INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team Cassis

Combining approaches for the security of infinite state systems

Nancy - Grand Est

THEME SYM

Table of contents

1. Team

Research Scientist

Serge Burckel [MC, U. de la Réunion, seconded to INRIA, HdR]
Yannick Chevalier [MC, U. Paul Sabatier, Toulouse, seconded to INRIA]
Véronique Cortier [CR, CNRS-LORIA]
Christophe Ringeissen [CR, INRIA-LORIA]
Michaël Rusinowitch [Team Leader, Research Director (DR), INRIA-LORIA, HdR]
Mathieu Turuani [CR, INRIA-LORIA]

Faculty Member

Fabrice Bouquet [PR, Université Franche-Comté, HdR]
Frédéric Dadeau [MC, Université Franche-Comté]
Alain Giorgetti [MC, Université Franche-Comté]
Pierre-Cyrille Héam [MC, Université Franche-Comté, seconded to CNRS, Cachan since September 1]
Olga Kouchnarenko [Vice-head of project team, PR, Université Franche-Comté, LIFC, HdR]
Abdessamad Imine [MC, Université Nancy 2]
Laurent Vigneron [MC, Université Nancy 2]

External Collaborator

Silvio Ranise [CR, INRIA-LORIA, in sabbatical stay, Univ. of Milan]

Technical Staff

Aloïs Dreyfus [Engineer ANR RAVAJ, LIFC, from October 1]
Kalou Cabrera [Engineer ODL, LIFC, from December 1]

PhD Student

Mumtaz Ahmad [SFERE (Pakistan), LORIA]
Tigran Avanesov [INRIA, LORIA]
Asma Berregba [MENRT, LORIA, from October 1]
Thibaut Brocard [BDI-CNRS, LIFC]
Pierre-Christophe Bué [MENRT, LIFC, from October 1]
Najah Chridi [MENRT, LORIA until August 31, and ATER, UHP since September 1]
Roméo Courbis [LIFC, until November 1, 2010]
Stéphane Debricon [INTERREG, LIFC]
Adrien de Kermadec [VALMI, Co-tutelle LIFC and New-Zeland]
Jonathan Lasalle [project VETESS, LIFC, from October 2008]
Mohamed Anis Mekki [INRIA, LORIA]
Vincent Pretre [INTERREG, LIFC]
Daniele Zucchelli [“Cotutelle” between U. of Milan and U. Henri Poincaré, Nancy]

Post-Doctoral Fellow

Enrica Nicolini [Post-doctoral INRIA]

Administrative Assistant

Emmanuelle Deschamps

2. Overall Objectives

2.1. Background

Cassis is a joint project between the *Laboratoire Lorrain de Recherche en Informatique et ses Applications (LORIA - UMR 7503)* and *Laboratoire d’Informatique de l’Université de Franche-Comté (LIFC - FRE 2661)*.

The objective of the project is to design and develop tools to verify the safety of systems with an infinite number of states. The analysis of such systems is based on a symbolic representation of sets of states in terms of formal languages or logical formulas. Safety is obtained via automatic proof, symbolic exploration of models or test generation. These validation methods are complementary. They rely on the study of accessibility problems and their reduction to constraint solving.

An originality of the project is its focus on infinite systems, parameterized or large scale, for which each technique taken separately shows its limits. This is the case for example with protocols operating on topologies of arbitrary size (ring networks), systems handling data structures of any size (sets), or whose control is infinite (automata communicating through an unbounded buffer). Ongoing or envisioned applications concern embedded software (e.g., smart cards, automotive controllers), cryptographic protocols (IKE, SET, TLS, Kerberos) designed to ensure trust in electronic transactions, and distributed systems.

The problem of validating or verifying reactive systems is crucial because of the increasing number of security-sensitive systems. The failure of these critical systems can have dramatic consequences since they may be embedded in vehicles components, or they control power stations or telecommunication networks. Beside obvious security issues, the reliability of products whose destination is millions of end-users has a tremendous economical impact.

There are several approaches to system verification: automated deduction, reachability analysis or model-checking, and testing. These approaches have different advantages and drawbacks. Automated deduction can address practical verification, however it remains complex to handle and requires a lot of expertise and guidance from the user. Model-checking is exhaustive but must face combinatorial explosion and becomes problematic with large-size or infinite systems. Testing is fundamental for validating requirements since it allows the discovery of many errors. However, it is almost never exhaustive and therefore only leads to partial solutions. Hence we believe that these approaches should not be considered as competing but as complementary.

The goal of our project is to contribute to new combinations of these three verification techniques in a framework that would apply them in an industrial context. In particular we expect some breakthrough in the infinite-state verification domain by joint applications of deductive, model-checking and testing techniques.

2.2. Context

For verifying the security of infinite state systems we rely on

- Different ways to express the safety, reachability or liveness properties of systems, linear-time or branching-time logics, and the application of abstraction or abstract interpretation.
- Test generation techniques.
- The modeling of systems by encoding states as words, terms or trees and by representing infinite sets of states by languages. To each of these structures corresponds appropriate action families, such as transductions or rewritings.

Our goal is to apply these different approaches for ensuring the security of industrial systems by providing adequate methods and tools. In more details we aim at the following contributions (see the continuous lines in Figure 1):

1. verification of abstract models derived from existing systems;
2. tests generation from the abstract model for validating the existing model;
3. cross-fertilization of the different validation techniques (deduction, model-checking, testing) by taking advantage of the complementary scopes and of their respective algorithmic contributions.

Let us mention that all these techniques comply with various development methodologies.

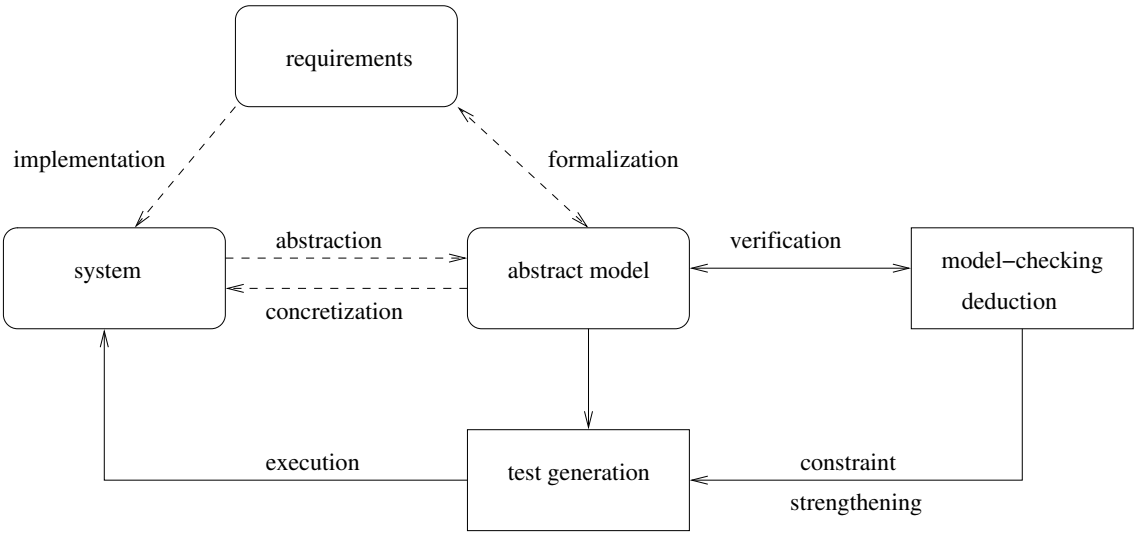

Figure 1. Software validation in Cassis

2.3. Challenge

Verifying the safety of infinite state systems is a challenge: nowadays algorithmic techniques only apply to very specific infinite state systems. On the other hand the deductive approaches are good candidates to capture infinite system safety verification but are difficult to bring into operation and require a deep expertise. A solution consists of integrating several verification methods by combining, for example, theorem-proving and model-checking.

The behavior of infinite states systems is expressed in the various models by composing or iterating actions. One of the main problems with algorithmic techniques is to compute the effect of these actions on the initial state. This computation is called *reachability analysis*. The verification of safety properties as well as the automatic generation of test cases relies heavily on the accuracy of reachability analysis.

The transverse goal is to push away the limitations on the use of formal verification techniques, to ease their applications, and to let them scale-up.

1. For properties that can be checked by reachability analysis we have proposed models based on regular languages and rational transductions. We have completed them by designing algorithms for verifying a refinement relation between two models $\mathcal{S}$ and $\mathcal{T}$ [55]. This refinement relation when satisfied preserves the safety properties and therefore allows them to be inherited. We shall investigate this approach with other representations.
2. In order to generate boundary-value functional test cases, we abstract models as constrained states. These constraints are solved by a customized solver, called CLPS. The test cases are derived in two steps [6]:
    1. partitioning of the formal model and extraction of boundary values,
    2. reachability graph exploration from constrained states in order to reach boundary values and generate state sequences (traces) as test cases with the oracle.

    After the generation phase, a concretization is used to produce the test drivers [7]. Furthermore, the kernel of the engine allows one to perform specification animations in order to validate the model [60].

3. For the safety of infinite state systems we have designed automated deduction tools based on term rewriting (*SPIKE*, *daTac*, *haRVey*) and an extensible and modular platform for detecting flaws and potential attacks on security protocols (*AVISPA*). The tools have been built on the modeling of systems by terms and rewrite rules. Our work with other models based on regular languages of words or trees and of transducers should complement these term rewriting models.

In order to address this challenge, we rely on complementary skills within the project. We believe that each of the three techniques will benefit from concepts and algorithms designed for the two others.

2.4. Highlights

1. We participate to two new european projects funded under FP7: AVANTSSAR — *Automated validation of trust and security of service-oriented architectures* (STREP Project) and SEES — *Software Engineering for lifelong Evolvable Systems* (IP Project).
2. Our former PhD students Y. Boichut and J.-F. Couchot have been hired as assistant professors at University of Orléans and University of Franche-Comté respectively. F. Bouquet has been promoted to a Professor position.

3. Scientific Foundations

3.1. Introduction

Our main goal is to design techniques and to develop tools for the verification of (safety-critical) systems, such as programs or protocols. To this end, we develop a combination of techniques based on automated deduction for program verification, constraint resolution for test generation, and reachability analysis for the verification of infinite state systems.

3.2. Automated Deduction

The main goal is to prove the validity of assertions obtained from program analysis. To this end, we develop techniques and automated deduction systems based on rewriting and constraint solving. The verification of recursive data structures relies on inductive reasoning or the manipulation of equations and it also exploits some form of reasoning modulo properties of selected operators (such as associativity and/or commutativity).

Rewriting, which allows us to simplify expressions and formulae, is a key ingredient for the effectiveness of many state-of-the-art automated reasoning systems. Furthermore, a well-founded rewriting relation can be also exploited to implement reasoning by induction. This observation forms the basis of our approach to inductive reasoning, with high degree of automation and the possibility to refute false conjectures.

The constraints are the key ingredient to postpone the activity of solving complex symbolic problems until it is really necessary. They also allow us to increase the expressivity of the specification language and to refine theorem-proving strategies. As an example of this, the handling of constraints for unification problems or for the orientation of equalities in the presence of interpreted operators (e.g., commutativity and/or associativity function symbols) will possibly yield shorter automated proofs.

Finally, decision procedures are being considered as a key ingredient for the successful application of automated reasoning systems to verification problems. A decision procedure is an algorithm capable of efficiently deciding whether formulae from certain theories (such as Presburger arithmetic, lists, arrays, and their combination) are valid or not. We develop techniques to build and combine decision procedures for the domains which are relevant to verification problems. We also perform experimental evaluation of the proposed techniques by combining propositional reasoning (implemented by means of Boolean solvers – Binary Decision Diagrams or SAT solvers) and decision procedures, and their extensions to semi-decision procedures for handling larger (possibly undecidable) fragments of first-order logic.

We investigate techniques to incorporate the use of decision procedures in the model-checking of infinite state systems. The state of such systems is described by the models of theories specifying data types (such as integers or arrays) and their behavior is identified by (possibly infinite) sequences of these models which share the interpretation of the symbols interpreted in the theories (e.g., the addition over the integers). In this context, checking if a system satisfies a certain property may be reduced to checking the satisfiability of a formula in the theory obtained as the combination of the theories describing the sequence of states in the computation. To solve this problem, it is crucial to develop new combination methods for non-disjoint unions of theories.

3.3. Synthesizing and Solving Set Constraints

Applying constraint logic programming technology in the validation and verification area is currently an active way of research. It usually requires the design of specific solvers to deal with the description language's vocabulary. We are interested in using a solver for set constraints based on the CLPS core [2], to evaluate set-oriented formal specifications. By evaluation, we mean the encoding of the formal model into a constraint system, and the ability for the solver to verify the invariant on the current constraint graph, to propagate preconditions or guards, and to apply the substitution calculus on this graph. The constraint solver is used for animating specifications and automatically generating abstract test cases.

3.4. Rewriting-based Safety Checking

Invariant checking and strenghtening is the dual of reachability analysis, and can thus be used for verifying safety properties of infinite-state systems. In fact, many infinite-state systems are just parameterized systems which become finite state systems when parameters are instantiated. Then, the challenge is to automatically discharge the maximal number of proof obligations coming from the decomposition of the invariance conditions. For parameterized systems, we develop a deductive approach where states are defined by first order formulae with equality, and proof obligations are checked by the automatic theorem prover *haRVey*. Thanks to this tool, we study the applicability of the superposition calculus (a modern version of resolution with a built-in treatment of the equality predicate and powerful techniques for reducing the search space) for deciding conditions arising from program verification.

4. Application Domains

4.1. Verification of Security Protocols

Security protocols such as SET, TLS and Kerberos, are designed for establishing the confidence of electronic transactions. They rely on cryptographic primitives, the purpose of which is to ensure integrity of data, authentication or anonymity of participants, confidentiality of transactions, etc.

Experience has shown that the design of those protocols is often erroneous, even when assuming that cryptographic primitives are perfect, i.e., that an encoded message cannot be decrypted without the appropriate key. An intruder can intercept, analyze and modify the exchanged messages with very few computations and therefore, for example, generate important economic damage.

Analyzing cryptographic protocols is complex because the set of configurations to consider is very large, and can even be *infinite*: one has to consider any number of sessions, any size of messages, sessions interleaving, some algebraic properties of encryption or data structures.

Our objective is to automatize as much as possible the analysis of protocols starting from their specification. This consists in designing a tool easy to use, permitting to specify a large number of protocols thanks to a standard high-level language, and permitting either to look for flaws in a given protocol or to check whether it satisfies a given property. Such a tool is essential for verifying existing protocols, but also for helping in designing new ones. For our tool to be easy to use, it has to provide a graphical interface allowing a user to do only click-button.

Our tools for verifying security protocols are available as components of the AVISPA platform. As an extension of the AVISPA specification language, we are working on a new environment called *CASRUL* for handling more general protocols like e-business protocols for example.

4.2. Automated Boundary Testing from Formal Specifications

In [7], we have presented a new approach for test generation from set-oriented formal specifications: the BZ-TT method. This method is based on Constraint Logic Programming (CLP) techniques. The goal is to test every operation of the system at every boundary state using all input boundary values of that operation. It has been validated in several industry case studies for smart card OS and application validation (GSM 11-11 standard [56] and Java Card Virtual Machine Transaction mechanism [59]) and for embedded automotive software (an automobile wind-screen wiper controller).

This test generation method can be summed up as follows: from the formal model, the system computes boundary values to create boundary states; test cases are generated by traversal of the state space with a preamble part (sequences of operations from the initial state to a boundary state), a body part (critical invocations), an identification part (observation and Oracle state computation) and a post-amble part (return path to initial or boundary state). Then, an executable test script file is generated using a test pattern and a table of correspondence between abstract operations (from the model) and concrete ones. This approach differs in several main points from the work of Dick, Faivre *et al*: first, using boundary goals as test objectives avoids the complete construction of the reachability graph; second, this process is fully automated and the test engineer could just drive it at the boundary value computation level or for the path computation.

The BZ-TT method is fully supported by the BZ-Testing-Tools tool-set. This environment is a set of tools dedicated to animation and test cases generation from B, Z or State-Chart formal specifications. It is based on the CLPS constraint solver, able to simulate the execution of the specification. By execution, we mean that the solver computes a so-called constrained state by applying the pre- and post-condition of operations. A constrained state is a constraint store where state variables and also input and output variables support constraints.

One orientation of the current work is to go beyond the finiteness assumption limitations by using symbolic constraint propagation during the test generation process and to extend the result to object oriented specifications.

4.3. Program Debugging and Verification

Catching bugs in programs is difficult and time-consuming. The effort of debugging and proving correct even small units of code can surpass the effort of programming. Bugs inserted while “programming in the small” can have dramatic consequences for the consistency of a whole software system as shown, e.g., by viruses which can spread by exploiting buffer overflows, a bug which typically arises while coding a small portion of code. To detect this kind of errors, many verification techniques have been put forward such as static analysis and software model checking.

Recently, in the program verification community, there seems to be a growing demand for more declarative approaches in order to make the results of the analysis readily available to the end user. To meet this requirement, a growing number of program verification tools integrate some form of theorem proving.

The goals of our research are twofold. First, we perform theoretical investigations of various combinations of propositional and first-order satisfiability checking in order to automate the theorem proving activity required to solve a large class of program analysis problems which can be encoded as first-order formulae. Second, we experimentally investigate how our techniques behave on real problems so to make program analysis more precise and scalable. Building tools capable of providing a good balance between precision and scalability is one of the crucial challenges to transfer theorem proving technology to the industrial domains.

4.4. Towards New Application Domains

4.4.1. Web Services

Driven by rapidly changing requirements and business needs, IT systems and applications are undergoing a paradigm shift: components are replaced by services, distributed over the network, and composed and reconfigured dynamically in a demand-driven way into service-oriented architectures [1]. Exposing services in future network infrastructures means a wide range of trust and security issues need to be adressed. Solving them is extremely hard since making the service components trustworthy is not sufficient: composing services leads to new subtle and dangerous vulnerabilities due to interference between component services and policies, the shared communication layer, and application functionality. Thus, one needs validation of both the service components and their composition into secure service architectures. In this context, there is an obvious need of applying formal methods. Our project aims at applying our proof and constraint solving techniques to reason on web services. More precisely, we plan to focus on the composition problem in the presence of security policies.

4.4.2. Microrobotics

Researchers in microrobotics have recently proposed the concept of a distributed and integrated micromanipulator called *smart surface*, based on an array of smart micromodules in order to realize an automated positioning and conveying surface. Each micro-module will be composed of a micro-actuator, a micro-sensor and a control unit. The cooperation of these micromodules will allow to recognize the parts and to control micro-actuators on order to move and position accurately the parts on the smart surface.

Our objective is to elaborate new specification languages and verification methods to validate distributed smart surfaces at different levels of abstraction. We bring our experience in formal verification, more especially in regular model-checking (RMC). This paradigm has been studied on classical regular languages, on regular tuples of words and on regular trees. We have a good experience on these different domains. To our knowledge, there has been no attempt of applying this approach to two-dimensional (picture) languages as required for the application. Therefore, an interesting challenge is to determine how far we can follow the RMC paradigm on (regular) picture languages. In order to cope with the parametric aspect of the smart surface, we will also consider constraint propagation on formulas representing sets of configurations.

We collaborate with the AS2M (Automatique et Systèmes Micro-Mécatroniques) department at the FEMTO-ST (Franche-Comté Electronique Mecanique Thermique et Optique - Sciences et Technologies) institute (UMR 6174) on verifying and validating an adaptative *microfactory* model they have developed. We have defined a complete information model of multi-cells microfactories in UML. This model is used as the communication basis between the robotic and computing researchers. It includes the structure of the physical components of the microfactory - cells and transports functions - and the logical components - information gathering and exchange. The next step will be to provide properties and a dynamic model of microfactories.

5. Software

5.1. Protocols Verification Tools

Keywords: *Cryptography*, *Security Protocols*, *Verification*.

Participants: Pierre-Cyrille Héam, Olga Kouchnarenko, Michaël Rusinowitch, Mathieu Turuani, Laurent Vigneron.

[1] see e.g. http://osoa.org/display/Main/Service+Component+Architecture+Home

5.1.1. AVISPA

Cassis has been one of the 4 partners involved in the European project AVISPA, which has resulted in the distribution of a tool for automated verification of security protocols, named *AVISPA* Tool. It is freely available on the web [2] and supported. The *AVISPA* Tool compares favourably to related systems in scope, effectiveness, and performance, by (i) providing a modular and expressive formal language for specifying security protocols and properties, and (ii) integrating 4 back-ends that implement automatic analysis techniques ranging from *protocol falsification* (by finding an attack on the input protocol) to *abstraction-based verification* methods for both finite and infinite numbers of sessions.

In 2008, no new release of the AVISPA Tool has been delivered, but the users mailing-list has been active and an important contribution has been proposed by Thomas Genet (LANDE Project, IRISA), SPAN, a protocol animator.

The tool has also been used in the group for analyzing non-repudiation protocols.

5.1.2. CL-AtSe

We develop, as a first back-end of *AVISPA*, *CL-AtSe*, a Constraint Logic based Attack Searcher for cryptographic protocols. The *CL-AtSe* approach to verification consists in a symbolic state exploration of the protocol execution, for a bounded number of sessions. This necessary restriction (for decidability, see [65]) allows *CL-AtSe* to be correct and complete, i.e., any attack found by *CL-AtSe* is a valid attack, and if no attack is found, then the protocol is secure for the given number of sessions. Each protocol step is represented by a constraint on the protocol state. These constraints are checked lazily for satisfiability, where satisfiability means reachability of the protocol state. *CL-AtSe* includes a proper handling of sets (operations and tests), choice points, specification of any attack states through a language for expressing fairness, non-abuse freeness, etc..., advanced protocol simplifications and optimizations to reduce the problem complexity, and protocol analysis modulo the algebraic properties of cryptographic operators such as XOR (exclusive or) and Exp (modular exponentiation). The handling of XOR and Exp has required to implement an optimized version of the combination algorithm of Baader & Schulz [54] for solving unification problems in disjoint unions of arbitrary theories.

CL-AtSe has been successfully used by Cassis members to analyse France Telecom R&D, Siemens AG, IETF, or Gemalto protocols in funded projects. It is also employed by external users, e.g., from the AVISPA's community. Moreover, *CL-AtSe* achieves very good analysis times, comparable and sometimes better than state-of-the art tools in the domain (see [69] for tool details and precise benchmarks).

5.1.3. TA4SP

We have developed, as a second back-end of *AVISPA*, TA4SP (Tree Automata based on Automatic Approximations for the Analysis of Security Protocols), an automata based tool dedicated to the validation of security protocols for an unbounded number of sessions. This tool provides automatic computations of over and under approximations of the knowledge accessible by an intruder. This knowledge is encoded as a regular tree language and protocol steps and intruder abilities are encoded as a term rewriting system. When given a reachability problem such as secrecy, TA4SP reports that (1) the protocol is safe if it manages to compute an over-approximation of intruder's knowledge that does not contain a secret term or (2) the protocol is unsafe in the rewrite model if it manages to compute an underapproximation of intruder's knowledge containing a secret term or (3) I don't know otherwise. TA4SP has verified 28 industrial protocols and case (3) occurred only once, for Kaochow protocol version 2.

TA4SP handles protocols using operators with algebraic properties. Thanks to a recent quadratic completion algorithm new experimental results have been obtained, for example for the Encrypted Key Exchange protocol (EKE2) using the exponential operator.

5.2. Testing Tools

Keywords: *Animation of Specifications*, *CLP*, *Formal Specification*, *Test generation*.

[2] http://www.avispa-project.org

Participants: Fabrice Bouquet, Frédéric Dadeau.

The Testing Tools is a tool-set for animation and test generation from B, JML, Z and State-chart specifications. It consists of two components:

- **BZ-Testing-Tools**[3] – BZ-TT – is a tool-set for animation and test generation from B, Z and State-chart specifications. BZ-TT provides several testing strategies (partition analysis, cause-effect testing, boundary-value testing and domain testing), and several test model coverage criteria (multiple condition coverage, boundary coverage and transition coverage).
- **JML-Testing-Tools**[4] – JML-TT – is a framework for the symbolic animation of formal models written using JML annotations [68] embedded within Java programs. JML-TT provides a simple and efficient way to semi-automatically validate a JML specification and to check model properties such as class invariant or history constraints during the animation. This tool is used in the ACI GECCOO project[5].

We develop a third tool **Test-For-Testing-Tools** to valid the tests. The tool takes as input a code program and a test suite (realized by several approaches such as BZ-TT/random/properties driven tests). The system performs a mutation of the code program and we observe how many mutants are killed with each test suite.

5.3. Automated Deduction Tools: haRVey

Keywords: *Automated Deduction*, *Boolean Reasoning*, *Equational Reasoning*, *Satisfiability*, *Saturation Theorem Proving*.

Participants: Alain Giorgetti, Silvio Ranise, Christophe Ringeissen.

haRVey[6] is a solver dedicated to satisfiability problems modulo theories. The main feature of *haRVey* is its capability of behaving as a decision procedure for the problem of checking the validity of certain classes of first-order formulae modulo some (combination of) theories of relevance in verification. The system features a combination of Boolean reasoning (supplied by a BDD or a SAT solver) to efficiently handle the boolean structure of formulae and a (generalization of the) Nelson-Oppen combination method between superposition theorem proving and decision procedures for linear arithmetic. The first version, called *haRVey*-FOL has been designed by Silvio Ranise and David Déharbe (UFRN Natal, Brazil). The new version, called *haRVey*-SAT, is developed by P. Fontaine (project-team MOSEL) and David Déharbe (UFRN Natal, Brazil). *haRVey* has been especially designed to be integrated in larger verification systems. It is integrated in Barvey,a tool to check the consistency of B specifications. It takes a B abstract machine as input, generates proof obligations encoding the fact that the invariant is inductive, and translates them into a validity problem that *haRVey* can discharge.

5.4. Others Tools

Most of the software tools described in previous sections are using tools that we have developed in the past: BZ-TT uses the set constraints solver CLPS and *SPIKE*, our induction-based theorem prover, is used in the system VOTE in collaboration with the ECOO project.

6. New Results

6.1. Automated Deduction

Keywords: *Consistency*, *Decision Procedure*, *Proof*, *Satisfiability*, *Tree Automata*.

[3] http://lifc.univ-fcomte.fr/~bztt
[4] http://lifc.univ-fcomte.fr/~jmltt
[5] http://geccoo.lri.fr
[6] http://harvey.loria.fr

We develop general techniques which allow us to re-use available tools in order to build a new generation of satisfiability solvers offering a good trade-off between expressiveness, flexibility, and scalability. We focus on the careful integration of combination techniques and rewriting techniques to design satisfiability procedures for a wide range of (combined) theories of interest in verification.

6.1.1. Decision Procedures and their Extensions

Participants: Enrica Nicolini, Silvio Ranise, Christophe Ringeissen, Michaël Rusinowitch, Daniele Zucchelli.

In [52], we develop a framework to design cooperation schemas between satisfiability procedures which allows us to maintain the modularity of their interfaces. In particular, we introduce the concept of deduction complete satisfiability procedures: we show how to build them for large classes of theories and we provide a schema to modularly combine them. Then, we consider the problem of modularly constructing explanations for combinations by re-using available proof-producing procedures for the component theories. To solve this problem, we introduce a proof-producing refinement of the Nelson-Oppen method and we study how the computed explanations relate to an appropriate notion of minimality.

In [51], we present a novel technique to combine satisfiability procedures for theories that model some data-structures and that share the integer offsets. This procedure extends the Nelson-Oppen approach to a family of non-disjoint theories that have practical interest in verification. The result is derived by showing that the considered theories satisfy the hypotheses of a general result on non-disjoint combination. In particular, the capability of computing logical consequences over the shared signature is ensured in a non trivial way by devising a suitable complete superposition calculus.

6.1.2. Decision Procedures and Model-checking of Infinite State Systems

Participants: Enrica Nicolini, Silvio Ranise, Daniele Zucchelli.

Daniele Zucchelli has defended his thesis [10]. The contributions of the thesis are the following: First of all, we give a decidability result for the constraint satisfiability problem for interesting extensions of the theory of arrays. Secondly, along the lines of Manna and Pnueli, who have shown how a mixture of first-order logic and linear time temporal logic is sufficient to state the verification problems for the class of reactive systems, we draw on the recent literature about the combination of decision procedures to give decidability and undecidability results for the satisfiability problem for logics that allow to plug reasoning modulo first-order theories into a temporal setting. The results obtained in the case of linear flows of time are then generalized to the temporal and modal logics whose relativized satisfiability problem is decidable. The last contribution is the decidability of the model checking problem for linear flows of time under suitable hypothesis over the first-order theories involved. The proofs of the decidability results suggest that efficient Satisfiability Modulo Theories solvers might be successfully employed in the model checking of infinite-state systems.

In [32], we introduce the notion of array-based system as a suitable abstraction of infinite state systems such as broadcast protocols or sorting programs. By using a class of quantified-first order formulae to symbolically represent array-based systems, we propose methods to check safety (invariance) and liveness (recurrence) properties on top of Satisfiability Modulo Theories solvers. We find hypotheses under which the verification procedures for such properties can be fully mechanized.

6.1.3. Hypothesis Selection

Participant: Alain Giorgetti.

In deductive verification of large C programs by SMT provers, some valid verification conditions cannot be automatically discharged by any automated prover mainly due to their size and a high number of irrelevant hypotheses. At the FTP'07 workshop, Couchot and Hubert have presented heuristics for relevant hypothesis selection. We extend these heuristics to axioms and comparison operators [49]. The relevance of a hypothesis is the combination of separated static dependency analyzes based on graph constructions and traversals. The approach is applied on two benchmarks issued from industrial program verification.

6.1.4. Tree Automata Extensions

Participants: Michaël Rusinowitch, Laurent Vigneron.

We have considered classes of tree automata combining automata with equality test and automata modulo equational theories with F. Jacquemard (DAHU project-team) [18]. These tree automata are obtained by extending their standard Horn clause representations with equational conditions and rewrite systems. We show in particular that a generalized membership problem (extending the emptiness problem) is decidable by proving that the saturation of tree automata presentations with suitable paramodulation strategies terminates. Alternatively our results can be viewed as new decidable classes of first-order formula. These tree automata classes can be applied to the reachability problem for a fragment of pi-calculus that can encode protocol verification problems.

6.2. Security Protocol Verification

Keywords: *Exclusive-Or*, *Exponentiation*, *Protocol*, *Security*, *Verification*.

Cryptographic protocols are successfully analyzed using formal methods and many techniques have appeared in the litterature [57]. However, formal approaches usually consider the encryption schemes as black boxes and assume that an adversary cannot learn anything from an encrypted message except if he has the key. Such an assumption is too strong in general since some attacks exploit in a clever way the interaction between protocol rules and properties of cryptographic operators.

6.2.1. Extension of the Dolev-Yao Model

Participants: Yannick Chevalier, Michaël Rusinowitch, Mathieu Turuani.

Some attacks exploit in a clever way the interaction between protocol rules and algebraic properties of cryptographic operators. In [64], we provide a list of such properties and attacks as well as existing formal approaches for analyzing cryptographic protocols under algebraic properties.

When modelling protocol steps as rigid Horn clauses, and the intruder abilities as an equational theory over a convergent rewrite system, the insecurity problem (for active intruder and a bounded number of sessions) can be interpreted as a Cap Unification problem which is an extension of Equational Unification: we look for a cap i.e. a context to be placed on a given set of terms, so that it unifies with a given term modulo the equational theory. With that approach, simpler proofs for the case of subterm convergent theories can be derived [43].

Symbolic Derivations. We have also continued the work on the symbolic derivation model for cryptographic protocols that was introduced in [63]. We were in particular interested by the problem of whether two distinct symbolic derivations have the same sets of solutions. We have obtained a preliminary decidability result for the syntactic Dolev-Yao intruder model case.

6.2.2. Soundness of the Dolev-Yao Model

Participants: Véronique Cortier, Mathieu Turuani.

All the previous results rely on symbolic models of protocol executions in which cryptographic primitives are abstracted by symbolic expressions. This approach enables significantly simple and often automated proofs. However, the guarantees that it offers have been quite unclear compared to cryptographic models that consider issues of complexity and probability. Cryptographic models capture a strong notion of security, guaranteed against all probabilistic polynomial-time attacks.

We have shown in recent years that it is possible to obtain the best of both cryptographic and formal worlds in the case of public encryption: fully automated proofs and strong, clear security guarantees. Most recent results have concentrated on trace-based properties such as authentication or specific indistinguishability properties such as secrecy of nonces or secrecy of keys. We show in [28], [48], [45] that computational proofs of indistinguishability can be considerably simplified, for a class of processes that covers most existing protocols. More precisely, we show a soundness theorem, following the line of research launched by Abadi and Rogaway in 2000: computational indistinguishability in presence of an active attacker is implied by the *observational equivalence* of the corresponding symbolic processes.

6.2.3. Securely Composing Protocols

Participant: Véronique Cortier.

Even when a protocol has been proved secure, there is absolutely no guarantee if the protocol is executed in an environment where other protocols, possibly sharing some common identities and keys like public keys or long-term symmetric keys, are executed. In [13], we show that security of protocols can be easily composed. More precisely, we show that whenever a protocol is secure, it remains secure even in an environment where arbitrary protocols are executed, provided each encryption contains some tag identifying each protocol, like e.g. the name of the protocol.

Protocols may also be built in a modular way. For example, authentication protocols may assume pre-distributed keys or may assume secure channel. How security of these protocols can be combined is an important issue. Stefan Ciobaca has started a PhD on this subject this year, in collaboration with the project-team SECSI (LSV, Cachan). He is also working on developing new techniques for analyzing e-voting protocols.

6.2.4. Security Properties and Advanced Class of Protocols

Participants: Tigran Avanesov, Najah Chridi, Véronique Cortier, Michaël Rusinowitch, Laurent Vigneron.

Most previous results focus on secrecy and authentication for simple protocols like the ones from Clark & Jacob library. We explore several directions to cover more complex protocols and security properties.

Security Properties. Non-repudiation protocols have an important role in many areas where secured transactions with proofs of participation are necessary. Formal methods are clever and without error, therefore using them for verifying such protocols is crucial. In this purpose, in collaboration with F. Klay (France Telecom R&D), we have shown how to partially represent non-repudiation as a combination of authentications. Because of the limits of this method, we have defined a new one, based on the handling of the knowledge of protocol participants. This method is very general and is of natural use, as it consists in adding simple annotations, like for authentication problems. The method is very easy to implement in tools able to handle participants knowledge. We have implemented it in the AVISPA Tool and analyzed two protocols: the Fair Zhou-Gollmann protocol and the optimistic Cederquist-Corin-Dashti protocol, discovering attacks for both of them [40]. This extension of the AVISPA Tool for handling non-repudiation opens a highway to the specification of many other properties, without any more change in the tool itself.

SIP Analysis. The recent and massive deployment of Voice over IP infrastructures had raised the importance of the VoIP security and more precisely of the underlying signalisation protocol SIP. We have formalized a new attack found by MADYNES EPI against the authentication mechanism of SIP. This attack allows to perform toll fraud and call hijacking. We have shown how to derive this vulnerability with AVISPA, highlighted a simple usage case and proposed a mitigation technique [22].

Mathilde Arnaud has recently started a PhD, in collaboration with the project-team SECSI (LSV, Cachan) on designing verification techniques adapted for protocols on wireless networks.

6.2.5. Analysing Group Protocols

Participants: Najah Chridi, Michaël Rusinowitch, Mathieu Turuani.

Although many works have been dedicated to standard protocols, very few address the more challenging class of group protocols. We investigated group protocol analysis in a synchronous model, that allows the specification of unbounded sets of agents with related behavior. Also, when used in an asycronous way, this generalizes standard protocol models with bounded number of agents by permitting unbounded lists inside messages (including unbounded number of variables, nonces, etc..). In this extended model we proposed [44] a correct and complete set of inference rules for checking security properties in presence of an active intruder for the class of well-tagged protocols. This inference system generalizes the ones that are implemented in several tools for a bounded number of sessions and fixed size lists in message. In particular when applied to protocols whose specification does not contain unbounded lists, this provides a decision procedure for secrecy in the case of a fixed number of sessions.

6.3. Model-based Verification

Keywords: *Model*, *Verification*.

We have investigated extensions of regular model-checking to new classes of rewrite relations on trees. We have applied model-checking to collaborative editors and studied generation of JML annotations for liveness properties. Finally we have proposed test-generation techniques from behaviours or scenarii.

6.3.1. Safety Verification Techniques with Regular Fixpoint Computations

Participants: Roméo Courbis, Pierre-Cyrille Héam, Olga Kouchnarenko.

Term rewriting systems are now commonly used as a modelling language for programs or systems. On those rewriting based models, reachability analysis, i.e. proving or disproving that a given term is reachable from a set of input terms, provides an efficient verification technique. Many recent works have shown the relevance of regular approximation techniques to tackle in practice undecidable reachability problems.

In [15] we show a theoretical limit of regular fixpoint-based techniques pointing out a regular tree language I, a left-linear term rewriting system R and a term t such that $t \notin R^*(I)$ and t is in every regular over-approximation of $R^*(I)$. Hence, it is not possible to prove $t \notin R^*(I)$ by using regular over-approximations.

In [24], we improve an over-approximation approach initially developed in [66] to check the reachability of terms. Given a term t, we try to compute an over-approximation which does not contain t by refining the approximation. If the approximation refinement fails then t is a reachable term. This semi-algorithm has been prototyped in the Timbuk tool. The above technique works for linear term-rewriting systems. This approach has been extended to left-linear term rewriting systems using results in [66]. However it requires to perform some determinisation steps with an exponential time and space complexity and it is therefore practically unfeasible. We address this problem for left-quadratic rules by proposing in [25] an algorithm replacing determinisation steps by polynomial time constructions on involved automata. It should be noticed that many industrial specifications give rise to non-left linear rules that are left-quadratic ones.

6.3.2. Partially Ordered Tree Automata

Participant: Pierre-Cyrille Héam.

Computing images of regular languages by transitive closures of semi-commutation relations (i.e. of the form $ab \rightarrow ba$) is a quite old trace theory problem that has been recently revisited in a regular model-checking context. The work in [58] shows that if a regular word language is accepted by a partially ordered automaton (or equivalently by a Σ_2 formula in First Order Logic), then its image by the transitive closure of semi-commutation relation is computable, regular and also accepted by a partially ordered automaton. We extended in [16] this result to a larger class of regular languages and by proposing a better computation algorithm. We investigate in [17] whether results can be extended to tree data structures. We show that the class of tree languages accepted by Σ_2 formulae on trees is strictly included in the class of tree languages accepted by partially ordered tree automata. Moreover, we point out a regular tree language K accepted by a Σ_2 formula and a semi-commutation relation R such that $R^*(K)$ is not regular.

6.3.3. Regular Model-Checking with Hedges

Keywords: *Hedge automata*, *reachability*, *regular languages*.
Participant: Michaël Rusinowitch.

We consider in collaboration with F. Jacquemard (DAHU project) [39] rewriting systems for unranked ordered terms, i.e. trees where the number of successors of a node is not determined by its label, and is not a priori bounded. The rewriting systems are defined such that variables in the rewrite rules can be substituted by hedges (sequences of terms) instead of just terms. Consequently, this notion of rewriting subsumes both standard term rewriting and word rewriting. We investigate some preservation properties for two classes of languages of unranked ordered terms under this generalization of term rewriting. The considered classes include languages of hedge automata (HA) and some extension (called CF-HA) with context-free languages in transitions, instead of regular languages. In particular, we show that the set of unranked terms reachable from a given HA language, using a so called inverse context-free rewrite system, is a HA language. Moreover, we prove that the closure of CF-HA languages with respect to restricted context-free rewrite systems, the symmetric case of the above rewrite systems, is a CF-HA language. As a consequence, the problems of ground reachability and regular hedge model checking are decidable in both cases. Several counterexamples show that we cannot relax the restrictions.

6.3.4. Model-Checking Optimistic Replication Algorithms

Participant: Abdessamad Imine.

We consider in collaboration with Hanifa Boucheneb (Professor at Ecole Polytechnique de Montréal, Canada) automatic verification of optimistic replication algorithms, based on the Operational Transformation (OT) approach, that are mostly used for supporting collaborative edition [47]. Using the UPPAAL Model Checker, we formally define the behavior and the main consistency requirement (*i.e. convergence property*) of the collaborative editing systems, as well as the abstract behavior of the environment where these systems are supposed to operate. Two models are proposed. The first one, called *concrete model*, is very close to the system implementation but runs up against a severe explosion of states. The second model, called *symbolic model*, aims to overcome the limitation of the concrete model by delaying the effective selection and execution of editing operations until the construction of symbolic execution traces of all sites is completed. Experimental results have shown that the symbolic model allows a significant gain in both space and time. Using the symbolic model, we have been able to show that if the number of sites exceeds 2 then the convergence property is not satisfied for all OT algorithms considered here. A counterexample is provided for every algorithm.

6.3.5. Liveness Properties

Participants: Alain Giorgetti, Olga Kouchnarenko.

In joint work with J. Groslambert (Trusted Labs) we address static checking of liveness properties via JML annotations [14]. Static checking is essential for the security of software components. As a component model, we consider a Java class enriched with annotations from the Java Modeling Language (JML). We define a formal execution semantics for repetitive method invocations from this annotated class, called the class in isolation semantics. A pattern of liveness properties is defined, together with its formal semantics, providing a foundation for both static and runtime checking. This pattern is then inscribed in a complete language of temporal properties, called JTPL (Java Temporal Pattern Language), extending JML. We particularly address the verification of liveness properties by automatically translating the temporal properties into JML annotations for this class. Correctness of the generated annotations ensures that the temporal property is established for the executions of the class in isolation.

6.4. Model-based Testing

Keywords: *Model*, *Test*, *Verification*.

Our advances in Model-based testing are related to language modelisation and test generation with properties.

6.4.1. Test Generation from Behavioral Models

Participants: Fabrice Bouquet, Thibaud Brocard, Pierre-Christophe Bué, Kalou Cabrera, Jean-Franois Couchot, Frédéric Dadeau, Stéphane Debricon, Alain Giorgetti, Adrien de Kermadec, Jonathan Lasalle, Vincent Pretre.

We have introduced an original model-based testing approach that takes a UML behavioural view of the system under test and automatically generates test cases and executable test scripts according to model coverage criteria [26]. In parallel, we are working on the improvement of the test generation technique, by combining constraint solving and theorem proving, in order to detect inconsistencies in the behaviors extracted from the model, and to find a relevant instantiation of the initial test data.

A rebuild of the architecture of the BZ-Testing-Tools engine will start in december 2008, with the help of an "ingénieur jeune diplomé". It aims at integrating the latest works on constraint solving and theorem proving, in a modular architecture dedicated to the analysis and exploitation of formal behavioral models for test generation purposes.

6.4.2. Test Generation from Scenarii

Participants: Fabrice Bouquet, Pierre-Christophe Bué, Kalou Cabrera, Frédéric Dadeau, Adrien de Kermadec.

In the context of the RNTL POSE project[7], the team has developed and experimented a language describing test scenarii. Basically, a scenario is a regular expression describing sequences of operations calls (without specifying their possible parameters) along with intermediate states that have to be reached. Each scenario is unfolded and played using a symbolic animation engine, that instantiates the sequence. This approach has been experimented on the IAS case study of Gemalto, and also applied on a model of the POSIX standard [29].

In addition, we have defined conformance relationships dedicated to establishing a verdict when testing the correct implementation of security policies (namely access control policies) in smart cards applications [30], [31]. These conformance relationships are variants of input-output conformance and are based on the inclusion of traces of the implementation w.r.t. traces computed on a security-dedicated model, involving possible mappings between the values of these two levels.

6.4.3. Random Combination

Participants: Frédéric Dadeau, Pierre-Cyrille Héam.

We are also beginning experiments on the combination of random- and model-based testing. A first attempt has been done to automatically produce LTL formula using uniform random test generation. More recently, an approach has considered the automated generation of automata in order to evaluate various FSM-based test generation algorithms. A major result is the highlighting of an error in a widely-spread implementation of the chinese postman algorithm. We also proposed a test generation technique, driven by a final number of test cases, and combining random testing and model-based testing. It consists in arbitrarily augmenting a FSM in order to reach a given number of test cases when selected FSM-based test generation algorithms are applied. A realistic experiment has illustrated the efficiency of this approach. These works are summarized in [50].

6.5. Verification for Service Oriented Computing

We have investigated several specific verification problems related to the composition of services including security issues and quality of service.

6.5.1. Towards An Automatic Analysis of Web Services Security

Participants: Tigran Avanesov, Yannick Chevalier, Mohamed Anis Mekki, Michaël Rusinowitch, Mathieu Turuani, Laurent Vigneron.

Automatic composition of web services is a challenging task. Many works have considered simplified automata models that abstract away from the structure of messages exchanged by the services. For the domain of security services (such as digital signing or timestamping) we propose a novel approach to automated composition of services based on their security policies [27]. The approach amounts to collecting the constraints on messages, parameters and control flow from the components services and the goal service requirements. A constraint solver checks the feasability of the composition, possibly adapting the message structure, while preserving the semantics, and displays the service composition as a message sequence chart. Moreover the resulting composed service can be verified automatically (in Dolev Yao model) for ensuring that it cannot be subject to active attacks from intruders. The services that are input to our system are provided in a declarative way using a high level specification language. The approach is fully automatic and we show on a case-study how it succeeds in deriving a composed service that is currently proposed as a product by a company.

6.5.2. Composition of Web Services

Participants: Christophe Ringeissen, Laurent Vigneron.

In collaboration with the project-team ECOO, we work on a framework for Web services composition, including both temporal and security aspects. In [34], a composition of services is represented as a product of automata. Our solution is based on the synthesis of a mediator in order to mimic the awaited composition. The compatibility of services is a key issue for the composition problem studied in [35].

[7] http://www.rntl-pose.info

We are also working on applying constraint programming techniques to the composition problem [19]. In [41], we consider the provisioning problem of Web services. Our approach consists in instantiating a given abstract representation of a composite Web service by selecting the most appropriate concrete Web services. This instantiation is based on constraint programming techniques which allows us to match the Web services according to a given request. Our proposal performs this instantiation in a distributed manner, i.e., the solvers for each service type are solving some constraints at one level, and they are forwarding the rest of the request (modified by the local solution) to the next services. When a service cannot provision part of the composition, a distributed backtrack mechanism enables to change previous solutions (i.e., provisions). A major interest of our approach is to preserve privacy: solutions are not sent to the whole composition, services know only the services to which they are connected, and parts of the request that are already solved are removed from the next requests.

6.5.3. Access Control Policies for Web Services

Participant: Yannick Chevalier.

We focus on the problem of the dynamicity of access control, *i.e.* on their evolution over time. In order to devise a language for expressing access control policies we have abstracted the XACML standard to keep only a set of rules defining a static policy which is employed to decide whether an access is granted, and a dynamic policy expressing the changes in the access control system induced by users actions. This approach permits us to express in a simple language all concepts attached to access control. For this language, we have studied in [23] the complexity of several decision problems related to access control, in particular: decide whether in a given state a set of actions is permitted; decide whether there is a sequence of states, and a sequence of sets of actions, such that each set of permissions can be granted in the final state of the sequence.

6.5.4. Controlling Access in Distributed Collaborative Editors

Participants: Asma Berregba, Abdessamad Imine.

One of the most challenging problems in Distributed Collaborative Editors (DCE) is how to balance the computing goals of collaboration and access control to shared information. In this work, we propose a reliable access control scheme that is well suited for DCE [53]. We first define generic access control requests for manipulating linear objects, such as texts and HTML documents. To allow for dynamic policies, we use editing techniques to modify the access control policy. We show formally the correct concurrent behavior of every access control request with respect to editing requests. A prototype based on our concurrency control framework [38], [37] has been implemented for supporting the secure and collaborative editing of HTML pages. This prototype is deployed on P2P JXTA platform.

6.5.5. Formalising QoS of Web Services with Weighted Automata

Participants: Pierre-Cyrille Héam, Olga Kouchnarenko.

In [36], we focus on the identification of a relevant abstraction for the Web-services expression and verification of properties like substitutivity: When is it possible to formally accept or reject the substitution of a Web-service in a composition? This work uses max/plus automata to tackle this problem when considering a new factor – Quality of Service (QoS). Four notions of *simulation*-based substitutivity managing QoS aspects are proposed, and related complexity issues on max/plus automata are investigated. This work extends the previous work on *trace*-based substitutivity [67], where a translation from Web service BPEL/WSDL specifications extended with QoS into max/plus automata was given.

6.5.6. Web Services Validation

Participants: Fabrice Bouquet, Vincent Pretre.

In order to validate Web Services applications, we explore model-based testing methodologies combined with common criteria. The results of tests are used to compute a mark that qualifies the quality of web services operations. This solution is then integrated in a validation framework based on an UDDI server. In this framework, named iTac-QoS,Web Services are tested when they are declared to the UDDI server, and the obtained marks are supplied to customers looking for services. We propose an original approach to take into account the composition of Web services from their models as described in [42].

7. Contracts and Grants with Industry

7.1. Research Result Transfer

The BZ-Testing-Tools technology has been transfered to LEIRIOS Technologies, at the end of 2004. The partnership between the Cassis project and the R&D LEIRIOS Department, located at the TEMIS Scientific and Industrial area at Besançon, will be continued through (national and international) projects or with a new transfer protocol. According to the law of innovation, F. Bouquet is scientific consultant of LEIRIOS Technologies.
Serge Burckel, who joined CASSIS in 2007, is recepient of the “Concours national 2008 d’aide à la création d’entreprises de technologies innovantes” for his works on the optimization of computations and data transmissions [62], [61]. With Emeric Gioan (LIRMM) and Emmanuel Thomé (project-team CACAO), he investigates formal methods for the automatic design of arbitrary operations on registers with sequential and “in-place” procedures [33]. The resulting codes enable to save energy and time in processors. In the field of data exchanges, he proposes an encoding of binary informations with time intervals. Again, the goal is to reduce the energy used in networks.

7.2. European Projects

- AVANTSSAR — *Automated validation of trust and security of service-oriented architectures*. STREP Project funded under 7th FP (Seventh Framework Programme) Research area: ICT-2007.1.4 Secure, dependable and trusted infrastructures. The coordinator is the University of Verona (Italy) and the Cassis project is one of the 10 partners. AVANTSSAR aims to propose a rigorous technology for the formal specification and "Automated VAlidatioN of Trust and Security of Service-oriented ARchitectures". This technology will be automated into an integrated toolset, the AVANTSSAR Validation Platform, tuned on relevant industrial case studies.
- SEES — *Software Engineering for lifelong Evolvable Systems*. SEES is funded under the 7th FP (Seventh Framework Programme) Research area: ICT-2007.8.6: ICT forever yours. The project will develop processes and tools that support design techniques for evolution, testing, verification, reconfiguration and local analysis of evolving software. Our focus is on mobile devices and homes, which offer both great research challenges and long-term business opportunities. The project is leaded by Fabio Massacci (University of Trento, Italy) and it is expected to start at the beginning of 2009 for a period of 36 months.

7.3. INTERREG

INTERREG TEST-INDUS— We are working with the university of Geneva, SMARTESTING Technologies and CLIO SA. The project concerns the test generation in industrial process. The consortium will propose methods, techniques and tools to integrate (model-based) testing into industrial process. The duration of the project is 18 months and started in May 2008.

8. Other Grants and Activities

8.1. International Grants

- Project INRIA-CNPq (Brazil), DA CAPO — *Automated deduction for the verification of specifications and programs*. This is a project on the development of proof systems (like *haRVey*) for the verification of specifications and software components. The coordinators are David Déharbe (UFRN Natal, Brazil) and Christophe Ringeissen. On the french side, DA CAPO also involves the project-team MOSEL and the former project-team PROTHEO.

- Project INRIA-CONICYT (Chile), CoreWeb — *Constraint Reasoning for the Composition of Web Services*. The coordinators are Eric Monfroy (UTFSM Valparaíso, Chile) and Michaël Rusinowitch. On the french side, CoreWeb also involves the project-team ECOO.
- Associate Team INRIA (with UTFSM Valparaíso, Chile), VanaWeb — *Hybrid and autonomous constraint solving and applications to composition problems for the Web*. The coordinators are Carlos Castro (UTFSM Valparaíso, Chile) and Christophe Ringeissen. On the french side, VanaWeb also involves the project-team ECOO and members of the former project-team PROTHEO.
- French-Tunisian project on *Security Policies and Configurations of Firewalls: Compilation and Automated Verification*. We collaborate with SupCom Tunis and the project-team DAHU in the context of STIC-Tunisia.
- French-Tunisian project on the design and implementation of e-voting systems and of tools for verifying e-voting protocols. Duration: 2 years, started in January 2007. This is a project founded by the INRIA/DGRST, action STIC-Tunisia.
- PHC Alliance project between the Cassis team and the University of Bristol on refinement of security systems. The coordinators of the projet are Bogdan Warinschi and Véronique Cortier. Duration: 2 years, started in January 2008.

8.2. National Grants

- ARA SSIA FormaCrypt—*Formal proofs and probabilistic semantics in cryptography*, duration: 3 years, started in January 2006. The verification of cryptographic protocols is a very active research area. Most works on this topic use either the computational approach, in which messages are bitstrings, or the formal approach, in which messages are terms. The computational approach is more realistic but more difficult to automate. The FormaCrypt project aims at bringing together these orthogonal approaches in order to get the best of the two worlds. Partners are: Liens (coordinator), project-team SECSI - LSV, Cachan.
- ARA SSIA COPS—*Composition Of Policies and Services*, duration: 3 years, started in December 2005. The aim is to build technologies enabling the security analysis of web services that take into account the potential flaws at communication level, at the access policy level or at the interface between communications and access policy. Partners are: IRIT Toulouse, LIM Marseille, Microsoft R&D.
- ARA SSIA ARROWS—*Safe Pointer-Based Data Structures: A Declarative Approach to their Specification and Analysis*, duration: 3 years, started in autumn 2005. The goal of this project is to develop new specification languages for programs manipulating pointers which are sufficiently precise to express many interesting properties and, at the same time, support automatic analyses. Partners are: CAPP-LEIBNIZ Grenoble (coordinator), LILaC-Irit Toulouse. The local coordinator is S. Ranise.
- ARA SETI RAVAJ [8] — *"Rewriting and Approximations for Java Applications Verification"*, duration: 39 months, started on January 2007. The goal of this project is to analyse MIdlets – Java programs designed for mobile devices like cell phones or PDA. In addition to classical proof tools of rewriting, we propose to use approximations of reachable terms. There are three academics partners: INRIA LANDE, INRIA PROTHEO and LIFC/Besançon; and an industrial: France Telecom R&D. The local coordinator is O. Kouchnarenko.
- ANR SESUR AVOTÉ—*Formal Analysis of Electronic-Voting protocols*, duration: 3 years, started in January 2008. Electronic voting promises the possibility of a convenient, efficient and secure facility for recording and tallying votes. However, the convenience of electronic elections comes with a risk of large-scale fraud. The AVOTÉ project aims at proposing techniques for formally analyzing e-voting protocols. The coordinator of the project is the Cassis team. Partners are: France Telecom Lannion, LSV Cachan, Verimag Grenoble.

[8] http://www.irisa.fr/lande/genet/RAVAJ/index.html

- ANR program “Systèmes interactifs et robotique”— *Smart Surface*, coordinated by AS2M (Automatique et Systèmes Micro-Mécatroniques) department at the FEMTO-ST (Franche-Comté Electronique Mecanique Thermique et Optique - Sciences et Technologies) institute (UMR 6174). This project started in July 2007 for three years. The CASSIS participant is A. Giorgetti.
- ANR DECERT — *Deduction and Certification*, coordinated by Th. Jensen (IRISA). This project focuses on the design of decision procedures, in particular for fragments of arithmetic, and their integration into larger verification systems, including skeptical proof assistants. Partners are: IRISA Rennes, LRI Orsay, INRIA Sophia, Systerel and CEA. From INRIA Nancy, MOSEL and CASSIS project-teams are involved. This project will start in January 2009 for three years.
- Competitiveness pole — *Microtechnique* and FUI [9] Project VALMI - *Validation automatique de microsystèmes embarqués de transaction électronique en billétique*. Duration : 18 months, started in November 2006. The aim of this project is to provide automated tools for generation tests of embedded system around distribution and validation of urban travel pass. There are four partners: ERG, Leirios, Parkeon and LIFC. The local coordinator is F. Bouquet.
- FCE Vetess — We are working with the university of Haute Alsace, SMARTESTING Technologies and PSA Citroen. The project is labelled by "pole de compétitivité Véhicule du Futur" and funded by the "Fonds de Compétitivité des Entreprises", an inter-ministry grant. It aims at verifying embedded systems vehicles by automatic model-based tests generation. The duration of the project is 18 months and started in September 2008.
- Collaborative Research Initiative INRIA, ARC CeProMi “Certification de Programmes manipulant la Mémoire”, coordinated by Claude Marché from the project-team PROVAL. This project started in 2008 for two years. The partners are the project-teams GALLIUM (François Pottier) and PROVAL (Claude Marché), and DCS Team (Marie-Laure Potet, Verimag, Grenoble). The local coordinator is Alain Giorgetti.
- SSS SeComMaNet—*Security of multicast communications in ad-hoc mobile networks*, duration: 2 years, started in January 2007. This action is coordinated by L. Vigneron. This is an action of the theme *Sûreté et Sécurité des Systèmes*, funded by the Project MISN of the *Contrat de Plan État-Région Lorraine 2007-2013*.

8.3. International Collaborations

- In the area of automated test generation from a formal model, we have an active collaboration with Dr Mark Utting from the Formal Method group from the University of Waikato [10]. This cooperation is supported by the France-New-Zealand scientific program.
- In the area of business applications, we are working on the soundness problem of coloured work-flow Petri nets with the Information System group of Professor K. van Hee from the Technical University of Eindhoven. This cooperation is supported by the NWO scientific program (The Netherlands).

8.4. Individual Involvement

F. Bouquet: Vice-head of LIFC laboratory, TPC Member of International Conference in Software Testing (ICST’08, ICST’09).

[9] Fonds Unique Interministériel & Fonds de Compétitivité des Entreprises (FCE)
[10] http://www.cs.waikato.ac.nz/Research/fm/index.html

V. Cortier: coordinator of the ANR SESUR AVOTÉ (started in January 2008); local coordinator of the ARA SSIA FormaCrypt (started in January 2006); French coordinator of the PHC Alliance project on refinement of security systems; French coordinator of the French-Tunisian project on e-voting; PC member of CSF 2008 (21st Computer Security Foundations Symposium), FCS-ARSPA-WITS 2008 (Joint Workshop on Foundations of Computer Security, Automated Reasoning for Security Protocol Analysis and Issues in the Theory of Security), SecCo 2008 (6th International Workshop on Security Issues in Concurrency); co-organizer and PC member of VETO 2008 (workshop sur la Sécurité Informatique et le Vote ElecTrOnique) and of the French-Japanese JST-CNRS Security workshop; member of the CSE 27 of the INPL and of the ENS Cachan, member of the recruitment committee 2008 of junior researchers at the Centre INRIA Grand Est, member of the Evaluation Committee of the INRIA since September 2008.
F. Dadeau: PC member of the 7th International Conference on integrated Formal Methods (iFM'09), Dusseldorf, Germany.
A. Giorgetti: Editorial committee member of *Techniques et Science Informatique (TSI)*. Member of the "CSE 27e section" of the University of Franche-Comté.
O. Kouchnarenko: director of the research team *Techniques Formelles et à Contraintes (TFCVESONTIO)* of the *Laboratoire d'informatique de Franche Comté (LIFC)*; PC member of "*International Workshop on Abstractions for Petri Nets and Other Models of Concurrency*", APNOC'09. Member of the "CSE 27" of the University of Franche-Comté, director of the "Licence Informatique 2008-2012" of the LMD2 in the University of Franche-Comté.
C. Ringeissen: PC member of IJCAR 2008 (the 4th International Joint Conference on Automated Reasoning).
M. Rusinowitch: member of the IFIP Working Group 1.6 (Rewriting); PC member of IJCAR 2008 (the 4th International Joint Conference on Automated Reasoning), RTA 2008 (International Conference on Rewriting Techniques and Applications), SecReT 2008 (3rd International Workshop on Security and Rewriting Techniques), the 2008 Workshop on Collaboration and Security (COLSEC'08), SAR - SSI 2008 (3ème Conférence sur la Sécurité des Architectures Réseaux et des Systèmes d'Information). Member of the CSE 27 of Nancy 2 University and Institut National Polytechnique de Lorraine.
Co-organizer of Dagstuhl Seminar: Beyond the Finite: *New Challenges in Verification and Semistructured Data* (20.04.08 - 25.04.08, Seminar 08171).
L. Vigneron: Member of the FTP steering committee; Secretary of the IFIP Working Group 1.6 (Rewriting); PC member of CRiSIS'2008; Webmaster of the site *Rewriting Home Page*, of the RTA conference site, and of the web page for the IFIP Working Group 1.6.
We are involved in several lectures of the "Master Informatique" of the universities of Nancy. L. Vigneron is in charge of the lecture on *Algorithmic verification*. V. Cortier is in charge of the lecture on *Theory of the security*, S. Ranise and C. Ringeissen are in charge of the lecture on *Decision procedures and program verification*.

8.5. Visits of Foreign Researchers

Bogdan Warinschi (University of Bristol) has visited LORIA to work on refinement of security systems (February 22-26th) and on combination techniques for soundness results of symbolic model (July 21-23rd).
Paliath Narendran (SUNY Albany) has visited LORIA from May 28 to June 5 to work on protocol verification.
Adel Bouhoula (SupCom Tunis) has visited LORIA from October 23 to October 28 to work on computer security.

8.6. Visits of Team Members

Véronique Cortier has visited Bogdan Warinschi (University of Bristol) to work on refinement of security systems (April 20-23rd) and on combination techniques for soundness results of symbolic models (October 15-18th).

9. Dissemination

9.1. Ph. D. Theses

Daniele Zucchelli has defended his Ph. D. thesis in co-tutelle with the University Henri Poincaré - Nancy 1 and the University of Milan, entitled "*Combination Methods for Verification Problems*", supervised by S. Ghilardi and M. Rusinowitch, on January 22, 2008.

9.2. Committees

O. Kouchnarenko is an expert for the committee CIFRE of the *Agence Nationale de Recherche Technologique*, (ANRT).
M. Rusinowitch is examiner for the theses of Mohsen Rouached (Nancy), Dong Cheng (Nancy), Barbara Fila-Kordy (Orléans) and has been expert for 2008 ANR Program.

9.3. Seminars, Workshops, and Conferences

We were invited to give the following talks.
V. CORTIER, Invited Talk on Verification techniques for cryptographic protocols at RTA 2008 (International Conference on Rewriting Techniques and Applications), Hagenberg, Austria, July 15th, 2008; Invited Talk on Building secure protocols at TFIT 2008 (Fourth Taiwanese-French Conference on Information Technology), Taipei, Taiwan, March 4th, 2008; Invited Talk on the use of formal models for proving cryptographic security notions at the LSV Seminar, Cachan, France, May 6th, 2008; Invited Talk on Verifying security protocols at the workshop in Honour of Hubert Comon-Lundh, Cachan, France, November 18th, 2008.
A. GIORGETTI, *Automates et annotations JML pour la vérification de programmes Java*, Invited Talk at the STIC-Tunisie project workshop, Sup'Com Tunis, November 1st, joint work with J. Groslambert, J. Julliand and O. Kouchnarenko.
P. HÉAM, Invited Talk on Regular Approximations at Institut Gaspard Monge, Université Marne-la-Vallée and at Laboratoire d'Informatique et d'Algorithmique : Fondements et Applications, Université Paris 7.
C. RINGEISSEN, Invited Talk on Combination of Proof-Producing Decision Procedures, Seminar Team CPR (Systèmes sûrs : Conception et Programmation Raisonnées), Cédric, CNAM, Paris, January 21, 2008.
M. RUSINOWITCH, Invited Talk on Constraint-based Verification of Cryptographic Protocols, at NIAS-Lorentz workshop Logic and information security, September 25 2008. Invited Talk on protocol and service verification at the STIC-Tunisie project workshop, Sup'Com Tunis, November 1st.
M. TURUANI, Invited Talk "CL-AtSe, théorie et applications", Seminar Crypto-sécurité, IRIT, Toulouse, November 27, 2008.
L. VIGNERON, Invited Talk "AVISPA, un outil d'analyse de protocoles cryptographiques", Seminar Crypto-sécurité, IRIT, Toulouse, November 27, 2008.

10. Bibliography

Major publications by the team in recent years

[1] A. ARMANDO, S. RANISE, M. RUSINOWITCH. *A Rewriting Approach to Satisfiability Procedures*, in "Journal of Information and Computation — Special Issue on Rewriting Techniques and Applications (RTA'01)", vol. 183, n^{o} 2, June 2003, p. 140–164.

[2] F. BOUQUET, B. LEGEARD, F. PEUREUX. *CLPS-B: A Constraint Solver to Animate a B Specification*, in "International Journal of Software Tools for Technology Transfer, STTT", vol. 6, n^{o} 2, August 2004, p. 143–157.

[3] Y. CHEVALIER, L. VIGNERON. *Strategy for Verifying Security Protocols with Unbounded Message Size*, in "Journal of Automated Software Engineering", vol. 11, n^{o} 2, April 2004, p. 141–166.

[4] H. COMON-LUNDH, V. CORTIER. *Security properties: two agents are sufficient*, in "Science of Computer Programming", vol. 50, n^{o} 1-3, March 2004, p. 51–71, http://www.loria.fr/~cortier/Papiers/ComonCortierSCP03.ps.

[5] F. JACQUEMARD, M. RUSINOWITCH, L. VIGNERON. *Compiling and Verifying Security Protocols*, in "Logic for Programming and Automated Reasoning (LPAR'00), Reunion Island, France", A. VORONKOV, M. PARIGOT (editors), Lecture Notes in Computer Science, vol. 1955, Springer, 2000, p. 131–160.

[6] B. LEGEARD, F. PEUREUX. *B-Testing-Tools : génération de tests aux limites à partir de spécifications B*, in "TSI, Techniques et Sciences Informatiques, Hermès-Lavoisier", vol. 21, n^{o} 9, 2002, p. 1189–1218.

[7] B. LEGEARD, F. PEUREUX, M. UTTING. *Automated Boundary Testing from Z and B*, in "Formal Methods Europe (FME 2002)", L.-H. ERIKSSON, P. LINDSAY (editors), Lecture Notes in Computer Science, vol. 2391, Springer, 2002, p. 21–40.

[8] M. RUSINOWITCH, M. TURUANI. *Protocol Insecurity with Finite Number of Sessions and Composed Keys is NP-complete*, in "Theoretical Computer Science", vol. 299, April 2003, p. 451–475, http://www.loria.fr/~rusi/pub/tcsprotocol.ps.gz.

[9] C. TINELLI, C. RINGEISSEN. *Unions of Non-Disjoint Theories and Combinations of Satisfiability Procedures*, in "Theoretical Computer Science", vol. 290, n^{o} 1, 2003, p. 291–353.

Year Publications

Doctoral Dissertations and Habilitation Theses

[10] D. ZUCCHELLI. *Combinaison de Méthodes de Vérification*, Ph. D. Thesis, Université Henri Poincaré - Nancy I, 01 2008, http://tel.archives-ouvertes.fr/tel-00329849/en/.

Articles in International Peer-Reviewed Journal

[11] Y. CHEVALIER, R. KUESTERS, M. RUSINOWITCH, M. TURUANI. *Complexity results for security protocols with Diffie-Hellman exponentiation and commuting public key encryption*, in "ACM Transactions on Computational Logic (TOCL)", vol. 9, 2008, Article 24, http://hal.inria.fr/inria-00329740/en/.

[12] Y. CHEVALIER, M. RUSINOWITCH. *Hierarchical combination of intruder theories*, in "Information and Computation", vol. 206, 2008, p. 352-377, http://hal.inria.fr/inria-00329715/en/.

[13] V. CORTIER, S. DELAUNE. *Safely Composing Security Protocols*, in "Formal Methods in System Design", 2008, http://hal.inria.fr/inria-00332354/en/.

[14] A. GIORGETTI, J. GROSLAMBERT, J. JULLIAND, O. KOUCHNARENKO. *Verification of class liveness properties with Java modeling language*, in "IET (Institution of Engineering and Technology) Software", vol. 2, n^{o} 6, December 2008, p. 500-514, http://hal.inria.fr/inria-00332862/en/.

[15] P.-C. HEAM, Y. BOICHUT. *A Theorerical Limit for Safety Verification Techniques with Regular Fix-point Computations*, in "Information Processing Letters", vol. 108, 2008, p. 1-2, http://hal.inria.fr/inria-00328487/en/.

[16] P.-C. HEAM, G. CÉCÉ, Y. MAINIER. *Efficiency of Automata in Semicommutations Verification Techniques*, in "Theoretical Informatics and Applications", vol. 42, 2008, p. 197-215, http://hal.inria.fr/inria-00328514/en/.

[17] P.-C. HEAM. *A Note on Partially Ordered Tree Automata*, in "Information Processing Letters", vol. 108, 2008, p. 242-246, http://hal.inria.fr/inria-00328495/en/.

[18] F. JACQUEMARD, M. RUSINOWITCH, L. VIGNERON. *Tree Automata with Equality Constraints Modulo Equational Theories*, in "Journal of Logic and Algebraic Programming", vol. 75, n° 2, April 2008, p. 182-208, http://hal.inria.fr/inria-00329693/en/.

[19] E. MONFROY, O. PERRIN, C. RINGEISSEN. *Modeling Web services Composition with Constraints*, in "Selected Papers of the Third Colombian Conference on Computer Science, Special Issue of "Revista Avances en Sistemas e Informática"", vol. 5, n° 1, 2008.

Articles in National Peer-Reviewed Journal

[20] G. CÉCÉ, P.-C. HÉAM, Y. MAINIER. *Clôtures transitives de semi-commutations et model-checking régulier*, in "TSI, Technique et Science Informatiques", vol. 27, n° 1-2, 2008, p. 7–28.

International Peer-Reviewed Conference/Proceedings

[21] T. ABBES, A. BOUHOULA, M. RUSINOWITCH. *An inference system for detecting firewall filtering rules anomalies*, in "23rd Annual ACM Symposium on Applied Computing - SAC'08, Fortaleza, Ceara, Brazil", R. L. WAINWRIGHT, H. HADDAD (editors), ACM, 2008, p. 2122-2128, http://hal.inria.fr/inria-00329730/en/.

[22] H. ABDELNUR, T. AVANESOV, M. RUSINOWITCH, R. STATE. *Abusing SIP Authentication*, in "Information Assurance and Security (ISIAS), Naples, Italy", IEEE, 2008, p. 237-242, http://hal.inria.fr/inria-00326077/en/.

[23] P. BALBIANI, Y. CHEVALIER, M. EL HOURI. *A Logical Approach to Dynamic Role-Based Access Control*, in "Artificial Intelligence: Methodology, Systems, and Applications, 13th International Conference, AIMSA 2008, Varna, Bulgaria", Lecture Notes in Computer Science, vol. 5253, Springer, September 2008, p. 194-208.

[24] Y. BOICHUT, R. COURBIS, P.-C. HEAM, O. KOUCHNARENKO. *Finer is better: Abstraction Refinement for Rewriting Approximations*, in "19th International Conference on Rewriting Techniques and Applications - RTA'2008, Hagenberg, Austria", A. VORONKOV (editor), Lecture Notes in Computer Science, vol. 5117, Springer, 2008, p. 48-62, http://hal.inria.fr/inria-00327583/en/.

[25] Y. BOICHUT, R. COURBIS, P.-C. HEAM, O. KOUCHNARENKO. *Handling Left-Quadratic Rules when Completing Tree Automata*, in "2nd Workshop on Reachability Problems - RP'08, Electronic Notes in Theoretical Computer Science, Liverpool, UK", V. HALAVA, I. POTAPOV (editors), Elsevier Science Publishers, 2008, http://hal.inria.fr/inria-00329900/en/.

[26] F. BOUQUET, C. GRANDPIERRE, B. LEGEARD, F. PEUREUX. *A test generation solution to automate software testing*, in "AST'08, 3rd Int. workshop on Automation of Software Test, Leipzig, Germany", ACM Press, May 2008, p. 45-48, http://doi.acm.org/10.1145/1370042.1370052.

[27] Y. CHEVALIER, M. A. MEKKI, M. RUSINOWITCH. *Automatic Composition of Services with Security Policies*, in "Web Service Composition and Adaptation Workshop (held in conjunction with SCC/SERVICES-2008), Honolulu, USA", IEEE, 2008, p. 529-537, http://hal.inria.fr/inria-00330338/en/.

[28] H. COMON-LUNDH, V. CORTIER. *Computational soundness of observational equivalence*, in "15th ACM Conference on Computer and Communications Security - CCS'08, Alexandria, USA", ACM, 2008, http://hal.inria.fr/inria-00323195/en/.

[29] F. DADEAU, A. DE KERMADEC, R. TISSOT. *Combining Scenario- and Model-Based Testing to ensure POSIX Compliance*, in "First International Conference on Abstract State Machines, B and Z - ABZ'08, London, UK", E. BÖRGER, M. BUTLER, J. P. BOWEN, P. BOCA (editors), Lecture Notes in Computer Science, vol. 5238, Springer, 2008, p. 153-166, http://hal.inria.fr/inria-00329965/en/.

[30] F. DADEAU, J. LAMBOLEY, T. MOUTET, M.-L. POTET. *A Verifiable Conformance Relationship between Smart Card Applets and B security Models*, in "First International Conference on Abstract State Machines, B and Z - ABZ'08, London, UK", E. BÖRGER, M. BUTLER, J. P. BOWEN, P. BOCA (editors), Lecture Notes in Computer Science, vol. 5238, Springer, 2008, p. 237-250, http://hal.inria.fr/inria-00329966/en/.

[31] F. DADEAU, R. TISSOT, M.-L. POTET. *A B Formal Framework for Security Developments in the Domain of Smart Card Applications*, in "23rd International Information Security Conference - SEC'08 IFIP, Milano, Italy", vol. 278, Springer, 2008, p. 141-155, http://hal.inria.fr/inria-00329973/en/.

[32] S. GHILARDI, E. NICOLINI, S. RANISE, D. ZUCCHELLI. *Towards SMT Model Checking of Array-based Systems*, in "Proceedings of the 4th International Joint Conference on Automated Reasoning, IJCAR, Sydney (Australia)", A. ARMANDO, P. BAUMGARTNER, G. DOWEK (editors), Lecture Notes in Computer Science, vol. 5195, Springer, 2008, p. 67-82.

[33] E. GIOAN, S. BURCKEL. *In Situ Design of Register Operations*, in "ISVLSI'08: IEEE Computer Society Annual Symposium on Very-Large-Scale Integration, Montpellier, France", vol. Trends in VLSI Technology and Design, IEEE Computer Society, 04 2008, 4, http://hal-lirmm.ccsd.cnrs.fr/lirmm-00287659/en/.

[34] N. GUERMOUCHE, O. PERRIN, C. RINGEISSEN. *A Mediator Based Approach For Services Composition*, in "International Conference on Software Engineering Research, Management and Applications (SERA'08), Prague, Czech Republic", 2008, http://hal.inria.fr/inria-00275221/en/.

[35] N. GUERMOUCHE, O. PERRIN, C. RINGEISSEN. *Timed Specification For Web Services Compatibility Analysis*, in "Proc. of the 3rd International Workshop on Automated Specification and Verification of Web Systems, Venice, Italy, Dec. 2007", Electronic Notes in Theoretical Computer Science, vol. 200/3, Elsevier, 2008, p. 155-170.

[36] P.-C. HEAM, O. KOUCHNARENKO, J. VOINOT. *Component Simulation-based Substitutivity Managing QoS Aspects*, in "5th International Workshop on Formal Aspects On Component Softwre - FACS'08, Malaga, Spain", C. CANAL, C. PASAREANU (editors), To appear in ENTCS, 2008, http://hal.inria.fr/inria-00329909/en/.

[37] A. IMINE. *Decentralized Concurrency Control for Real-time Collaborative Editors*, in "8th international conference on New technologies in distributed systems, NOTERE, Lyon, France", D. BENSLIMANE, A. OUKSEL (editors), ACM, June 2008, p. 313-321, http://hal.inria.fr/inria-00338859/en/.

[38] A. IMINE. *Flexible Concurrency Control for Real-time Collaborative Editors*, in "28th International Conference on Distributed Computing Systems Workshops, ICDCSW, Beijing, China", IEEE Computer Society, June 2008, p. 423-428, http://hal.inria.fr/inria-00338871/en/.

[39] F. JACQUEMARD, M. RUSINOWITCH. *Closure of Hedge-Automata Languages by Hedge Rewriting*, in "19th International Conference on Rewriting Techniques and Applications - RTA 2008, Hagenberg, Austria", A. VORONKOV (editor), Lecture Notes in Computer Science, vol. 5117, Springer, 2008, p. 157-171, http://hal.inria.fr/inria-00329803/en/.

[40] F. KLAY, L. VIGNERON. *Automatic Methods for Analyzing Non-Repudiation Protocols with an Active Intruder*, in "5th International Workshop on Formal Aspects in Security and Trust (FAST), Malaga, Spain", P. DEGANO, J. GUTTMAN, F. MARTINELLI (editors), 15 pages. To appear as LNCS, 2008, p. 165-180, http://hal.inria.fr/inria-00329808/en/.

[41] E. MONFROY, O. PERRIN, C. RINGEISSEN. *Dynamic Web Services Provisioning with Constraints*, in "International Conference on Cooperative Information Systems, OTM 2008, Part I, Monterrey, Mexico", R. MEERSMAN, Z. TARI (editors), Lecture Notes in Computer Science, This work is partly funded by the INRIA-CONICYT project "CoreWeb" and the INRIA associate team "VanaWeb"., vol. 5331, Springer, 2008, p. 26-43, http://hal.inria.fr/inria-00329815/en/.

[42] V. PRETRE, F. BOUQUET, C. LANG. *Automating UML Models Merge for Web Services Testing*, in "ii-WAS2008, 10th int. Conf. on Information Integration and Web-based Applications and Services", To appear, 2008.

Workshops without Proceedings

[43] S. ANANTHARAMAN, H. LIN, C. LYNCH, P. NARENDRAN, M. RUSINOWITCH. *Active Intruders with Caps*, in "FCS-ARSPA-WITS'08, Pittsburgh, USA", 2008, http://hal.inria.fr/inria-00330532/en/.

[44] N. CHRIDI, M. TURUANI, M. RUSINOWITCH. *Towards a Constrained-based Verification of Parameterized Cryptographic Protocols*, in "LOPSTR 2008: Logic-based Program Synthesis and Transformation, Valencia, Spain", M. HANUS (editor), 2008, http://hal.inria.fr/inria-00332484/en/.

[45] H. COMON-LUNDH, V. CORTIER. *Computational soundness of observational equivalence*, in "4th Workshop on Formal and Computational Cryptography - FCC 2008, Pittsburgh, USA", 2008, http://hal.inria.fr/inria-00323199/en/.

Research Reports

[46] Y. BOICHUT, P.-C. HEAM. *A Theoretical Limit for Safety Verification Techniques with Regular Fix-point Computations*, Research Report, n^{o} RR-6411, INRIA, January 2008, http://hal.inria.fr/inria-00204579/en/.

[47] H. BOUCHENEB, A. IMINE. *Experiments in Model-Checking Optimistic Replication Algorithms*, RR-6510, Research Report, INRIA, 2008, http://hal.inria.fr/inria-00274423/en/.

[48] H. COMON-LUNDH, V. CORTIER. *Computational soundness of observational equivalence*, RR-6508, Research Report, INRIA, 2008, http://hal.inria.fr/inria-00274158/en/.

[49] J.-F. COUCHOT, A. GIORGETTI, N. STOULS. *Graph-based Reduction of Program Verification Conditions*, RR-6702, Research Report, INRIA, October 2008, http://hal.inria.fr/inria-00339847/en/.

[50] F. DADEAU, P.-C. HÉAM, J. LEVREY. *A Combination of Model-Based Testing and Random Testing Approaches using Automata*, 21 pages, Research Report, n^{o} RR2008-10, LIFC - Laboratoire d'Informatique de l'Université de Franche Comté, October 2008.

[51] E. NICOLINI, C. RINGEISSEN, M. RUSINOWITCH. *Satisfiability Procedures for Combination of Theories Sharing Integer Offsets*, RR-6697, Research Report, INRIA, 2008, http://hal.inria.fr/inria-00331735/en/.

[52] D.-K. TRAN, C. RINGEISSEN, S. RANISE, H. KIRCHNER. *Combination of Convex Theories: Modularity, Deduction Completeness, and Explanation*, RR-6688, Research Report, INRIA, 2008, http://hal.inria.fr/inria-00331479/en/.

Other Publications

[53] A. BERREGBA. *Du Contrôle d'Accès Dynamique pour les Editeurs Collaboratifs*, Mémoire de Master Recherche, LORIA, Université Henri Poincaré, Nancy, 2008.

References in notes

[54] F. BAADER, K. U. SCHULZ. *Unification in the Union of Disjoint Equational Theories: Combining Decision Procedures*, in "Journal of Symbolic Computation", vol. 21, n^{o} 2, February 1996, p. 211–243.

[55] F. BELLEGARDE, C. DARLOT, J. JULLIAND, O. KOUCHNARENKO. *Reformulation: a Way to Combine Dynamic Properties and Refinement*, in "International Symposium Formal Methods Europe (FME 2001)", LNCS, vol. 2021, Springer-Verlag, 2001.

[56] E. BERNARD, B. LEGEARD, X. LUCK, F. PEUREUX. *Generation of Test Sequences from Formal Specifications: GSM 11-11 Standard Case-Study*, in "International Journal on Software Practice and Experience", vol. 34, n^{o} 10, 2004, p. 915–948.

[57] Y. BOICHUT, P.-C. HÉAM, O. KOUCHNARENKO. *Vérifier automatiquement les protocoles de sécurité*, in "Techniques de l'ingénieur", October 2007, p. RE95-1–RE95-8.

[58] A. BOUAJJANI, A. MUSCHOLL, T. TOUILI. *Permutation Rewriting and Algorithmic Verification*, in "LICS", 2001.

[59] F. BOUQUET, B. LEGEARD. *Reification of Executable Test Scripts in Formal Specification-Based Test Generation: The Java Card Transaction Mechanism Case Study*, in "Formal Methods, FME 2003", vol. 2805, Springer-Verlag, September 2003, p. 778–795.

[60] F. BOUQUET, B. LEGEARD, F. PEUREUX. *CLPS-B - A Constraint Solver for B*, in "International Conference on Tools and Algorithms for Construction and Analysis of Systems, TACAS2002, Grenoble, France", Lecture Notes in Computer Science, vol. 2280, Springer, April 2002, p. 188–204.

[61] S. BURCKEL. *Procédé et Système de Transmission de Données*, INPI 05 12491 - FR 2894743 - PCT FR2006/002692, december 2005, Patent.

[62] S. BURCKEL, E. GIOAN. *Procédé d'Optimisation des Ressources Mémoires*, INPI FR0705152, july 2007, Patent.

[63] Y. CHEVALIER, D. LUGIEZ, M. RUSINOWITCH. *Towards an Automatic Analysis of Web Service Security*, in "Proceedings of the 6th International Symposium on Frontiers of Combining Systems (FroCoS'07), Liverpool, UK", F. WOLTER (editor), Lecture Notes in Artificial Intelligence, vol. 4720, Springer, September 2007, p. 133-147.

[64] V. CORTIER, S. DELAUNE, P. LAFOURCADE. *A Survey of Algebraic Properties Used in Cryptographic Protocols*, in "Journal of Computer Security", vol. 14, n^{o} 1, 2006, p. 1–43, http://www.loria.fr/~cortier/Papiers/survey.ps.

[65] S. EVEN, O. GOLDREICH. *On the Security of Multi-Party Ping-Pong Protocols*, in "IEEE Symposium on Foundations of Computer Science", 1983, p. 34-39, http://citeseer.ist.psu.edu/46982.html.

[66] G. FEUILLADE, T. GENET, V. V. T. TONG. *Reachability Analysis over Term Rewriting Systems*, in "J. Autom. Reasoning", vol. 33, n^{o} 3-4, 2004, p. 341-383.

[67] P.-C. HEAM, O. KOUCHNARENKO, J. VOINOT. *How to Handle QoS Aspects in Web Services Substitutivity Verification*, in "International Workshop on Information Systems & Web Services, as part of the 16th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE 2007), Paris, France", June 2007.

[68] G. T. LEAVENS, A. L. BAKER, C. RUBY. *JML: a Java Modeling Language*, in "Formal Underpinnings of Java Workshop (at OOPSLA '98)", October 1998.

[69] M. TURUANI. *The CL-AtSe Protocol Analyser*, in "Term Rewriting and Applications - Proc. of RTA, Seattle, WA, USA", Lecture Notes in Computer Science, vol. 4098, 2006, p. 277–286.