these objectives. DistribCom is contributing to this field with its background on the modeling of distributed/concurrent systems, and its expertise in distributed algorithms. Networks are perfect examples of large distributed and concurrent systems, with specific features like the dynamicity (their structure evolves) and a hierarchical structure (multiple layers, multiple description granularities). We have proposed model-based distributed algorithms to solve problems like failure diagnosis, negotiation of QoS (quality of service) parameters, parameter optimization, graceful shutdown of OSPF routers for maintenance operations... The present activities in this domain are related to the joint diagnosis for access network + core network + services, within the European IP UniverSelf. The challenges cover self-modelling methods (how to obtain the network model that is used by the management algorithms), active diagnosis methods that both adapt the scope of their network model and perform tests to explain a fault situation, and self-healing methods.

4.2. Web services and active structured documents

Keywords: Active documents, Web services, choreographies, orchestrations, QoS.

Web services architectures are usually composed of distant services, assembled in a composite framework. This raises several practical issues: one of them is how to choose services, assemble them, and coordinate their executions in a composite framework. Another issue is to guarantee good properties of a composite framework (safety but also QoS properties). All this has to be done in a context where a distant service provided by a subcontractor is only perceived as an interface, specifying legal inputs and outputs, and possibly a quality contract. The standard in industry for Web-services is now BPEL [43] but most of the problems listed above are untractable for this language. Composition of services can also be performed using choreography languages such as ORC [55]. The implementation of orchestration and choreography description languages raises a number of difficulties related to efficiency, clean semantics, and reproducibility of executions, issues of composite QoS associated with orchestrations. We develop studies in these areas, with the aim of proposing service composition frameworks equipped with tools to specify, but also to monitor and analyze the specified architectures. Another issue is the convergence between data and workflows. Web Services architectures are frequently considered exclusively as workflows, or as information systems. Many approaches to Web Service orchestration and choreography abstract data away. Symmetrically, modern approaches to Web data management typically based on XML and Xqueries rely on too simplistic forms of control. We develop a line of research on Active documents. Active documents are structured data embedding references to services, which allow for the definitions of complex workflows involving data aspects. The original model was proposed by S. Abiteboul (see for instance [42]), but the concept of active document goes beyond AXML, and offers a document oriented alternative to Web services orchestrations and choreographies. This approach is in particular well adapted to the modeling of E-business processes, or information processing in organizations, etc. Our aim is to extend and promote the concept of active document. This means developing verification and composition tools for document-based architectures, considered not only as theoretical models but also as effectively running systems. To this extend, we develop an active document platform.

5. Software

5.1. SOFAT

Participants: Loïc Hélouët [correspondant], Rouwaida Abdallah.

SOFAT is the acronym for Scenario Oracle and Formal Analysis Toolbox. As this name suggests it is a formal analysis toolbox for scenarios. Scenarios are informal descriptions of behaviors of distributed systems. SOFAT allows the edition and analysis of distributed systems specifications described using Message Sequence Charts, a scenario language standardized by the ITU [Z.120]. The main functionalities proposed by SOFAT are the textual edition of Message Sequence Charts, their graphical visualization, the analysis of their formal properties, and their simulation. The analysis of the formal properties of a Message Sequence Chart specification determines if a description is regular, local choice, or globally cooperative. Satisfaction of these properties allow respectively for model-checking of logical formulae in temporal logic, implementation, or comparison of specifications. All these applications are either undecidable problems or unfeasible if the Message Sequence Chart description does not satisfy the corresponding property. The SOFAT toolbox implements most of the theoretical results obtained on Message Sequence Charts this last decade. It is regularly updated and re-distributed. The purpose of this is twofold:

- Provide a scenario based specification tool for developers of distributed applications
- Serve as a platform for theoretical results on scenarios and partial orders

SOFAT provides several functionalities, that are: syntactical analysis of scenario descriptions, Formal analysis of scenario properties, Interactive Simulation of scenarios when possible, and diagnosis. This year, SOFAT was extended with code synthesis functionalities, allowing to generate communicating automata, promela code, or rest based web services from HMSCs. A new release of the software is expected before the end of the year.

See also the web page http://www.irisa.fr/distribcom/Prototypes/SOFAT/index.html.
- AMS: Order; lattices; ordered algebraic structures
- APP: IDDN.FR.001.080027.000.S.P.2003.00.10600
- Programming language: Java

5.2. PLASMA

Participants: Sean Sedwards, Benoit Boyer, Kevin Corre, Axel Legay [correspondant].

PLASMA is our implementation of Statistical Model Checking. PLASMA adopts a modular architecture to facilitate the extension of its features. Models can currently be specified using the PRISM reactive modules syntax or a biochemical syntax, while properties are specified in a discrete bounded temporal logic. Our goal is to allow the implementation of other modeling languages and logics by means of self-contained *drop-in* modules. PLASMA facilitates this by providing an intermediate language to generate transition systems based on the notion of the construct (guard, rate, actions), where guard, rate and actions are functions over the current state of the system and control whether and how fast the system may perform certain actions in each state. New modeling languages may be thus added to PLASMA's repertoire by constructing parsers that translate such languages into the intermediate language.

Web site: https://project.inria.fr/plasma-lab/

5.3. LotrecScheme

Participant: François Schwarzentruber [correspondant].

LotrecScheme is the implementation of a generic tableau method prover based on LoTREC (http://www.irit.fr/Lotrec/). LotrecScheme is more expressive than LoTREC. Both LoTREC and LotrecScheme provides tableau methods for standard modal logic K, KT, S4, etc. Contrary to LoTREC, LotrecScheme is expressive enough to capture some satisfiability problem for Dynamic Epistemic Logic.

The prover inside LotrecScheme is written in Scheme and embedded in a JAVA application.

See also the web page http://www.irisa.fr/distribcom/Prototypes/LotrecScheme/index.html.

6. New Results

6.1. Fundamental results and algorithms: distributed planning

Participants: Eric Fabre, Loig Jézéquel.

A planning problem consists in organizing some actions in order to reach an objective. Formally, this is equivalent to finding a path from an initial state to a goal/marked state in a huge automaton. The latter is specified by a collection of resources, that may be available or not (which defines a state), and actions that consume and produce resources (which defines a transition). In the case of optimal planning, actions have a cost, and the objective is to find a path of minimal cost to the goal.

Our interest in this problem is threefold. First, it is naturally an instance of a concurrent system, given that actions have local effects on resources. Secondly, it is a weak form of an optimal control problem for a concurrent/distributed system. Finally, we are interested in distributed solutions to such problems, which is an active topic in the planning community under the name of "factored planning."

Our previous contribution to the domain was the first optimal factored planning algorithm [47]. The main idea is to represent a planning problem as a network of interacting weighted automata, the objective being to jointly drive all of them to a target state, while minimizing the cost of their joint trajectory. We have developed and tested [53] a distributed algorithm to solve this problem, based on a weighted automata calculus, and that takes the shape of a message passing procedure. Components perform local computations, exchange messages with their neighbors, in an asynchronous manner, and the procedure converges to the path that each component should follow. The optimal global plan is thus given as a tuple of (compatible) local plans, i.e. a partial order of actions.

In 2012, we have extended this framework in two directions. The first one considers large planning problems for which the interaction graph of components is not a tree. It is well known that message passing algorithms (also called belief propagation) is optimal on trees. To recover such a situation where distributed optimal planning can be resolved exactly, one therefore has to smartly group components into larger ones in order to recover a tree of larger components. This is done at the expense of the complexity in the resolution of local planning problems (which augments exponentially with the number of assembled components). Alternately, one can also ignore that the graph is not a tree, and thus use the so-called loopy belief propagation, which requires minor adaptations. This results in a new approach to the resolution of planning problems, where approximate solutions are provided: one can check that the computed plans are valid, but their optimality is not guaranteed. We have experimented this turbo-planning idea on a series of random benchmarks, some of them being not accessible to standard planning methods. The results are surprisingly good: distributed plans are found in most cases, and are often close to optimal. However, no theoretical results can yet support this phenomenon [30].

The second extension to distributed planning concerns the multi-agent version of the central A* (A-star) algorithm, which is at the core of numerous planners. By contrast with the previous setting, we do not build all plans here, in a distributed manner, but perform a search for an optimal plan. The centralized version of A* performs a depth-first search of a winning path in a graph, guided by some heuristic function that orients the search towards the goal. In our setting, several path searches must be performed in the graphs of the different components (or local planning problems), under the constraint that the provided paths are compatible, i.e. agree on the execution of the common actions. The resulting local paths must also be jointly optimal, once their costs are added. We have proposed a complete solution to this problem, called A# (A-sharp) [29]. Our efforts now aim at mixing these ideas with the turbo planning approach.

6.2. Fundamental results and algorithms: communication with messages and scenarios

Participants: Loïc Hélouët, Rouwaida Abdallah, Claude Jard, Blaise Genest, Sundararaman Akshay.

In this paragraph, we collect our fundamental results regarding the models and algorithms we use for communicating systems, and in particular, scenarios.

A major challenge with models communicating with messages (e.g.: scenarios) is to *exhibit good classes of models* allowing users to *specify easily complex distributed systems* while *preserving the decidability* of some key problems, such as diagnosis, equality and intersection. Furthermore, when these problems are decidable for the designed models, the second challenge is to design algorithms to keep the *complexity low enough* to allow *implementation in real cases*.

The first part of our work is the study of Time-Constrained MSC graphs (TC-MSGS for short). Time-constrained MSCs (TC-MSCs) are simply MSCs decorated with constraints on the respective occurrence dates of events. The semantics of a TC-MSC T is a dated MSC, that is a MSC where events are associated with an occurrence date. For a given TC-MSC, there can be an infinite set $L(T)$ of dated MSCs satisfying its constraints. Note however that some time-constraints in a TC-MSC may not be satisfiable, and hence $L(T)$ can simply be empty. TC-MSCs can be extended by composition mechanisms such as TC-MSC graphs. TC-MSC graphs are simply automata labeled by TC-MSC. Each path ρ of a TC-MSC G is associated with a TC-MSC T_ρ obtained by concatenation of TC-MSC along ρ. The language $L(G) = \bigcup_{\rho \text{ path of } G} L(T_\rho)$ of a TC-MSC Graph is then the union of all dated MSCs associated with paths of G. Because of inconsistent timing constraints, some path may have no possible realization (i.e $L(T_\rho = \varnothing)$). One can even design a MSC Graph G such that $L(G) = \varnothing$ - such TC-MSC graph is clearly inconsistent-. It has been shown [49] that checking whether $L(G) = \varnothing$ is an undecidable problem in general, but can be decided for the restricted subclass of regular TC-MSC graphs (that have the expressive power of event-count timed automata). We have proposed two restrictions allowing for the decision of emptiness. The first one is K-drift boundedness, which imposes for a fixed integer K that for every T_ρ there exists one dated realization such that for every pair of events e, f appearing in the same transition of G, the dates of e and f differ by at most K. We have shown that

K-drift boundedness is decidable in a symbolic and efficient way, and that for $K-$drift bounded TC-MSC graphs, emptiness is decidable. This extends decidability results beyond regular specifications. The second restriction is K-non-zenoness, which imposes that for a fixed K, for every path ρ of G, there exists one realization such that at every date d, at most K events occur between dates d and $d + 1$. When a TC-MSC graph is A-drift-bounded and B-non-zeno, then $L(G)$ has a regular set of representants, which opens the way for more involved model-checking applications [10]. We actually succeeded to use a different technique by symbolically encoding the configuration reached. It allows to remove the K-non-zeno restriction, we don't need the seminal result on timed automata of Alur-Dill 1994, and we have a true partial order algorithm, which does not need to consider different interleavings of the same execution [18].

The second part of our work is the study of realistic implementation of scenarios. The main idea is to propose distributed implementation (communicating state machines) of High-level MSCs that do not contain deadlocks, and behave exactly as the original specification. It is well known [51] that a simple projection of a HMSC on each of its processes to obtain communicating finite state machines results in an implementation with more behaviors than the original specification. An implementation of a HMSC H is considered as consistent if and only if it exhibits the same prefix closed set of behaviors as H. We have proposed an implementation solution that uses local controllers allows the distributed synthesized behavior to remain consistent with the original specification. This work has been implemented in our scenario prototype (see the Software section). This synthesis algorithm is consistent for a particular syntactic class of scenarios, namely the class of local HMSCs. This work was accepted for publication in [14].

6.3. Fundamental results and algorithms: timed models

Participants: Claude Jard, Aurore Junier, Sundararaman Akshay, Loïc Hélouët.

Our work on that subject mainly concerns Time Petri Nets (TPNs) and their robustness. Robustness of timed systems aims at studying whether infinitesimal perturbations in clock values can result in new discrete behaviors. A model is robust if the set of discrete behaviors is preserved under arbitrarily small (but positive) perturbations. We have tackled this problem for Time Petri Nets (TPNs for short) by considering the model of parametric guard enlargement which allows time-intervals constraining the firing of transitions in TPNs to be enlarged by a (positive) parameter.

We have shown that TPNs are not robust in general and that checking if they are robust with respect to standard properties (such as boundedness, safety) is undecidable. We have also provided two decidable robustly bounded subclasses of TPNs, and shown that one can effectively build a timed automaton which is timed bisimilar even in presence of perturbations. This allowed us to apply existing results for timed automata to these TPNs and show further robustness properties. This work was published in [20].

In a second work, we have considered robustness issues in Time Petri Nets (TPN) under constraints imposed by an external architecture. Our main objective was to check whether a timed specification, given as a TPN behaves as expected when subject to additional time and scheduling constraints. These constraints are given by another TPN that constrains the specification via read arcs. Our robustness property says that the constrained net does not exhibit new timed or untimed behaviors. We show that this property is not always guaranteed but that checking for it is always decidable in 1-safe TPNs. We further show that checking if the set of untimed behaviors of the constrained and specification nets are the same is also decidable. Next we turn to the more powerful case of labeled 1-safe TPNs with silent transitions. We show that checking for the robustness property is undecidable even when restricted to 1-safe TPNs with injective labeling, and exhibit a sub-class of 1-safe TPNs (with silent transitions) for which robustness is guaranteed by construction. This sub-class already lies close to the frontiers of intractability. This work was published in [19].

Finally, in cooperation with IRCCyN in Nantes, we defined a more general model, called "clock transition systems", which generalizes both TPNs and networks of timed automata [32]. This model will allow us to transfer new results on TPNs to the timed automata community.

6.4. Fundamental results and algorithms: dynamic epistemic logic

Participants: Guillaume Aucher, François Schwarzentruber.

Within the research line related to Dynamic Epistemic Logic (DEL), we have addressed two parallel lines of research, which have resulted in two publications [22] and [21]. The first deals with the computational complexity of the model checking problem and the satisfiability problem of DEL and the second deals with providing formal means to reason about the effects of sequences of events on the beliefs of multiple agents when these events are only partially specified. This second line of research is a continuation of the work started last year and was motivated by concerns and problems stemming from the Univerself project of Eric Fabre about IMS network.

1. Although DEL is an influential logical framework for representing and reasoning about information change, little is known about the computational complexity of its associated decision problems. In fact, we only know that for public announcement logic, a fragment of DEL, the satisfiability problem and the model-checking problem are respectively PSPACE-complete and in P. We contributed to fill this gap by proving that for the DEL language with event models, the model-checking problem is, surprisingly, PSPACE-complete. Also, we proved that the satisfiability problem is NEXPTIME-complete. In doing so, we provided a sound and complete tableau method deciding the satisfiability problem.

2. Let us consider a sequence of formulas providing partial information about an initial situation, about a set of events occurring sequentially in this situation, and about the resulting situation after the occurrence of each event. From this whole sequence, we want to infer more information, either about the initial situation, or about one of the events, or about the resulting situation after one of the events. Within the framework of Dynamic Epistemic Logic, we show that these different kinds of problems are all reducible to the problem of inferring what holds in the final situation after the occurrence of all the events. We then provide a tableau method deciding whether this kind of inference is valid. We implement it in LotrecScheme and show that these inference problems are NEXPTIME-complete. We extend our results to the cases where the accessibility relation is serial and reflexive and illustrate them with the coordinated attack problem.

Parallely to the study of abstract dynamic epistemic logic, we initiate the study of the interaction of argumentation theory and epistemic reasoning [33].

6.5. Fundamental results and algorithms: statistical model checking

Participants: Sean Sedwards, Benoit Boyer, Kevin Corre, Cyrille Jégourel, Axel Legay.

Our work on statistical model checking (SMC) avoids an explicit representation of the state space by building a statistical model of the executions of a system and giving results within confidence bounds. The key challenges of this approach are to reduce the length (simulation steps and cpu time) and number of simulation traces necessary to achieve a result with given confidence. Rare properties pose a particular problem in this respect, since they are not only difficult to observe but their probability is difficult to bound. A further goal is to make a tool where the choice of modeling language and logic are flexible.

We have developed the prototype of a compact, modular and efficient SMC platform which we have named *PLASMA* (PLatform for Statistical Model checking Algorithms). PLASMA incorporates an efficient discrete event simulation algorithm and features an importance sampling engine that can reduce the necessary number of simulation runs when properties are rare. We have found that PLASMA performs significantly better than PRISM (the de facto reference probabilistic model checker) when used in a similar mode: PLASMA's simulation algorithm scales with a lower order and can handle much larger models. When using importance sampling, PLASMA's performance with rare properties is even better.

Plasma has been embedded in a tool chain for the design and the verification of Systems of Systems. The tool has also been used in a planing algorithm.

6.6. Fundamental results and algorithms: quantitative model checking and quantitative specification theories

Participants: Ulrich Fahrenberg, Blaise Genest, Axel Legay, Sundararaman Akshay, Louis-Marie Traonouez, Benoit Delahaye.

In 2012 we have successfully widened the applicability of interface and specification theories to systems with quantitative information such as energy usage, time constraints, or hybrid variables. Building on work done in 2011, we have introduced general quantitative specification theories. These provide a framework for reasoning about a wide range of different specification theories for different quantitative settings. We have provide one particularly important instantiation of the framework, which allows quantitative reasoning about real-time specifications.

Work on timed specifications theory has been continued in 2012 around the tool ECDAR. New case studies have been tested using the tool. These results, published in STTT, demonstrate the interest of the compositional approach for analyzing large systems. Besides the theory of robust specifications has been extended to allow a parametric estimation of the robustness. These results have been implemented in a new tool PyECDAR.

In 2012, we also successfully pursued our work on probabilistic specification theories by enhancing the framework of Abstract Probabilistic Automata, that we introduced in 2010, with several new operators. We first introduced a notion of satisfaction for stuttering implementations and showed how this new notion fits in the framework of APAs. Stuttering implementations are Probabilistic Automata that allow "silent" transitions by using local variables that are invisible to the specification. In this context, we also introduced a new logic, called ML-(A)PA that allows specifying properties of APA specifications and stuttering PA implementations. Our next contribution was to introduce a new difference operator. Given two specification APAs, their difference is a new APA that represents all implementations satisfying the one but not the other. This novel operator brings a new light to the well-known domain of counter-example generation.

Concerning Markov Chains, we have developed a new logic, LTL-I, which can only reason about fixed intervals instead of point values. We developed ϵ under and over approximation of formulas of this logics in [17], with associated algorithms. In all but few cases, we know that results of these algorithms are exact answers, while we didn't need to compute precisely and explicitly every probability involved. Another line of research is to consider very large Markov chain represented by Dynamic Bayesian Network. In [15], we compute only approximated results, as the size of the underlying Markov Chain is too big. However, evaluation of the algorithm shows small errors of our algorithm compared with the exact value.

6.7. Specific studies: Web services orchestrations

Participants: Ajay Kattepur, Albert Benveniste, Claude Jard.

Web services *orchestrations* and *choreographies* refer to the composition of several Web services to perform a co-ordinated, typically more complex task. We decided to base our study on a simple and clean formalism for WS orchestrations, namely the ORC formalism proposed by Jayadev Misra and William Cook [55].

Main challenges related to Web services QoS (Quality of Service) include: 1/ To model and quantify the QoS of a service. 2/ To establish a relation between the QoS of queried Web services and that of the orchestration (contract composition); 3/ To monitor and detect the breaching of a QoS contract, possibly leading to a reconfiguration of the orchestration. Typically, the QoS of a service is modeled by a *contract* (or Service Level Agreement, SLA) between the provider and the consumer of a given service. To account for variability and uncertainty in QoS, we proposed in previous work soft probabilistic contracts specified as probabilistic distributions involving the different QoS parameters; we studied *contract composition* for such contracts; we developed probabilistic QoS contract monitoring; and we studied the *monotonicity* of orchestrations; an orchestration is monotonic if, when a called service improves its performance, then so does the overall orchestration.

Last year, in the framework of the Associated Team FOSSA with the University of Texas at Austin (John Thywissen (PhD), Jayadev Misra and William Cook), we extended our approach to general QoS parameters, i.e., beyond response time. We now encompass composite parameters, which are thus only partially, not totally, ordered. We developed a general algebra to capture how QoS parameters are transformed while traversing the orchestration and we extended our study of monotonicity. Finally, we have developed corresponding contract composition procedures. This year, John Thywissen (from UT Austin) and Ajay Kattepur have prototyped a toolbox for Orc to support QoS-management. A journal paper is submitted.

A key task in extending Orc for QoS was to extend the Orc engine so that causalities between the different site calls are made explicit at run time while execution progresses. This benefits from our previous work on Orc semantics, but a new set of rules has been proposed to generate causalities in an efficient way, by covering new features of the language. This is joint work of Claude Jard, Ajay Kattepur and John Thywissen from Austin. An implementation on Orc is under development and a publication is in preparation.

Besides this main line of work, the additional topic of *Negotiation Strategies for Probabilistic Contracts in Web Services Orchestrations* has been addressed by Ajay Kattepur as part of his thesis, see [31]. Service Level Agreements (SLAs) have been proposed in the context of web services to maintain acceptable quality of service (QoS) performance. This is specially crucial for composite service orchestrations that can invoke many atomic services to render functionality. A consequence of SLA management entails efficient negotiation proto- cols among orchestrations and invoked services. In composite services where data and QoS (modeled in a probabilistic setting) interact, it is difficult to pick an individual atomic service to negotiate with. A superior improvement in one negotiated domain (eg. latency) might mean deterioration in another domain (eg. cost). In this work, we propose an integer programming formulation based on first order stochastic dom- inance as a strategy for re-negotiation over multiple services. A consequence of this is better end-to-end performance of the orchestration compared to random strategies for re-negotiation. We also demonstrate this optimal strategy can be applied to negotiation protocols specified in languages such as Orc. Such strategies are necessary for composite services where QoS contributions from individual atomic services vary significantly.

6.8. Specific studies: active documents and web services

Participants: Albert Benveniste, Loïc Hélouët, Sundararaman Akshay.

Active Documents have been introduced by the GEMO team at Inria Futurs, headed by Serge Abiteboul, mainly through the language *Active XML* (or *AXML* for short). AXML is an extension of XML which allows to enrich documents with *service calls* or sc's for short. These sc's point to web services that, when triggered, access other documents; this materialization of sc's produces in turn AXML code that is included in the calling document. One therefore speaks of dynamic or intentional documents. In the past years, we have collaborated with the GEMO team to study a distributed version of their language.

Last year, we have developed a distributed Active XML engine, which can be distributed over a network. We have built a lightweight experimentation platform, made of four Linux machines, that run DAXML services and communicate with one another. This year, we have started an experiment with a case study. We have proposed a distributed chess service palteform; the main idea is to use choreographies to provide solutions for chess problems, relying on an orchestration of specialized services for different phases of a game (opening, end of game, or collecting positions databases. We expect preliminary results in 2013.

Last year, we have proposed a new model, that combines arbitrary numbers of finite workflows, hence allowing for the definition of sessions. Sessions is a central paradigm in web-based systems. As messages exchange between two sites need not follow the same route over the net, a site can not rely on the identity of machines to uniquely define a transaction. This unique identification is essential: a commercial site, for instance, needs to manage several interactions at a given time. The current trend, as in BPEL, is to associate a unique identifier with each session. Modeling realistic sessions hence often forces to include session counters, and hence render most of properties undecidable. The session formalism studied in 2011 can be seen as a mix of BPEL and Orc elements, but was designed to keep several properties decidable (the formalism has the expressive power of reset Petri nets). The strength of this formalism is to allow designing systems that use sessions without the obligation to provide identifiers. Its drawback is that it only allows for the design of systems with a fixed number of agents. This year, we have continued extending last year's work with Ph. Darondeau from the S4 Team, and with M. Mukund from the Chennai Mathematical Institute to allow design of systems with sessions and allowing for an arbitrary number of agents.

6.9. Specific studies: network maintenance

Participants: Eric Fabre, Carole Hounkonnou.

This work represents part of our activities within the research group "High Manageability," supported by the common lab of Alcatel-Lucent Bell Labs (ALBLF) and Inria. It concerns a methodology for the graceful shut down and restart of routers in OSPF networks, one of the core protocols of IP networks. A methodology has been proposed to safely switch off the software layer of a router while still maintaining this router in the forwarding plane: the router still forwards packets, but is not able to adapt its routing table to changes in network conditions or topology. Nevertheless, it is possible to check whether this frozen router is harmless or can cause packet losses, through a centralized or distributed algorithm. And if ever it puts the network at risk, minimal patches can be set up temporarily until the router comes back to normal activity. This avoids running twice a global OSPF update at all nodes (one for shutdown of the equipment, one for restart). This work has been patented in June 2012 jointly with Alcatel-Lucent, and a publication on the topic was accepted at IM'2013.

6.10. Specific studies: network and service diagnosis

Participants: Eric Fabre, Carole Hounkonnou.

This work represents part of our activities within the research group "High Manageability," supported by the common lab of Alcatel-Lucent Bell Labs (ALBLF) and Inria. It is also supported by the UniverSelf EU integrated project, and conducted in cooperation with Orange Labs.

The objective is to develop a framework for the joint diagnosis of networks and of the supported services. We are aiming at a model-based approach, in order to tailor the methods to a given network instance and to follow its evolution. We also aim at active diagnosis methods, that collect and reason on alarms provided by the network, but that can also trigger tests or the collection of new observations in order to refine a current diagnosis.

Since 2011, an important effort was dedicated to a key and difficult part of this approach: the definition of a methodology for self-modeling. This consists in automatically building a model of the monitored system, by instantiating generic network elements. There are several difficulties to address:

- The model must capture several layers, from the physical architecture up to the service architecture and its protocols. As a case-study, we have chosen VoIP services on an IMS network, deployed over a wired IP network.

- The model should be hierarchical, to allow for multiscale reasoning, and to reflect the intrinsic hierarchical nature of the managed network.

- The model should be generic, i.e. obtained by assembling component instances coming from a reduced set of patterns, just like a text is obtained by assembling words.

- The model should be adaptive, to capture the evolving part of the network (e.g. introduction of new elements) but also its intrinsically dynamic nature (e.g. opened/closed connections).

- The model should display the hierarchical dependency of resources, specifically the fact that lower-level resources are assembled to provide a support to a higher level resource or functionality.

- The model should allow progressive discovery and refinement: for a matter of size, it is not possible to first build a model of the complete network and then monitor it; one must adopt an approach where the model is build on-line, and where the construction is guided by the progress of the diagnosis algorithms.

Elements of methodology achieving these goals were proposed in 2011, and further refined in 2012. Besides, we have also worked on the definition of generic Bayesian networks, that could translate into mathematical terms the dependency relations between network resources, in order to reason about them for failure diagnosis. A methodology was then designed to reason on such models. The idea is that one should first consider a subset of network resources (at a given granularity), in order to localize the origin of a given malfunction. The natural start point is the graph of all resources involved in the delivery of the malfunctioning service. As the fault localization is statistical, the model is then progressively expanded to capture more network elements and thus more observations, and thus refine the diagnosis. This model expansion is performed by introducing first

the most informative network elements, using information theory criteria. The result is a fault localization algorithm that explores only part of the network, and builds at runtime the necessary part of the model it should use to explain a malfunction [28]. The current efforts aim at extending these ideas to allow for the refinement of the model of some component (multiresolution reasoning).

7. Bilateral Contracts and Grants with Industry

7.1. Bilateral Contracts with Industry

7.1.1. HIMA

High Manageability (HiMa) is a research team hosted by the virtual joint research lab between Alcatel-Lucent Bell Labs France and Inria. This team is in its last year of existence, and most of its activity is now absorbed by the UniverSelf Eu IP (see below). DistribCom is involved in two topics: joint fault diagnosis in IMS networks and services (Carole Hounkonnou's thesis), and the early detection of anomalies in networks by analyzing the timed behavior of protocols (Aurore Junier's thesis). This work resulted in two publications at CNSM'12, and two joint patents on early fault detection and on the graceful maintenance of OSPF networks.

8. Partnerships and Cooperations

8.1. Regional Initiatives

8.1.1. ESTASE

Participant: Axel Legay.

- Title: Estase
- Type: Regional project
- Defi: New techniques for statistical model checking
- Instrument: Regional project
- Duration: March 2011 - February 2014
- Coordinator: Inria Rennes

8.2. National Initiatives

8.2.1. ANR

8.2.1.1. IMPRO
Participant: Loïc Hélouët.

Title: IMPRO

Type: ANR

Defi: Implementability and Robustness of Timed Systems

Duration: march 2011 - march 2014

Coordinator: IRCCYN Nantes

Others partners: IRCCyN (Nantes), IRISA (Rennes), LIP6 (Paris), LSV (Cachan), LIAFA (Paris), LIF (Marseilles)

See also: http://anr-impro.irccyn.ec-nantes.fr/

Abstract: This project addresses the issues related to the practical implementation of formal models for the design of communicating embedded systems: such models abstract many complex features or limitations of the execution environment. The modeling of time, in particular, is usually ideal, with infinitely precise clocks, instantaneous tests or mode commutations, etc. Our objective is thus to study to what extent the practical implementation of these models preserves their good properties. We will first define a generic mathematical framework to reason about and measure implementability, and then study the possibility to integrate implementability constraints in the models. We will particularly focus on the combination of several sources of perturbation such as resource allocation, the distributed architecture of applications, etc. We will also study implementability through control and diagnostic techniques. We will finally apply the developed methods to a case study based on the AUTOSAR architecture, a standard of the automotive industry.

8.3. European Initiatives

8.3.1. DISC

Participant: Eric Fabre.

The DISC Eu project (STREP) officially ended in Dec. 2011, and the final review took place in Feb. 2012. This project was oriented toward the development of supervision and control methods for large systems. Inria was involved in particular for the diagnosis of stochastic systems, and for distributed planning methods. These activities are still going on, with several publications in 2012 and others in preparation. Among the salient facts related to DISC in 2012 were Loig Jezequel's PhD defense (Dec. 2012), and the contribution to 2 chapters of the book "Control of discrete-event systems" seatzu:silva:vanschuppen:2013, to appear in 2013.

8.3.2. Sys2SOFT

Participant: Axel Legay.

Title: SyS2SOFT

Type: Grand emprunt

Defi: Designing for adaptability and evolution in systems of sytems engineering

Instrument: Grand emprunt

Duration: Juin 2012 - Mai 2015

Coordinator: DASSAULT

8.3.3. FP7 Projects

8.3.3.1. Dali

Participant: Axel Legay.

- Title: Dali
- Type: COOPERATION (ICT)
- Defi: design of a device for assisted living.
- Instrument: Strep.
- Duration: November 2011 - October 2014
- Coordinator: Trento (Italy)

8.3.3.2. DANSE
    Participant: Axel Legay.

        Title: DANSE

        Type: COOPERATION (ICT)

        Defi: Designing for adaptability and evolution in systems of sytems engineering

        Instrument: Integrated Project (IP)

        Duration: November 2011 - October 2014

        Coordinator: OFFIS (Germany)

        Abstract: DANSE represents the next step in research about component based design and it is thus central in our research activities. The purpose of this project is the development of a new methodology for the design of Systems of Systems (SoS). SoS are modeled using the UPDM Language. In these settings, Statistical Model Checking is the solution to evaluate the SoS capabilities to ensure some properties. During the first period (Nov. 2011 - Nov. 2012), we and ALES company both worked to interface PLASMA and DESYRE to provide the first statistical model-checker tool for the UPDM modeling framework. PLASMA-DESYRE is available and run under the Eclipse environment. To obtain the first prototype of PLASMA-DESYRE we provide a new release of Plasma. It is specially designed to perform SMC using different simulation engines, by reducing the adaptation effort: it can be connected to DESYRE, SciLab, MatLab, and some simulators dedicated to Bio or Prism languages. We also extended UPDML specification with a new contract language designed to specify some requirements. These requirements are viewed as behavioral objectives that lead the system architect for designing some good strategies of the SoS. These requirements (called contracts) are written in English using some patterns that are simple to handle and have a strong semantics expressed with the Bounded Linear-Temporal-Logic (B-LTL), the property language of PLASMA. This new language is defined using the standard OCL language to define state constraints of the SoS, English temporal patterns that overlay the state constraints to specify some contracts about the behavior of the SoS. It adds the time support that is not initially provided by OCL. These contracts are then compiled into B-LTL formulas and checked by PLASMA-DESYRE, the SoS Statistical Model Checker, against a compiled implementation of the UPDM model. The result estimates the satisfiability of the contract, e.g. the probability that the model satisfies the contract.

8.3.3.3. Univerself
    Participant: Eric Fabre.

        Title: Univerself

        Type: COOPERATION (ICT)

        Defi: The Network of the Future

        Instrument: Integrated Project (IP)

        Duration: September 2010 - August 2013

        Coordinator: Alcatel Lucent (France)

        Others partners:

        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 F$\sqrt{\partial}$rderung 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).

See also: http://www.univerself-project.eu/

Abstract: UniverSelf unites 17 partners with the aim of overcoming the growing management complexity of future networking systems, and to reduce the barriers that complexity and ossification pose to further growth. Univerself has been launched in October 2010 and is scheduled for four years.

8.3.3.4. SENSATION

Participant: Axel Legay.

- Title: Sensation
- Type: COOPERATION (ICT)
- Defi: Study of new techniques for energy saving
- Instrument: Strep.
- Duration: October 2012 - September 2015
- Coordinator: Aalborg (Denmark)

8.4. International Initiatives

8.4.1. Inria Associate Teams

8.4.1.1. FOSSA

Participants: Claude Jard, Albert Benveniste.

The associated team FOSSA studies the **formalization of service orchestrations in the open world** of the Internet. The original FOSSA consortium involved two teams on the Inria side, namely Distribcom (Albert Benveniste and Claude Jard, Rennes, leader of FOSSA) and Mexico (Stefan Haar, Saclay). In early 2011, both teams agreed that Mexico did not have the resources to participate in FOSSA at an appropriate level. So they agreed that Mexico would no longer participate in FOSSA. The team of Cook and Misra at the Computer Science Department, University of Texas at Austin, is among the leading teams on wide area distributed systems and programming. Jayadev Misra [1] has a long record of results tracing back to the 1980's with his work on the Unity language. Since 2000, he and William Cook are committed to the design and development of the ORC[2] script language for composite services over the Web. This team is therefore the premier player in this area, combining both a strong theoretical research and a professional tool development. Since his launching in 2004, the DistribCom [3] Inria team, with Albert Benveniste, Claude Jard, and Loïc Hélouët, is involved in the study of Quality of Service (QoS) issues in service orchestrations as well as document based workflows. FOSSA was started with the overall objective of enhancing ORC with the advances performed by DistribCom on the above two subjects.

---

[1] http://www.cs.utexas.edu/~misra/
[2] http://orc.csres.utexas.edu
[3] http://www.irisa.fr/distribcom/

FOSSA has lived from 2010 to 2012. QoS weaving was the main topic developed in 2012. John Thywissen (Austin side), Ajay Kattepur and Claude Jard (Inria side) were the principal contributors. The strategy was to first focus on causality tracking. This has been implemented in ORC using transformations in the OIL intermediate form. Causality has then been extended with QoS and implemented. A joint paper is under finalization. This year, we have also worked on a joint general paper on the overall approach. On the topic of Active XML and ORC integration, the team has decided to put energy on the development of the AXML REST platform developed by Loïc Hélouët and Benoît Masson (post-doctorate). This platform is a natural candidate for integrating AXML+ORC, as we think. But the cooperative work has not really started, due to overload of the corresponding teams.

8.4.2. Inria International Partners

Distribcom has lively collaboration with the National University of Singapore, where Blaise Genest spent the last 3 years. We also have long lasting collaboration with the Chennai Mathematical Institute.

8.4.3. Participation In International Programs

8.4.3.1. Danish-French collaboration

> Program: Action des ambassades de France
>
> Title: Modular design and verification of stochastic systems
>
> Inria principal investigator: Axel LEGAY
>
> International Partner (Institution - Laboratory - Researcher):
>
> > University of Aalborg (Denmark)
>
> Duration: Jan 2010 - Dec 2012

8.4.3.2. Tournesol (Belgium)

> Program: PHC
>
> Title: Vérification de lignes de produits logiciels
>
> Inria principal investigator: Axel LEGAY
>
> International Partner (Institution - Laboratory - Researcher):
>
> > University of Namur (Belgium)
>
> Duration: Jan 2011 - Dec 2012

8.5. International Research Visitors

8.5.1. Visits of International Scientists

Narayan K. Kumar and Madhavan Mukund from the Chennai mathematical institute visited Dostribcom in april (1 week each) to continue working on session models in web services, and to launch new research on robustness in distributed systems.

Danilo Ardagna visited Distribcom in October 2012

Prof. Michele Pinna (niv. Cagliari) visited DistribCom from Sept. 1 to Sept. 30.

Andrzej Wasowski visited Distribcom in February 2012 Jan Kretiensky visited Distribcom in September 2012

8.5.1.1. Internships

Guillaume Aucher supervised the internship of Himani Rajora (IIT, Delhi, India) entitled "Distances between Kripke models".

Axel Legay supervised the internship of Alessio Colombu (Trento), Hoa Lee (Trento), and Fabrizio Biondi (ITU Copenhagen).

8.5.2. *Visits to International Teams*

Guillaume Aucher has visited Thomas Bolander (DTU, Copenhagen) the last week of August 2012. The collaboration was very fruitful and has resulted in significant results related to epistemic planning (DEL) (to be submitted).

Guillaume Aucher visited Leon van der Torre at the university of Luxembourg in November 2012. This visit was scheduled at the same time Samir Chopra and Guido Boella were in Luxembourg. Guido Boella is specialist of law and computer science and Samir Chopra is a logician who recently published a book on law and autonomous agents together with the jurist Laurence White. The visit was very instructive and profitable.

Guillaume Aucher was invited (his travel and accommodation expenses have been reimbursed) by Sonja Smets and Alexandru Baltag at the University of Amsterdam the last week of September 2012 to give two seminars at the ILLC and to work in collaboration with them.

Guillaume Aucher was an invited speaker of the workshop "dynamics in logic II" (Lille, March 2012).

Loïc Hélouët spent 10 days in march 2012 at the Chennai Mathematical Institute to pursue collaboration on verification of session models.

Axel Legay was invited researcher at Namur University multiple times. He was also an invited researcher at ITU Copenhagen.

Eric Fabre visited MIT (LIDS) from June 16 to June 20.

9. Dissemination

9.1. Scientific Animation

A. Benveniste is the Scientific Director of the CominLabs Excellence Center (Laboratoire d'Excellence, part of the program *Investissements d'Avenir* of the french government). He is member of the Strategic Advisory Council of the Institute for Systems Research, Univ. of Maryland, College Park, USA. He is president of the Scientific Committee of the *Common Bell Labs Inria Laboratory*. He is member of the Scientific Council of France Telecom. [4] Albert Benveniste is a member of the French Academy of Technologies.

Loïc Hélouët co-organizes the 68NQRT weekly seminar on formal methods. This seminar proposes around 40 talks each year. For more details, see http://68nqrt.inria.fr.

Claude Jard was the scientific director of the research of the Brittany extension of the ENS Cachan. He recently moved to Nantes, where he is the director of the CNRS cluster Atlantic, gathering the public laboratories in ITCS in Pays de la Loire.

Eric Fabre leads the High Manageability joint team of Inria and Alcatel Lucent.

Axel Legay Co-organized the quantitative methods workshop at formal methods days. He also co-organized the winter school on quantitative methods at IT Copenhagen.

Guillaume Aucher was a PC member of AAMAS 2013, an auxiliary reviewer of TARK 2013, and served as a reviewer for the Logic Journal of the IGPL. He has been elected a member of the scientific council of the University of Rennes 1.

François Schwarzentruber was an auxiliary reviewer of TARK 2013 and AAMAS 2013.

---

[4]Only facts related to the activities of DistribCom team are mentioned. Other roles or duties concern the S4 team, to which A. Benveniste also belongs.

9.2. Teaching - Supervision - Juries

9.2.1. Teaching

Licence : Guillaume Aucher, Programmation Impérative 1, 40h eq. TD, L1, University of Rennes 1, France

Licence : Guillaume Aucher, Algorithms for graphs, 20h eq. TD, L3, University of Rennes 1, France

Doctorat : Enseignant, titre du cours, nombre d'heures en $\sqrt{}$©quivalent TD, universit$\sqrt{}$©, pays

Licence : Loïc Hélouët, JAVA courses, 32 h eq TD, L1, INSA, France

Master : Loïc Hélouët, Algorithms courses to students at the aggregation level, 16 h eq TD, aggregation, ENS Cachan-Antenne de Bretagne, France

Master : Eric Fabre, Distributed Algorithms and Distributed Systems, 12h eq. TD, M2 Rech. Comp. Sc., Univ. Rennes 1, France

Master : Eric Fabre, Information Theory, 15h eq. TD, M1, ENS Cachan (Rennes), France

Licence : Claude Jard, Formal languages, Distributed Computing, 44 h eq TD, L1, ENS Cachan-antenne de Bretagne, France

Master : Claude Jard, Algorithms courses to students at the aggregation level, 16 eq TD, aggregation, ENS Cachan-antenne de Bretagne, France

Licence: François Schwarzentruber, Introduction to algorithms (ALGO1), 32h eq TD, L3, ENS Cachan-antenne de Bretagne, France

Master: François Schwarzentruber, Software design and verification (CVFP), 32h eq TD, M1, ENS Cachan-antenne de Bretagne, France

Licence and master: François Schwarzentruber, Seminars for students (SEMIN1, SEMIN2), 36h eq TD, L3, M1, ENS Cachan-antenne de Bretagne, France

Master: François Schwarzentruber, Complexity theory, 6h eq TD, agrégation, ENS Cachan-antenne de Bretagne, France

Master: François Schwarzentruber, Introduction to discrete mathematics. 9 eq TD, University of Rennes 1, France

Master: François Schwarzentruber, Practical sessions in algorithmics. 6 eq TD, agrégation, ENS Cachan-antenne de Bretagne, France

9.2.2. Supervision

PhD : Ajay Kattepur, Flexible Quality of Service Management of Web Services Orchestrations, Université de Rennes 1, Nov. 8, 2012, Albert Benveniste, Claude Jard

PhD in progress : Rouwaida Abdallah, Synthèse à partir de scénarios, February 2010, Loïc Hélouët, Claude Jard

PhD in progress : Aurore Junier, Network calculus applied to network stability analysis, sept. 2010, C. Jard, A. Bouillard.

PhD in progress : Carole Hounkonou, A methodology for joint network and service self-diagnosis, Oct. 2009, Eric Fabre.

PhD in progress: Cyril Jegourel, statistical model checking for rare-event systems, Axel Legay.

PhD in progress : Bastien Maubert, Logical Foundations of Games with Imperfect Information, University of Rennes 1, 09/2010, Guillaume Aucher and Sophie Pinchinat (S4, Irisa)

PhD : Loig Jezequel, Distributed Optimal Planning in Large Distributed systems, ENS Cachan, Dec. 2012, Eric Fabre

10. Bibliography

Major publications by the team in recent years

[1] S. ABBES, A. BENVENISTE. *True-concurrency probabilistic models: Markov nets and a law of large numbers*, in "Theoretical Computer Science", 2008, vol. 390, n° 2-3, p. 129-170.

[2] A. BENVENISTE, E. FABRE, C. JARD, S. HAAR. *Diagnosis of asynchronous discrete event systems, a net unfolding approach*, in "IEEE Transactions on Automatic Control", May 2003, vol. 48, n° 5, p. 714–727.

[3] T. CHATAIN, C. JARD. *Complete finite prefixes of symbolic unfoldings of safe time Petri nets*, in "Proc. of ICATPN", LNCS, June 2006, n° 4024, p. 125–145.

[4] E. FABRE, A. BENVENISTE. *Partial Order Techniques for Distributed Discrete Event Systems: why you can't avoid using them*, in "Journal of Discrete Event Dynamic Systems (JDEDS)", 2007, vol. 17, p. 355-403.

[5] E. FABRE. *Trellis Processes: a Compact Representation for Runs of Concurrent Systems*, in "Journal of Discrete Event Dynamic Systems (JDEDS)", 2007, vol. 17, n° 3, p. 267–306.

[6] E. FABRE, L. JÉZÉQUEL. *Distributed optimal planning: an approach by weighted automata calculus*, in "CDC", 2009, p. 211-216.

[7] T. GAZAGNAIRE, B. GENEST, L. HÉLOUËT, P. S. THIAGARAJAN, S. YANG. *Causal Message Sequence Charts*, in "Theoretical Computer Science (TCS)", 2009, vol. 410, n° 41, p. 4094-4110.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[8] L. JEZEQUEL. *Distributed Optimal Planning in Large Distributed Systems*, ENS Cachan, 2012.

[9] A. KATTEPUR. *Flexible Quality of Service (QoS) Management of Web Services Orchestrations*, Univ. Rennes 1, 2012.

Articles in International Peer-Reviewed Journals

[10] S. AKSHAY, B. GENEST, L. HÉLOUËT, S. YANG. *Regular set of representatives for time-constrained MSC graphs*, in "Inf. Process. Lett.", 2012, vol. 112, n° 14-15, p. 592-598.

[11] G. AUCHER. *DEL-Sequents for regression and epistemic planning*, in "Journal of Applied Non-Classical Logics", 2012, vol. 22, n° 4, p. 337-367, http://dx.doi.org/10.1080/11663081.2012.736703.

[12] A. BASU, S. BENSALEM, M. BOZGA, B. DELAHAYE, A. LEGAY. *Statistical abstraction and model-checking of Large Heterogeneous Systems*, in "International Journal on Software Tools for Technology Transfer", 2012, vol. 14, n° 1, p. 53-72 [*DOI :* 10.1007/s10009-011-0201-2], http://hal.inria.fr/hal-00722489.

[13] A. CLASSEN, M. CORDY, P. HEYMANS, P.-Y. SCHOBBENS, A. LEGAY. *Model Checking for Software Product Lines with SNIP*, in "International Journal on Software Tools for Technology Transfer (STTT)", 2012 [*DOI :* 10.1007/s10009-012-0234-1], http://hal.inria.fr/hal-00717956.

[14] L. HÉLOUËT, C. JARD, R. ABDALLAH. *Distributed Implementation of Message Sequence Charts*, in "SOSYM", 2013, to appear.

[15] S. K. PALANIAPPAN, S. AKSHAY, B. LIU, B. GENEST, P. S. THIAGARAJAN. *A Hybrid Factored Frontier Algorithm for Dynamic Bayesian Networks with a Biopathways Application*, in "IEEE/ACM Trans. Comput. Biology Bioinform.", 2012, vol. 9, n° 5, p. 1352-1365.

[16] F. SCHWARZENTRUBER. *Complexity Results of STIT Fragments*, in "Studia Logica", 2012, vol. 100, n⁰ 5, p. 1001-1045.

International Conferences with Proceedings

[17] M. AGRAWAL, S. AKSHAY, B. GENEST, P. S. THIAGARAJAN. *Approximate Verification of the Symbolic Dynamics of Markov Chains*, in "Proceedings of the 27th Annual IEEE Symposium on Logic in Computer Science, LICS 2012", IEEE, 2012, p. 55-64.

[18] S. AKSHAY, B. GENEST, L. HÉLOUËT, S. YANG. *Symbolically Bounding the Drift in Time-Constrained MSC Graphs*, in "Theoretical Aspects of Computing - ICTAC 2012 - 9th International Colloquium", A. ROYCHOUDHURY, M. D'SOUZA (editors), Lecture Notes in Computer Science, Springer, 2012, vol. 7521, p. 1-15.

[19] S. AKSHAY, L. HÉLOUËT, C. JARD, D. LIME, O. H. ROUX. *Robustness of Time Petri Nets under Architectural Constraints*, in "FORMATS", Lecture Notes in Computer Science, 2012, vol. 7595, p. 11-26.

[20] S. AKSHAY, L. HÉLOUËT, C. JARD, P.-A. REYNIER. *Robustness of Time Petri Nets under Guard Enlargement*, in "RP", Lecture Notes in Computer Science, 2012, vol. 7550, p. 92-106.

[21] G. AUCHER, B. MAUBERT, F. SCHWARZENTRUBER. *Generalized DEL-Sequents*, in "JELIA", L. F. DEL CERRO, A. HERZIG, J. MENGIN (editors), Lecture Notes in Computer Science, Springer, 2012, vol. 7519, p. 54-66.

[22] G. AUCHER, F. SCHWARZENTRUBER. *On the complexity of Dynamic Epistemic Logic*, in "TARK2013", 2013, to appear.

[23] M. CORDY, A. CLASSEN, P. HEYMANS, P.-Y. SCHOBBENS, A. LEGAY. *Managing Evolution in Software Product Lines : A Model-Checking Perspective*, in "6th International Workshop on Variability Modelling of Software Intensive Systems", Leipzig, Saxe, Germany, ACM, January 2012, p. 183-191, http://hal.inria.fr/hal-00718216.

[24] M. CORDY, A. CLASSEN, G. PERROUIN, P.-Y. SCHOBBENS, P. HEYMANS, A. LEGAY. *Simulation-Based Abstractions for Software Product-Line Model Checking*, in "34th International Conference on Software Engineering, ICSE 2012", Zurich, Switzerland, M. GLINZ, G. MURPHY, M. PEZZE (editors), June 2012, p. 672-682, http://hal.inria.fr/hal-00718047.

[25] M. CORDY, P. HEYMANS, P.-Y. SCHOBBENS, A. LEGAY. *Behavioural Modelling and Verification of Real-time Software Product Lines*, in "16th Software Product Line Conference", Salvador de Bahia, Brazil, ACM, September 2012, http://hal.inria.fr/hal-00718157.

[26] X. DEVROEY, M. CORDY, G. PERROUIN, E.-Y. KANG, P.-Y. SCHOBBENS, P. HEYMANS, A. LEGAY, B. BAUDRY. *A Vision for Behavioural Model-Driven Validation of Software Product Lines*, in "International Symposium On Leveraging Applications of Formal Methods, Verification and Validation", Amirandes, Heraclion, Greece, October 2012, http://hal.inria.fr/hal-00726121.

[27] X. DEVROEY, M. CORDY, G. PERROUIN, E.-Y. KANG, P.-Y. SCHOBBENS, P. HEYMANS, A. LEGAY, B. BAUDRY. *Towards Behavioural Model-Driven Validation of Software Product Lines*, in "5th International

Sympoisum on Leveraging Applications of Formal Methods", Héraklion, Crète, Greece, October 2012, http://hal.inria.fr/hal-00718180.

[28] C. HOUNKONNOU, E. FABRE. *Empowering self-diagnosis with self-modeling*, in "Conf. on Network and System Management, CNSM'12", 2012, p. 364-370.

[29] L. JEZEQUEL, E. FABRE. *A# : a distributed version of A* for factored planning*, in "IEEE Conf. on Decision and Control, CDC'12", 2012, p. 7377-7382.

[30] L. JEZEQUEL, E. FABRE. *Turbo Planning*, in "Workshop on Discrete Event Systems, WODES'12", 2012.

[31] A. KATTEPUR, A. BENVENISTE, C. JARD. *Negotiation Strategies for Probabilistic Contracts in Web Services Orchestrations*, in "ICWS", IEEE, 2012, p. 106-113.

[32] D. LIME, O. H. ROUX, C. JARD. *Clock Transition Systems*, in "CS&P", 2012, p. 227-238.

[33] F. SCHWARZENTRUBER, S. VESIC, T. RIENSTRA. *Building an Epistemic Logic for Argumentation*, in "JELIA", L. F. DEL CERRO, A. HERZIG, J. MENGIN (editors), Lecture Notes in Computer Science, Springer, 2012, vol. 7519, p. 359-371.

Scientific Books (or Scientific Book chapters)

[34] L. F. DEL CERRO, A. HERZIG, J. MENGIN (editors). *Logics in Artificial Intelligence - 13th European Conference, JELIA 2012, Toulouse, France, September 26-28, 2012. Proceedings*, Lecture Notes in Computer Science, Springer, 2012, vol. 7519.

[35] C. SEATZU, M. SILVA, J. H. VAN SCHUPPEN (editors). *Control of discrete-event systems – Automata and Petri net perspectives*, Lecture Notes in Control and Information Sciences, Springer Verlag London Ltd., London, 2013, n° 433, 479, DISC Project, DISC Ph.D. School 2011 [*DOI :* DOI 10.1007/978-1-4471-4276-8], http://link.springer.com/book/10.1007/978-1-4471-4276-8.

[36] G. AUCHER. *Principles of Knowledge, Belief and Conditional belief*, in "Dialogue, Rationality, Formalism.", G. HEINZMANN, M. MUSIOL, M. REBUSCHI, A. TROGNON (editors), Logic, Epistemology and the Unity of Science, Springer, 2012, to appear.

Research Reports

[37] G. AUCHER, B. MAUBERT, F. SCHWARZENTRUBER. *Generalized DEL-sequents*, Inria, July 2012, n^o RR-8012, 23, http://hal.inria.fr/hal-00716074.

[38] G. AUCHER, F. SCHWARZENTRUBER. *On the Complexity of Dynamic Epistemic Logic (Extended Version)*, Inria, November 2012, n^o RR-8164, 27, http://hal.inria.fr/hal-00759544.

[39] L. JEZEQUEL, E. FABRE. *A-sharp: a Distributed A-star for Factored Planning*, Inria, March 2012, n^o RR-7927, http://hal.inria.fr/hal-00687434.

[40] A. JUNIER, A. BOUILLARD, R. BENOIT. *Hidden Anomaly Detection in Telecommunication Networks*, Inria, October 2012, n^o RR-7979, http://hal.inria.fr/hal-00702587.

[41] D. LIME, O. H. ROUX, C. JARD. *Clock Transition Systems*, August 2012, http://hal.inria.fr/hal-00725792.

References in notes

[42] S. ABITEBOUL, L. SEGOUFIN, V. VIANU. *Static analysis of active XML systems*, in "ACM Trans. Database Syst.", 2009, vol. 34, n° 4.

[43] T. ANDREW, F. CURBERA, H. Y. GOLAND, J. KLEIN, F. LEYMANN, K. LIU, D. ROLLER, D. SMITH, S. THATTE, I. TRICKOVIC, S. WEERAWARANA. *Business Process Execution Language for Web Services. [BPEL4WS.], version 1.1.*, May 2003, http://xml.coverpages.org/BPELv11-May052003Final.pdf.

[44] M. G. BUSCEMI, V. SASSONE. *High-Level Petri Nets as Type Theories in the Join Calculus*, in "FoSSaCS", Lecture Notes in Computer Science, Springer, 2001, vol. 2030, p. 104-120.

[45] C. CASSANDRAS, S. LAFORTUNE. *Introduction to discrete event systems*, Kluwer Academic Publishers, 1999.

[46] R. DEVILLERS, H. KLAUDEL. *Solving Petri net recursions through finite representation*, in "IASTED International Conference on Advances in Computer Science and Technology, ACST'2004", ACTA Press, 2004, p. 145-150, ISBN 0-88986-497-3.

[47] E. FABRE, L. JÉZÉQUEL. *Distributed Optimal Planning: an Approach by Weighted Automata Calculus*, in "Conference on Detection and Control (CDC)", 2009.

[48] R. FAGIN, J. HALPERN, Y. MOSES, M. VARDI. *Reasoning about knowledge*, MIT Press, 1995.

[49] P. GASTIN, K. NARAYAN. KUMAR, M. MUKUND. *Reachability and boundedness in time-constrained MSC graphs*, in "Perspectives in Concurrency Theory, festchrift for P.S. Thiagarajan", 2008.

[50] J. HALPERN, Y. MOSES. *Knowledge and common knowledge in a distributed environment*, in "Journal of the ACM", 1990, vol. 37, n° 3, p. 549–587.

[51] L. HÉLOUËT, C. JARD. *Conditions for synthesis of communicating automata from HMSCs*, in "5th International Workshop on Formal Methods for Industrial Critical Systems (FMICS)", 2000.

[52] ITU-TS. *ITU-TS Recommendation Z.120: Message Sequence Chart (MSC)*, ITU-TS, Geneva, September 1999.

[53] L. JEZEQUEL, E. FABRE, P. HASLUM, S. THIEBAUX. *Cost-Optimal Factored Planning: Promises and Pitfalls*, in "ICAPS, Int. Conf. on Applications of Planning and Scheduling", May 2010.

[54] J.-J. C. MEYER, W. VAN DER HOEK. *Epistemic Logic for AI and Computer Science*, Cambridge University Press, Cambridge, 1995.

[55] J. MISRA. *A Programming Model for the Orchestration of Web Services*, in "SEFM", IEEE Computer Society, 2004, p. 2-11.

[56] M. NIELSEN, G. PLOTKIN, G. WINSKEL. *Petri nets, event structures and domains, part 1*, in "Theoretical Computer Science", 1981.

[57] OBJECT MANAGEMENT GROUP. *Unified Modeling Language Specification version 2.0: Superstructure*, OMG, 2003, n^o pct/03-08-02.

[58] W. REISIG. *Petri nets*, Springer Verlag, 1985.

[59] M. RENIERS, S. MAUW. *High-level Message Sequence Charts*, in "SDL97: Time for Testing - SDL, MSC and Trends", Evry, France, A. CAVALLI, A. SARMA (editors), Proceedings of the Eighth SDL Forum, September 1997, p. 291-306.

[60] M. RENIERS. *Message Sequence Charts: Syntax and Semantics*, Eindhoven University of Technology, 1998.

[61] E. RUDOLPH, P. GRAUBMAN, J. GRABOWSKI. *Tutorial On Message Sequence Charts*, in "Computer Networks and ISDN Systems", 1996, vol. 28, p. 1629-1641.

[62] G. WINSKEL. *Event structures semantics in CCS and related languages*, in "Lecture Notes in Computer Science", Springer Verlag, 1982, vol. 140.

[63] G. WINSKEL. *Categories of Models for Concurrency*, in "Seminar on Concurrency", Lecture Notes in Computer Science, Springer, 1985, vol. 197, p. 246-267.

[64] J. VAN BENTHEM, J. GERBRANDY, E. PACUIT. *Merging Frameworks for Interaction: DEL and ETL*, in "Theoretical Aspect of Rationality and Knowledge (TARK XI)", Brussels, D. SAMET (editor), June 2007, p. 72–82.