RESEARCH CENTER

FIELD
Algorithmics, Programming, Software and Architecture

Activity Report 2012

Section New Results

Edition: 2013-04-24

ARIC Team

6. New Results

6.1. Applications

Florent de Dinechin contributed high-performance signal processing on an FPGA to a prototype of high-throughput receiver for optical fiber transmission developed by Alcatel [33]. He also wrote a book chapter exposing the potential of FPGA-specific arithmetic for high-performance computing [49].

6.2. Hardware and FPGA Arithmetic

6.2.1. Mixed-precision fused multiply-and-add

With B. de Dinechin, from Kalray, N. Brunie and F. de Dinechin proposed to extend the classical fused-multiply-and-add operator with a larger addend and result. This enables higher-precision computation of sums of products at a cost that remains close to that of the classical FMA [29].

6.2.2. Multiplication by rational constants versus division by a constant

Motivated by the division by 3 or by 9 appearing in some stencil kernels, F. de Dinechin investigated how the periodicity of the binary representation of a rational constant could be exploited to design an architecture multiplying by this constant [18]. With L. S. Didier, this approach was then compared to a specialisation of divider architectures to the division by small integer constants, which is shown to match well the fine structure of FPGAs [32].

6.2.3. Floating-point exponentiation on FPGA

F. de Dinechin, with P. Echeverria and M. Lopez-Vallejo (U. Madrid) and B. Pasca (Altera), implemented the first floating-point unit for the pow and powr functions of the IEEE-754-2008 standard [50]. These functions compute x^y, and differ only in the specification of special cases. The implementation, parameterized in exponent and significand size, combines suitably modified exponential and logarithm units.

6.2.4. Arithmetic around the bit heap

F. de Dinechin, M. Istoan, G. Sergent, K. Illyes, B. Popa, and N. Brunie extended FloPoCo with a versatile framework for manipulating sums of weighted bits [51], [44]. This is a relevant way of implementing polynomials, filters and other coarse arithmetic cores.

6.2.5. Improving computing architectures

To improve High-Level Synthesis (HLS) for FPGAs, B. Pasca (former PhD student in AriC), with Ch. Alias (Inria Compsys) and A. Plesco (Zettice) developed tiling and scheduling algorithms that exploit the deeply pipelined operator at the core of a computing kernel [14].

With S. Collange and G. Diamos, N. Brunie proposed improvements in the architecture of general-purpose graphical processing units [28].

N. Brunie and F. de Dinechin, with Kalray's B. de Dinechin, are investigating embedding a reconfigurable core in the Kalray MPPA architecture. For this purpose, N. Brunie developed an environment for the design exploration of such an accelerator. This environment produces the hardware on one side, and its programming tools on the other side [43].

6.3. Elementary Functions

6.3.1. (M,p,k)-friendly points: a table-based method for trigonometric function evaluation

N. Brisebarre, M. Ercegovac (U. California at Los Angeles) and J.-M. Muller [25] present a new way of approximating the sine and cosine functions by a few table look-ups and additions. It consists in first reducing the input range to a very small interval by using rotations with "(M, p, k) friendly angles", proposed in this work, and then by using a bipartite table method in a small interval. An implementation of the method for 24-bit case is described and compared with CORDIC. Roughly, the proposed scheme offers a speedup of 2 compared with an unfolded double-rotation radix-2 CORDIC.

6.3.2. On Ziv's rounding test

With Ch. Lauter (LIP6), F. de Dinechin, J.-M. Muller and S. Torres proved and generalized a code sequence due to Ziv, which is used to round correctly a real value approximated (with a known error bound) as the unevaluated sum of two floating-point numbers [52].

6.4. Arithmetic Algorithms

6.4.1. Binary floating-point operators for VLIW integer processors

C.-P. Jeannerod and J. Jourdan-Lu [35] proposed software implementations of sinf, cosf and sincosf over [-pi/4, pi/4] that have proven 1-ulp accuracy and whose respective latencies on STMicroelectronics' ST231 VLIW integer processor are 19, 18 and 19 cycles. To get such performances they introduced a novel algorithm for simultaneous sine and cosine that combines univariate and bivariate polynomial evaluation schemes.

In the same context, C.-P. Jeannerod, J. Jourdan-Lu and C. Monat (STMicroelectronics Compilation Expertise Center, Grenoble) [36] studied the implementation of *custom* (i.e., specialized, fused, or simultaneous) operators, and provided qualitative evidence of the benefits of supporting such operators in addition to the five basic ones: this allows to be up to 4.2x faster on individual calls, and up to 1.59x faster on DSP kernels and benchmarks.

6.4.2. Error bounds for complex floating-point division with an FMA

Assuming that a fused multiply-add (FMA) instruction is available, C.-P. Jeannerod, N. Louvet and J.-M. Muller [37] obtained sharp error bounds for various alternatives to Kahan's 2 by 2 determinant algorithm. Combining such alternatives with Kahan's original scheme leads to componentwise-accurate algorithms for complex floating-point division, and for these algorithms sharp or reasonably sharp error bounds were also obtained.

6.4.3. Computation of correctly-rounded sums

P. Kornerup (U. of Southern Denmark), V. Lefèvre and J.-M. Muller [19] have shown that among the set of the algorithms with no comparisons performing only floating-point additions/subtractions, the 2Sum algorithm introduced by Knuth is minimal, both in terms of number of operations and depth of the dependency graph. They also prove that under reasonable conditions, an algorithm performing only round-to-nearest additions/subtractions cannot compute the round-to-nearest sum of at least three floating-point numbers. They also present new results about the computation of the correctly-rounded sum of three floating-point numbers.

6.4.4. Comparison between binary64 and decimal64 floating-point numbers

N. Brisebarre, C. Lauter (U. Paris 6), M. Mezzarobba and J.-M. Muller [27] introduce an algorithm that allows one to quickly compare a binary64 floating-point (FP) number and a decimal64 FP number, assuming the "binary encoding" of the decimal formats specified by the IEEE 754-2008 standard for FP arithmetic is used. It is a two-step algorithm: a first pass, based on the exponents only, makes it possible to quickly eliminate most cases, then when the first pass does not suffice, a more accurate second pass is required. They provide an implementation of several variants of their algorithm, and compare them.

6.5. Computer Algebra

6.5.1. Faster multivariate interpolation with multiplicities

M. Chowdhury (U. Western Ontario), C.-P. Jeannerod, V. Neiger (ENS de Lyon), É. Schost (U. Western Ontario) and G. Villard proposed fast randomized algorithms for interpolating multivariate polynomials with multiplicities. In the special bivariate case, this allows to accelerate the interpolation step of Guruswami and Sudan's list-decoding by a factor (list size)/(multiplicity).

6.5.2. On the complexity of solving quadratic boolean systems

M. Bardet (U. Rouen), J.-Ch. Faugère (PolSys), B. Salvy, and P.-J. Spaenlehauer (PolSys) [16] dealt with the fundamental problem in computer science of finding all the common zeroes of polynomials systems of quadratic polynomials over the field with 2 elements. The cryptanalysis of several modern ciphers reduces to this problem. Up to now, the best complexity bound was reached by an exhaustive search. They gave an algorithm that reduces the problem to a combination of exhaustive search and sparse linear algebra. This algorithm has several variants depending on the method used for the linear algebra step. Under precise algebraic assumptions, their complexity breaks the 2^n barrier. Experiments on random systems show that the algebraic assumptions are satisfied with probability very close to 1.

6.5.3. Power series solutions of singular (q)-differential equations

A. Bostan (Algorithms), M. F. I. Chowdhury (U. Western Ontario), R. Lebreton (Lix), B. Salvy, and É. Schost (U. Western Ontario) provided in [23] algorithms computing power series solutions of a large class of differential or q-differential equations or systems. Their number of arithmetic operations grows linearly with the precision, up to logarithmic terms.

6.5.4. Fast computation of common left multiples of linear ordinary differential operators

A. Bostan (Algorithms), F. Chyzak (Algorithms), Ziming Li (Chinese Academy of Sciences), and B. Salvy studied in [24] tight bounds and fast algorithms for LCLMs of several linear differential operators with polynomial coefficients. They analyzed the arithmetic complexity of existing algorithms for LCLMs, as well as the size of their outputs. They proposed a new algorithm that recasts the LCLM computation in a linear algebra problem on a polynomial matrix. This algorithm yields sharp bounds on the coefficient degrees of the LCLM, improving by one order of magnitude the best bounds obtained using previous algorithms. The complexity of the new algorithm is almost optimal, in the sense that it nearly matches the arithmetic size of the output.

6.5.5. Space complexity of fast D-finite function evaluation

M. Mezzarobba [41] showed that D-finite functions, i.e., solutions of linear differential equations with polynomial coefficients, can be evaluated in quasi-linear time and linear space with respect to the precision. In comparison, existing fast algorithms due to Chudnovsky and Chudnovsky and to van der Hoeven achieved the same time complexity with an overhead of a logarithmic factor in terms of memory usage.

6.5.6. Multiple precision evaluation of the Airy function with reduced cancellation

The series expansion at the origin of the Airy function $\mathrm{Ai}(x)$ is alternating and hence problematic to evaluate for $x > 0$ due to cancellation. Based on a method recently proposed by Gawronski, Müller, and Reinhard, Sylvain Chevillard and Marc Mezzarobba [31] exhibit two functions F and G, both with nonnegative Taylor expansions at the origin, such that $\mathrm{Ai}(x) = G(x)/F(x)$. The sums are now well-conditioned, but the Taylor coefficients of G turn out to obey an ill-conditioned three-term recurrence. They use the classical Miller algorithm to overcome this issue. They bound all errors and their implementation allows an arbitrary and certified accuracy, that can be used, e.g., for providing correct rounding in arbitrary precision.

6.5.7. Algorithms for combinatorial structures: well-founded systems and Newton iterations

C. Pivoteau (U. Marne-la-Vallée), B. Salvy, and M. Soria (UPMC) [21] considered systems of recursively defined combinatorial structures. They gave algorithms checking that these systems are well founded, computing generating series and providing numerical values. Their framework is an articulation of the constructible classes of Flajolet and Sedgewick with Joyal's species theory. They extend the implicit species theorem to structures of size zero. A quadratic iterative Newton method was shown to solve well-founded systems combinatorially. From there, truncations of the corresponding generating series were obtained in quasi-optimal complexity. This iteration transfers to a numerical scheme that converges unconditionally to the values of the generating series inside their disk of convergence. These results provide important subroutines in random generation. Finally, the approach was extended to combinatorial differential systems.

6.6. Euclidean Lattice Reduction and Applications

6.6.1. Lattice algorithms and hardness proofs

X.-W. Chang (McGill), D. Stehlé and G. Villard [17] proposed the first fully rigorous perturbation analysis of the R-factor of LLL-reduced matrices under column-wise perturbations. This study is very useful to devise LLL-type algorithms relying on floating-point approximations.

L. Luzzi (ENSEA), C. Ling (Imperial College) and D. Stehlé improved [20] the analyses of efficient Bounded Distance Decoding algorithms for lattices, and investigated the consequences for lattice-coded multiple-input multiple-output (MIMO) systems.

A. Langlois and D. Stehlé [54] introduced the Module-SIS and Module-LWE average-case lattice problems and reduced worst-case lattice problems to them. This provides a progressive transformation from the non-structured average-case lattices problems SIS and LWE, to the quite restricted but efficient average-case lattices problems Ring-SIS and Ring-LWE.

6.6.2. Cryptography

S. Ling (Nanyang Technological University, Singapore) and D. Stehlé [55] described the first public-key traitor tracing encryption scheme with security relying on the hardness of standard worst-case problems on Euclidean lattices.

J.-C. Belfiore (Telecom Paritech), L. Luzzi (ENSEA), C. Ling (Imperial College) and D. Stehlé [53] proved that nested lattice codes can achieve semantic security and strong secrecy over the Gaussian wiretap channel.

S. Ling (Nanyang Technological University, Singapore), K. Nguyen (NTU), H. Wang (NTU) and D. Stehlé [40] generalized Stern's zero-knowledge proof of knowledge protocol to obtain a statistical zero-knowledge proof of knowledge for the Inhomogeneous Small Integer Solution ISIS problem (in the infinity norm). This scheme is the first one that comes with no norm loss in the knowledge extraction procedure, leading to cryptographic constructions with tighter security proofs.

N. Attrapadung (AIST, Japan), J. Herranz (UPC, Spain), F. Laguillaumie, B. Libert (UCL, Belgium), E. de Panafieu (ENS Cachan), C. Ràfols (UPC, Spain) [15] proposed the first attribute-based encryption (ABE) schemes allowing for truly expressive access structures and with constant ciphertext size.

G. Castagnos (IMB) and F. Laguillaumie [38] gave a generic approach to design homomorphic encryption schemes, which extends Gjosteen's framework. A specific scheme allows an arbitrary number of multiplications in the groups, as well as a pairing evaluation on the underlying plaintexts.

J. Herranz (UPC, Spain), F. Laguillaumie, B. Libert (UCL, Belgium) and C. Ràfols (URV, Catalonia) [34] proposed the first two attribute-based (for threshold predicates) signature schemes with constant size signatures. Their security is proven in the selective-predicate and adaptive-message setting, in the standard model, under chosen message attacks.

S. Canard (Orange Labs), G. Fuchsbauer (University of Bristol, UK), A. Gouget (Gemalto), F. Laguillaumie [30] defined a new cryptographic primitive called plaintext-checkable encryption, which extends public-key encryption by the following functionality: given a plaintext, a ciphertext and a public key, it is universally possible to check whether the ciphertext encrypts the plaintext under the key. They provide efficient generic random-oracle constructions based on any probabilistic or deterministic encryption scheme as well as a practical construction in the standard model.

6.7. Reliability and Accuracy

6.7.1. Standardization of interval arithmetic

We contributed to the creation in 2008 and N. Revol chairs the IEEE 1788 working group on the standardization of interval arithmetic http://grouper.ieee.org/groups/1788/. More than 140 persons from over 20 countries take part in the discussions, around 1500 messages were exchanged in 2012. We are currently voting on portions of the text of the standard and have good hope that the group will reach a final version of the standard within the allotted time. An extension has been granted for 2 more years, until December 2014.

The annual in-person meeting, chaired by N. Revol, took place at the end of the SCAN 2012 conference in Novosibirsk, Russia, the 28th of September. It was broadcasted via the Web and feedback was possible through e-mails. More than 20 persons attended the meeting.

V. Lefèvre participated in various discussions, either in the mailing-list or in small subgroups (he sent around 390 mail messages in 2012). He proposed a motion, which passed, on properties needed by number formats for operations between intervals and numbers (constructors, midpoint, etc.).

The latest discussions dealt with:

- flavors: even if there continues to be a give-and-take between proponents of a "small" standard involving just basic interval arithmetic and those who also want to also include the less common "modal arithmetic", this motion about "flavors" intends to allow inclusion of modal interval arithmetic consistently and simply, possibly at a later stage or revision of the standard;
- expressions: what is regarded as an expression by P1788, the relation with the programming languages, what this implies concerning the allowed optimizations, etc.;
- decorations: what are the properties of functions we want to track along a computation, how the empty interval is handled, etc.;
- reproducibility: across several runs of a translated (e.g., compiled) program or across platforms, representation-independent behavior, reproducibility for parallel programs, etc.

A personal view of the current status of the work of the IEEE P1788 group and of directions for future work has been presented in [46], [45].

6.7.2. Interval matrix multiplication

Several formulas exist for the product of two intervals using the midpoint-radius representation: they trade off accuracy for efficiency. The use of these formulas for the product of matrices with interval coefficients allows to use BLAS3 routines and to benefit from their performances in terms of execution time [48]. The accuracy of these methods are studied in [42]. As it can be difficult to ensure that a prescribed rounding mode is actually in use, formulas that are oblivious to the rounding mode are developed [22]. The implementations of these variants on multicores are compared in [47].

6.7.3. Rigorous polynomial approximation using Taylor models in Coq

One of the most common and practical ways of representing a real function on machines is by using a polynomial approximation. It is then important to properly handle the error introduced by such an approximation. N. Brisebarre, M. Joldes (Uppsala Univ., Sweden), E. Martin-Dorel, M. Mayero, J.-M. Muller, I. Pasca, L. Rideau (Marelle), and L. Théry (Marelle) have worked on the problem of offering guaranteed error bounds for a specific kind of rigorous polynomial approximation called Taylor model [26]. They carry out this work in

the Coq proof assistant, with a special focus on genericity and efficiency for our implementation. They give an abstract interface for rigorous polynomial approximations, parameterized by the type of coefficients and the implementation of polynomials, and they instantiate this interface to the case of Taylor models with interval coefficients, while providing all the machinery for computing them.

CARAMEL Project-Team

6. New Results

6.1. Sieve for FFS

Participants: Jérémie Detrey, Pierrick Gaudry [contact], Marion Videau.

Jérémie Detrey, Pierrick Gaudry and Marion Videau have worked on the relation collection step of the Function Field Sieve and especially on its implementation. This is still an ongoing work but the first results have been accepted for publication [13] in the ARITH-2013 conference.

6.2. Bilinear Maps

Participants: Răzvan Bărbulescu, Jérémie Detrey, Nicolas Estibals, Paul Zimmermann [contact].

As a result of an internal working group in the team, we have found and published at the WAIFI conference a new algorithm to find optimal formulae for bilinear maps [8]. This algorithm enables one to rediscover Karatsuba's multiplication algorithm, but has many other applications, for example to matrix multiplication.

6.3. Number Field Sieve

Participants: Emmanuel Thomé, Paul Zimmermann [contact].

Together with Shi Bai (Australian National University), E. Thomé and P. Zimmermann used CADO-NFS to factor RSA-704, a 212-digit number, to check scalability of the software on large factorizations [10]. This is the second largest number factored by any GNFS software so far, and the largest one factored by CADO-NFS. This experiment was very helpful, since it demonstrated several weaknesses of the code, that have been addressed since then.

Together with Shi Bai (Australian National University), P. Zimmermann wrote a preprint describing the algorithm used in CADO-NFS for the size-optimization of sextic polynomials [11].

Alain Filbois, Shi Bai (Australian National University) and P. Zimmermann improved the polynomial selection code. With parameters used to find good polynomials for RSA-896, a total speedup by a factor 14 was obtained, with both algorithmic and implementation improvements.

6.4. Sparse linear algebra modulo p

Participants: Hamza Jeljeli, Emmanuel Thomé [contact].

The resolution of linear algebra problems with subexponential methods, which is the topic of the ANR-CATREL project (to begin in 2013) calls for the resolution of large sparse linear systems defined over finite fields. In preparation for this, H. Jeljeli has developed software for performing sparse matrix times vector multiplication on NVIDIA GPUS [16]. This code provides a very significant speedup over the use of CPUs for this task, and achieves this speedup by a clever use of a "residue number system" representation of the finite field elements.

As a complement, a recent re-implementation of Thomé's algorithm for the (matrix) Berlekamp-Massey step in the block Wiedemann algorithm has been done. This program can of course be special-cased to the simple non-matrix case. The GPU code above and this special case, together, form the needed software to have a sparse linear system solver over finite fields using Wiedemann's algorithm. This has been put to use, and led to the completion of a discrete logarithm record in $\mathbb{F}_{2^{619}}$, the linear system part taking only 17 hours in total on one GPU (plus 1 hour on one CPU for the Berlekamp-Massey step).

6.5. Using symmetries in elliptic curve discrete logarithm

Participant: Pierrick Gaudry.

In a joint work by Jean-Charles Faugère, Pierrick Gaudry, Louise Huot and Guénaël Renault, it has been shown that the geometric symmetries of an elliptic curve, in particular, the symmetries of an Edwards curve, could be used to speed up the index calculus attack for computing discrete logarithms in an elliptic curve defined over an extension field. The corresponding article [14] is currently under revision.

6.6. Galois properties of curves for ECM

Participants: Răzvan Bărbulescu, Cyril Bouvier.

In collaboration with Joppe Bos, Peter Montgomery and Thorsten Kleinjung, Răzvan Bărbulescu and Cyril Bouvier proved some divisibility properties of the group order of an elliptic curve, using the Galois structure of its division polynomial. It explains the good behaviour of some curves that have been experimentally found to factor more numbers than others, and gives a way to find new curves with this property. The corresponding article [7] was presented in ANTS-X.

6.7. Computation of CM class polynomials for genus 2 Jacobians

Participants: Sorina Ionica, Emmanuel Thomé [contact].

In collaboration with Andreas Enge, Emmanuel Thomé has developed software for computing class polynomials, in the context of complex multiplication theory in genus 2. The current computations set new records which are well above the previous state of the art. A publication is in the works.

Using similar underlying tools and theory, and based on work by Sorina Ionica [15], Sorina Ionica and Emmanuel Thomé have worked on the analysis of isogeny graphs in genus 2, when certain properties of the endomorphism ring are satisfied.

6.8. Filtering step for NFS and FFS

Participant: Cyril Bouvier.

Cyril Bouvier studied the filtering step for the Number Field Sieve. A better weight function, used during the clique removal step, was found which allows to construct smaller matrices for the linear algebra step. A preprint is avalaible [12]. The filtering step for the Function Field Sieve was written in CADO-NFS.

CASCADE Project-Team (section vide)

GALAAD Project-Team

6. New Results

6.1. Algebraic representations for geometric modeling

6.1.1. Fitting ideals and multiple-points of surface parameterizations

Participants: Nicolàs Botbol, Laurent Busé.

Given a birational parameterization ϕ of an algebraic surface $\mathcal{S}$ in the projective space $\mathbb{P}^3$, the purpose of this ongoing work is to investigate the sets of points $D_k(\phi)$ on $\mathcal{S}$ whose preimage consists in k or more points, counting multiplicity. Our main result is an explicit description of these algebraic sets $D_k(\phi)$ in terms of Fitting ideals of some graded parts of a symmetric algebra associated to the parameterization ϕ.

This work is done in collaboration with Marc Chardin (University Pierre et Marie Curie).

6.1.2. Algebraic geometry tools for the study of entanglement: an application to spin squeezed states

Participant: Alessandra Bernardi.

In [18] a short review of Algebraic Geometry tools for the decomposition of tensors and polynomials is given from the point of view of applications to quantum and atomic physics. Examples of application to assemblies of indistinguishable two-level bosonic atoms are discussed using modern formulations of the classical Sylvester's algorithm for the decomposition of homogeneous polynomials in two variables. In particular, the symmetric rank and symmetric border rank of spin squeezed states is calculated as well as their Schrödinger-cat-like decomposition as the sum of macroscopically different coherent spin states; Fock states provide an example of states for which the symmetric rank and the symmetric border rank are different.

This is a joint work with I. Carusotto (University of Trento, Italy).

6.1.3. A partial stratification of secant varieties of Veronese varieties via curvilinear subschemes.

Participant: Alessandra Bernardi.

In [11] we give a partial quasi-stratification of the secant varieties of the order d Veronese variety $X_{m,d}$ of $\mathbb{P}^m$. It covers the set $\sigma_t(X_{m,d})^\dagger$ of all points lying on the linear span of curvilinear subschemes of $X_{m,d}$, but two quasi-strata may overlap. For low border rank, two different quasi-strata are disjoint and we compute the symmetric rank of their elements. Our tool is the Hilbert schemes of curvilinear subschemes of Veronese varieties. To get a stratification we attach to each $P \in \sigma_t(X_{m,d})^\dagger$ the minimal label of a quasi-stratum containing it.

This is a joint work with E. Ballico (University of Trento, Italy).

6.1.4. Decomposition of homogeneous polynomials with low rank.

Participant: Alessandra Bernardi.

Let F be a homogeneous polynomial of degree d in $m+1$ variables defined over an algebraically closed field of characteristic 0 and suppose that F belongs to the s-th secant variety of the d-uple Veronese embedding of $\mathbb{P}^m$ into $\mathbb{P}^{\binom{m+d}{d}-1}$ but that its minimal decomposition as a sum of d-th powers of linear forms $M_1, ..., M_r$ is $F = M_1^d + \cdots + M_r^d$ with $r > s$. In [12], we show that if $s + r \leq 2d + 1$ then such a decomposition of F can be split into two parts: one of them is made by linear forms that can be written using only two variables. The other part is uniquely determined once one has fixed the first part. We also obtain a uniqueness theorem for the minimal decomposition of F if r is at most d and a mild condition is satisfied.

This is a joint work with E. Ballico (University of Trento, Italy).

6.1.5. Higher secant varieties of $\mathbb{P}^n \times \mathbb{P}^1$ embedded in bi-degree (a, b)

Participant: Alessandra Bernardi.

In [15], we compute the dimension of all the higher secant varieties to the Segre-Veronese embedding of $\mathbb{P}^n \times \mathbb{P}^1$ via the section of the sheaf $\mathcal{O}(a, b)$ for any $n, a, b \in \mathbb{Z}^+$. We relate this result to the Grassmann Defectivity of Veronese varieties and we classify all the Grassmann $(1, s-1)$-defective Veronese varieties.

This is a joint work with E. Ballico, M. V. Catalisano (University of Trento, Italy).

6.1.6. Symmetric tensor rank with a tangent vector: a generic uniqueness theorem

Participant: Alessandra Bernardi.

Let $X_{m,d} \subset \mathbb{P}^N$, $N := \binom{m+d}{m} - 1$, be the order d Veronese embedding of $\mathbb{P}^m$. Let $\tau(X_{m,d}) \subset \mathbb{P}^N$, be the tangent developable of $X_{m,d}$. For each integer $t \geq 2$ let $\tau(X_{m,d}, t) \subseteq \mathbb{P}^N$, be the join of $\tau(X_{m,d})$ and $t-2$ copies of $X_{m,d}$. In [13], we prove that if $m \geq 2$, $d \geq 7$ and $t \leq 1 + \lfloor \binom{m+d-2}{m} / (m+1) \rfloor$, then for a general $P \in \tau(X_{m,d}, t)$ there are uniquely determined $P_1, \cdots, P_{t-2} \in X_{m,d}$ and a unique tangent vector ν of $X_{m,d}$ such that P is in the linear span of $\nu \cup \{P_1, \cdots, P_{t-2}\}$. In other words, a degree d linear form f (a symmetric tensor T of order d) associated to P may be written as

$$f = L_{t-1}^{d-1} L_t + \sum_{i=1}^{t-2} L_i^d, \quad (T = v_{t-1}^{\otimes(d-1)} v_t + \sum_{i=1}^{t-2} v_i^{\otimes d})$$

with L_i linear forms on $\mathbb{P}^m$ (v_i vectors over a vector field of dimension $m+1$ respectively), $1 \leq i \leq t$, that are uniquely determined (up to a constant).

This is a joint work with E. Ballico (University of Trento, Italy).

6.1.7. General tensor decomposition, moment matrices and applications.

Participants: Alessandra Bernardi, Bernard Mourrain.

In [17] the tensor decomposition addressed may be seen as a generalisation of Singular Value Decomposition of matrices. We consider general multilinear and multihomogeneous tensors. We show how to reduce the problem to a truncated moment matrix problem and give a new criterion for flat extension of Quasi-Hankel matrices. We connect this criterion to the commutation characterisation of border bases. A new algorithm is described. It applies for general multihomogeneous tensors, extending the approach of J.J. Sylvester to binary forms. An example illustrates the algebraic operations involved in this approach and how the decomposition can be recovered from eigenvector computation.

This is a joint work with J. Brachat and P. Comon (i3S, CNRS).

6.1.8. On the cactus rank of cubic forms

Participant: Alessandra Bernardi.

In this work, we prove that the smallest degree of an apolar 0-dimensional scheme of a general cubic form in $n+1$ variables is at most $2n+2$, when $n \geq 8$, and therefore smaller than the rank of the form. For the general reducible cubic form the smallest degree of an apolar subscheme is $n+2$, while the rank is at least $2n$.

This is a joint work with K. Ranestad (University of Oslo, Norway) that will be published in 2013 in the Journal of Symbolic Computation. The preprint is available at http://hal.inria.fr/inria-00630456.

6.1.9. Tensor ranks on tangent developable of Segre varieties

Participant: Alessandra Bernardi.

In [14] we describe the stratification by tensor rank of the points belonging to the tangent developable of any Segre variety. We give algorithms to compute the rank and a decomposition of a tensor belonging to the secant variety of lines of any Segre variety. We prove Comon's conjecture on the rank of symmetric tensors for those tensors belonging to tangential varieties to Veronese varieties.

This is a joint work with E. Ballico (University of Trento, Italy).

6.1.10. On the dimension of spline spaces on planar T-meshes

Participant: Bernard Mourrain.

In [33], we analyze the space of bivariate functions that are piecewise polynomial of bi-degree $\leq (m, m')$ and of smoothness r along the interior edges of a planar T-mesh. We give new combinatorial lower and upper bounds for the dimension of this space by exploiting homological techniques. We relate this dimension to the weight of the maximal interior segments of the T-mesh, defined for an ordering of these maximal interior segments. We show that the lower and upper bounds coincide, for high enough degrees or for hierarchical T-meshes which are enough regular. We give a rule of subdivision to construct hierarchical T-meshes for which these lower and upper bounds coincide. Finally, we illustrate these results by analyzing spline spaces of small degrees and smoothness.

6.1.11. On the problem of instability in the dimension of a spline space over a T-mesh

Participant: Bernard Mourrain.

In [23], we discuss the problem of instability in the dimension of a spline space over a T-mesh. For bivariate spline spaces $S(5, 5, 3, 3)$ and $S(4, 4, 2, 2)$, the instability in the dimension is shown over certain types of T-meshes. This result could be considered as an attempt to answer the question of how large the polynomial degree (m, m') should be relative to the smoothness (r, r') to make the dimension of a spline space stable. We show in particular that the bound $m \geq 2r + 1$ and $m' \geq 2r' + 1$ are optimal.

This is a joint work with Berdinsky Dmitry, Oh Min-Jae and Kim Taewan (Department of Naval Architecture and Ocean Engineering Seoul National University, South Korea).

6.1.12. Homological techniques for the analysis of the dimension of triangular spline spaces

Participant: Bernard Mourrain.

The spline space $C_k^r(\Delta)$ attached to a subdivided domain Δ of $\mathbb{R}^d$ is the vector space of functions of class C^r which are polynomials of degree $\leq k$ on each piece of this subdivision. Classical splines on planar rectangular grids play an important role in Computer Aided Geometric Design, and spline spaces over arbitrary subdivisions of planar domains are now considered for isogeometric analysis applications. In [34], we address the problem of determining the dimension of the space of bivariate splines $C_k^r(\Delta)$ for a triangulated region Δ in the plane. Using the homological introduced by Billera (1988), we number the vertices and establish a formula for an upper bound on the dimension. There is no restriction on the ordering and we obtain more accurate approximations to the dimension than previous methods. Furthermore, in certain cases even an exact value can be found. The construction makes it also possible to get a short proof for the dimension formula when $k \geq 4r + 1$, and the same method we use in this proof yields the dimension straightaway for many other cases.

This is a joint work with Nelly Villamizar (CMA, University of Oslo, Norway).

6.1.13. Analysis-suitable volume parameterization of multi-block computational domain in isogeometric applications

Participants: Bernard Mourrain, André Galligo.

Parameterization of computational domain is a key step in isogeometric analysis just as mesh generation is in finite element analysis. In [36], we study the volume parameterization problem of multi-block computational domain in isogeometric version, i.e., how to generate analysis-suitable parameterization of the multi-block computational domain bounded by B-spline surfaces. Firstly, we show how to find good volume parameterization of single-block computational domain by solving a constraint optimization problem, in which the constraint condition is the injectivity sufficient conditions of B-spline volume parametrization, and the optimization term is the minimization of quadratic energy functions related to the first and second derivatives of B-spline volume parameterization. By using this method, the resulted volume parameterization has no self-intersections, and the isoparametric structure has good uniformity and orthogonality. Then we extend this method to the multi-block case, in which the continuity condition between the neighbor B-spline volume should be added to the constraint term. The effectiveness of the proposed method is illustrated by several examples based on three-dimensional heat conduction problem.

This is a joint work with Régis Duvigneau (Inria, EPI OPALE) and Xu Gang (College of computer - Hangzhou Dianzi University, China).

6.1.14. A new error assessment method in isogeometric analysis of 2D heat conduction problems

Participants: Bernard Mourrain, André Galligo.

In [35], we propose a new error assessment method for isogeometric analysis of 2D heat conduction problems. A posteriori error estimation is obtained by resolving the isogeometric analysis problem with several k-refinement steps. The main feature of the proposed method is that the resulted error estimation surface has a B-spline form, according to the main idea of isogeometric analysis. Though the error estimation method is expensive, it can be used as an error assessment method for isogeometric analysis. Two comparison examples are presented to show the efficiency of the proposed method.

This is a joint work with Régis Duvigneau (Inria, EPI OPALE) and Xu Gang (College of computer - Hangzhou Dianzi University, China).

6.1.15. On the cut-off phenomenon for the transitivity of randomly generated subgroups

Participant: André Galligo.

Consider $K \geq 2$ independent copies of the random walk on the symmetric group S_N starting from the identity and generated by the products of either independent uniform transpositions or independent uniform neighbor transpositions. At any time $n \in \mathbb{N}$, let G_n be the subgroup of S_N generated by the K positions of the chains. In the uniform transposition model, we prove in [28] that there is a cut-off phenomenon at time $N\ln(N)/(2K)$ for the non-existence of fixed point of G_n and for the transitivity of G_n, thus showing that these properties occur before the chains have reached equilibrium. In the uniform neighbor transposition model, a transition for the non-existence of a fixed point of G_n appears at time of order $N^{1+\frac{2}{K}}$ (at least for $K \geq 3$), but there is no cut-off phenomenon. In the latter model, we recover a cut-off phenomenon for the non-existence of a fixed point at a time proportional to N by allowing the number K to be proportional to $\ln(N)$. The main tools of the proofs are spectral analysis and coupling techniques.

This is a joint work with Laurent Miclo (University of Toulouse).

6.2. Algebraic algorithms for geometric computing

6.2.1. On the isotopic meshing of an algebraic implicit surface

Participant: Bernard Mourrain.

In [22], we present a new and complete algorithm for computing the topology of an algebraic surface given by a squarefree polynomial in $\mathbb{Q}[X, Y, Z]$. Our algorithm involves only subresultant computations and entirely relies on rational manipulation, which makes it direct to implement. We extend the work in [15], on the topology of non-reduced algebraic space curves, and apply it to the polar curve or apparent contour of the surface S. We exploit simple algebraic criterion to certify the pseudo-genericity and genericity position of the surface. This gives us rational parametrizations of the components of the polar curve, which are used to lift the topology of the projection of the polar curve. We deduce the connection of the two-dimensional components above the cell defined by the projection of the polar curve. A complexity analysis of the algorithm is provided leading to a bound in $\widetilde{\mathcal{O}}_B(d^{15}\tau)$ for the complexity of the computation of the topology of an implicit algebraic surface defined by integer coefficients polynomial of degree d and coefficients size τ . Examples illustrate the implementation in Mathemagix of this first complete code for certified topology of algebraic surfaces.

This is a joint work with Daouda Niang Diatta, Olivier Ruatta (XLIM, University of Limoges).

6.2.2. Moment matrices, border basis and real radical computation

Participant: Bernard Mourrain.

In [32], we describe new methods to compute the radical (resp. real radical) of an ideal, assuming its complex (resp. real) variety is finite. The aim is to combine approaches for solving a system of polynomial equations with dual methods which involve moment matrices and semi-definite programming. While border basis algorithms are efficient and numerically stable for computing complex roots, algorithms based on moment matrices allow the incorporation of additional polynomials, e.g., to restrict the computation to real roots or to eliminate multiple solutions. The proposed algorithm can be used to compute a border basis of the input ideal and, as opposed to other approaches, it can also compute the quotient structure of the (real) radical ideal directly, i.e., without prior algebraic techniques such as Gröbner bases. It thus combines the strengths of existing algorithms and provides a unified treatment for the computation of border bases for the ideal, the radical ideal and the real radical ideal.

This is a joint work with Jean-Bernard Lasserre (LAAS, Toulouse), Monique Laurent (CWI, Amsterdam, Netherland), Philipp Rostalski (University of California, Berkeley, US) Philippe Trébuchet (APR, LIP6, Paris).

6.2.3. On the computation of matrices of traces and radicals of ideals

Participant: Bernard Mourrain.

Let $f_1, ..., f_s \in \mathbb{K}[x_1, ..., x_m]$ be a system of polynomials generating a zero-dimensional ideal I, where $\mathbb{K}$ is an arbitrary algebraically closed field. In [31], we study the computation of "matrices of traces" for the factor algebra $\mathcal{A} := \mathbb{C}[x_1, ..., x_m]/I$, i.e. matrices with entries which are trace functions of the roots of I. Such matrices of traces in turn allow us to compute a system of multiplication matrices $\{M_{x_i} | i = 1, ..., m\}$ of the radical $\sqrt{I}$. We first propose a method using Macaulay type resultant matrices of $f_1, ..., f_s$ and a polynomial J to compute moment matrices, and in particular matrices of traces for $\mathcal{A}$. Here J is a polynomial generalizing the Jacobian. We prove bounds on the degrees needed for the Macaulay matrix in the case when I has finitely many projective roots in $\mathbb{P}^m_{\mathbb{C}}$. We also extend previous results which work only for the case where $\mathcal{A}$ is Gorenstein to the non-Gorenstein case. The second proposed method uses Bezoutian matrices to compute matrices of traces of $\mathcal{A}$. Here we need the assumption that $s = m$ and $f_1, ..., f_m$ define an affine complete intersection. This second method also works if we have higher dimensional components at infinity. A new explicit description of the generators of $\sqrt{I}$ are given in terms of Bezoutians.

This is a joint work with Itnuit Janovitz-Freireich (Departamento de Matemáticas, Mexico), Lajos Ronayi (Hungarian Academy of Sciences and Budapest University of Technology and Economics, Budapest), Agnes Szanto (Department of Computer Science, North Carolina State University, US).

6.2.4. Border basis representation of a general quotient algebra

Participant: Bernard Mourrain.

In [40], we generalized the construction of border bases to non-zero dimensional ideals for normal forms compatible with the degree, tackling the remaining obstacle for a general application of border basis methods. First, we give conditions to have a border basis up to a given degree. Next, we describe a new stopping criterion to determine when the reduction with respect to the leading terms is a normal form. This test based on the persistence and regularity theorems of Gotzmann yields a new algorithm for computing a border basis of any ideal, which proceeds incrementally degree by degree until its regularity. We detail it, prove its correctness, present its implementation and report some experimentations which illustrate its practical good behavior.

This is a joint work with Philippe Trébuchet (APR, LIP6, Paris).

6.2.5. *Voronoï diagrams of algebraic distance fields*

Participant: Bernard Mourrain.

In [25], we design and implement an efficient and certified algorithm for the computation of Voronoi Diagrams (VD's) constrained to a given domain. Our framework is general and applicable to any VD-type where the distance field is given explicitly or implicitly by a polynomial, notably the anisotropic VD or VD's of non-punctual sites. We use the Bernstein form of polynomials and DeCasteljau's algorithm to subdivide the initial domain and isolate bisector, or domains that contain a Voronoi vertex. The efficiency of our algorithm is due to a filtering process, based on bounding the field over the subdivided domains. This allows us to exclude functions (thus sites) that do not contribute locally to the lower envelope of the lifted diagram. The output is a polygonal description of each Voronoi cell, within any user-defined precision, isotopic to the exact VD. Correctness of the result is implied by the certified approximations of bisector branches, which are computed by existing methods for handling algebraic curves. First experiments with our C++ implementation, based on double precision arithmetic, demonstrate the adaptability of the algorithm.

This is a joint work with Ioannis Emiris (ERGA, National Kapodistrian University of Athens, Greece), Angelos Mantzaflaris (RICAM, Austrian Academy of Sciences, Austria).

6.2.6. *Rational invariants of scalings from Hermite normal forms*

Participant: Evelyne Hubert.

Scalings form a class of group actions that have both theoretical and practical importance. A scaling is accurately described by an integer matrix. In [39] tools from linear algebra are exploited to compute a minimal generating set of rational invariants, trivial rewriting and rational sections for such a group action. The primary tools used are Hermite normal forms and their unimodular multipliers. With the same line of ideas, a complete solution to the scaling symmetry reduction of a polynomial system is also presented.

This is joint work with George Labahn (University of Waterloo, Canada).

6.2.7. *Scaling invariants and symmetry reduction of dynamical systems*

Participant: Evelyne Hubert.

The motivation for this subject is to offer an algorithmic scheme for reducing the number of parameters in physical, chemical or biological models. This comes as a special case of a symmetry reduction scheme that can be fully realized by linear algebra over the integers. See http://hal.inria.fr/hal-00668882. We provide there the algebraic determination of the scaling symmetry of a dynamical system and an complete explicit symmetry reduction scheme with polynomial complexity.

This is joint work with George Labahn (University of Waterloo, Canada).

6.2.8. *A computational approach to the discriminant of homogeneous polynomials*

Participant: Laurent Busé.

In this work, the discriminant of homogeneous polynomials is studied in two particular cases: a single homogeneous polynomial and a collection of $n-1$ homogeneous polynomials in n variables. In these two cases, the discriminant is defined over a large class of coefficient rings by means of the resultant. Many formal properties and computational rules are provided and the geometric interpretation of the discriminant is investigated over a general coefficient ring, typically a domain.

This work is done in collaboration with Jean-Pierre Jouanolou (University of Strasbourg). A preprint is available at http://hal.inria.fr/hal-00747930/en/.

6.2.9. Intersection between rational curves and surfaces by means of matrix representations

Participant: Laurent Busé.

In [37], we propose a survey of matrix representations for parameterized curves and surfaces. Illustrations of the properties of these representations are given for intersection problems. In particular, we focus on the ray/surface intersection which is an important step in ray-tracing algorithms.

6.2.10. A root isolation algorithm for sparse univariate polynomials

Participant: André Galligo.

In [38], we consider a univariate polynomial f with real coefficients having a high degree N but a rather small number $d+1$ of monomials, with $d \ll N$. Such a sparse polynomial has a number of real roots smaller or equal to d. Our target is to find for each real root of f an interval isolating this root from the others. The usual subdivision methods, relying either on Sturm sequences or Moebius transform followed by Descartes's rule of signs, destruct the sparse structure. Our approach relies on the generalized Budan-Fourier theorem of Coste, Lajous, Lombardi, Roy and the techniques developed in some previous works of Galligo. To such a f is associated a set of $d+1\mathbb{F}$-derivatives. The Budan-Fourier function $V_f(x)$ counts the sign changes in the sequence of $\mathbb{F}$-derivatives of f evaluated at x. The values at which this function jumps are called the $\mathbb{F}$-virtual roots of f. These include the real roots of f. We also consider the augmented $\mathbb{F}$-virtual roots of f and introduce a genericity property which eases our study. We present a real root isolation method and an algorithm which has been implemented in Maple. We rely on an improved generalized Budan-Fourier count applied to both the input polynomial and its reciprocal, together with Newton like approximation steps.

This is a joint work with Maria Emilia Alonso (University of Madrid).

6.2.11. Deformation of roots of polynomials via fractional derivatives

Participant: André Galligo.

In [26], we first recall the main features of Fractional calculus. In the expression of fractional derivatives of a real polynomial $f(x)$, we view the order of differentiation q as a new indeterminate; then we define a new bivariate polynomial $Pf(x,q)$. For $0 \leq q \leq 1$, $Pf(x,q)$ defines a homotopy between the polynomials $f(x)$ and $xf'(x)$. Iterating this construction, we associate to $f(x)$ a plane spline curve, called the stem of f. Stems of classic random polynomials exhibits intriguing patterns; moreover in the complex plane $Pf(x,q)$ creates an unexpected correspondence between the complex roots and the critical points of $f(x)$. We propose 3 conjectures to describe and explain these phenomena. Illustrations are provided relying on the Computer Algebra System Maple.

GEOMETRICA Project-Team

6. New Results

6.1. Mesh Generation and Geometry Processing

6.1.1. New bounds on the size of optimal meshes

Participant: Donald Sheehy.

The theory of optimal size meshes gives a method for analyzing the output size (number of simplices) of a Delaunay refinement mesh in terms of the integral of a sizing function over the input domain. The input points define a maximal such sizing function called the feature size. This work aims to find a way to bound the feature size integral in terms of an easy to compute property of a suitable ordering of the point set. The key idea is to consider the pacing of an ordered point set, a measure of the rate of change in the feature size as points are added one at a time. In previous work, Miller et al. showed that if an ordered point set has pacing ϕ, then the number of vertices in an optimal mesh will be $O(\phi^d n)$, where d is the input dimension. We give a new analysis of this integral showing that the output size is only $\Theta(n + n \log \phi)$. The new analysis tightens bounds from several previous results and provides matching lower bounds. Moreover, it precisely characterizes inputs that yield outputs of size $O(n)$ [20].

6.1.2. State of the art in quad meshing

Participant: David Bommes.

Triangle meshes have been nearly ubiquitous in computer graphics, and a large body of data structures and geometry processing algorithms based on them has been developed in the literature. At the same time, quadrilateral meshes, especially semi-regular ones, have advantages for many applications, and significant progress was made in quadrilateral mesh generation and processing during the last several years. In this work, we discuss the advantages and problems of techniques operating on quadrilateral meshes, including surface analysis and mesh quality, simplification, adaptive refinement, alignment with features, parametrization, and remeshing [23].

6.1.3. Meshing the hyperbolic octagon

Participants: Mathieu Schmitt, Monique Teillaud.

We propose a practical method to compute a mesh of the octagon, in the Poincaré disk, that respects its symmetries. This is obtained by meshing the Schwartz triangle $T(8, 3, 2)$ and applying relevant hyperbolic symmetries (ie., Euclidean reflexions or inversions). The implementation is based on CGAL 2D meshes and on the ongoing implementation on CGAL hyperbolic Delaunay triangulations [44]. Further work will include solving robutsness issues and generalizing the method to any Schwartz triangle [62].

6.1.4. Index-based data structure for 3D polytopal complexes

Participant: David Bommes.

OpenVolumeMesh is a data structure which is able to represent heterogeneous 3-dimensional polytopal cell complexes and is general enough to also represent non-manifolds without incurring undue overhead [30]. Extending the idea of half-edge based data structures for two-manifold surface meshes, all faces, i.e. the two-dimensional entities of a mesh, are represented by a pair of oriented half-faces. The concept of using directed half-entities enables inducing an orientation to the meshes in an intuitive and easy to use manner. We pursue the idea of encoding connectivity by storing first-order top-down incidence relations per entity, i.e. for each entity of dimension d, a list of links to the respective incident entities is stored. For instance, each half-face as well as its orientation is uniquely determined by a tuple of links to its incident half-edges or each 3D cell by the set of incident half-faces. This representation allows for handling non-manifolds as well as mixed-dimensional mesh configurations. No entity is duplicated according to its valence, instead, it is shared by all incident entities in order to reduce memory consumption. Furthermore, an array-based storage layout is used in combination with direct index-based access. This guarantees constant access time to the entities of a mesh. Although bottom-up incidence relations are implied by the top-down incidences, our data structure provides the option to explicitly generate and cache them in a transparent manner. This allows for accelerated navigation in the local neighbor- hood of an entity. We provide an open-source and platform-independent implementation of the proposed data structure written in C++ using dynamic typing paradigms. The library is equipped with a set of STL compliant iterators, a generic property system to dynamically attach properties to all entities at run-time, and a serializer/deserializer supporting a simple file format. Due to its similarity to the OpenMesh data structure, it is easy to use, in particular for those familiar with OpenMesh. Since the presented data structure is compact, intuitive, and efficient, it is suitable for a variety of applications, such as meshing, visualization, and numerical analysis. OpenVolumeMesh is open-source software licensed under the terms of the LGPL [29].

6.1.5. *Editable SQuad representation for triangle meshes*

Participant: Olivier Devillers.

In collaboration with Luca Castelli Aleardi (LIX, Palaiseau) and Jarek Rossignac (Georgia Tech).

We consider the problem of designing space efficient solutions for representing the connectivity information of manifold triangle meshes. Most mesh data structures are quite redundant, storing a large amount of information in order to efficiently support mesh traversal operators. Several compact data structures have been proposed to reduce storage cost while supporting constant-time mesh traversal. Some recent solutions are based on a global re-ordering approach, which allows to implicitly encode a map between vertices and faces. Unfortunately, these compact representations do not support efficient updates, because local connectivity changes (such as edge-contractions, edge-flips or vertex insertions) require re-ordering the entire mesh. Our main contribution is to propose a new way of designing compact data structures which can be dynamically maintained. In our solution, we push further the limits of the re-ordering approaches: the main novelty is to allow to re-order vertex data (such as vertex coordinates), and to exploit this vertex permutation to easily maintain the connectivity under local changes. We describe a new class of data structures, called Editable SQuad (ESQ), offering the same navigational and storage performance as previous works, while supporting local editing in amortized constant time. As far as we know, our solution provides the most compact dynamic data structure for triangle meshes. We propose a linear-time and linear-space construction algorithm, and provide worst-case bounds for storage and time cost [25].

6.1.6. *Surface reconstruction through point set structuring*

Participants: Pierre Alliez, Florent Lafarge.

We present a method for reconstructing surfaces from point sets. The main novelty lies into a structure-preserving approach where the input point set is first consolidated by structuring and resampling the planar components, before reconstructing the surface from both the consolidated components and the unstructured points. The final surface is obtained through solving a graph-cut problem formulated on the 3D Delaunay triangulation of the structured point set where the tetrahedra are labeled as inside or outside cells. Structuring facilitates the surface reconstruction as the point set is substantially reduced and the points are enriched with structural meaning related to adjacency between primitives. Our approach departs from the common dichotomy between smooth/piecewise-smooth and primitive-based representations by gracefully combining

canonical parts from detected primitives and free-form parts of the inferred shape. Our experiments on a variety of inputs illustrate the potential of our approach in terms of robustness, flexibility and efficiency [59].

6.1.7. Feature-preserving surface reconstruction and simplification from defect-laden point sets

Participants: Pierre Alliez, David Cohen-Steiner, Julie Digne.

In collaboration with Fernando de Goes and Mathieu Desbrun from Caltech.

We introduce a robust and feature-capturing surface reconstruction and simplification method that turns an input point set into a low triangle-count simplicial complex. Our approach starts with a (possibly non-manifold) simplicial complex filtered from a 3D Delaunay triangulation of the input points. This initial approximation is iteratively simplified based on an error metric that measures, through optimal transport, the distance between the input points and the current simplicial complex, both seen as mass distributions. Our approach is shown to exhibit both robustness to noise and outliers, as well as preservation of sharp features and boundaries (Figure 1). Our new feature-sensitive metric between point sets and triangle meshes can also be used as a post-processing tool that, from the smooth output of a reconstruction method, recovers sharp features and boundaries present in the initial point set [58].

6.1.8. Similarity based filtering of point clouds

Participant: Julie Digne.

Denoising surfaces is a crucial step in the surface processing pipeline. This is even more challenging when no underlying structure of the surface is known, that is when the surface is represented as a set of unorganized points. We introduce a denoising method based on *local similarities*. The contributions are threefold: first, we do not denoise directly the point positions but use a low/high frequency decomposition and denoise only the high frequency. Second, we introduce a local surface parameterization which is proved stable. Finally, this method works directly on point clouds, thus avoiding building a mesh of a noisy surface which is a difficult problem. Our approach is based on denoising a height vector field by comparing the neighborhood of the point with neighborhoods of other points on the surface (Figure 2). It falls into the non-local denoising framework that has been extensively used in image processing, but extends it to unorganized point clouds [26].

6.1.9. Progressive compression of manifold polygon meshes

Participant: Pierre Alliez.

In collaboration with Adrien Maglo, Clément Courbet and Céline Hudelot from Ecole Centrale Paris.

We present a new algorithm for the progressive compression of surface polygon meshes. The input surface is decimated by several traversals that generate successive levels of detail through a specific patch decimation operator which combines vertex removal and local remeshing. This operator encodes the mesh connectivity through a transformation that generates two lists of Boolean symbols during face and edge removals. The geometry is encoded with a barycentric error prediction of the removed vertex coordinates. In order to further reduce the size of the geometry and connectivity data, we propose a curvature prediction method and a connectivity prediction scheme based on the mesh geometry. We also include two methods that improve the rate-distortion performance: a wavelet formulation with a lifting scheme and an adaptive quantization technique. Experimental results demonstrate the effectiveness of our approach in terms of compression rates and rate-distortion performance. Our approach compares favorably to compression schemes specialized to triangle meshes [31].

6.2. Topological and Geometric Inference

6.2.1. Homological reconstruction and simplification in $\mathbb{R}^3$

Participants: Olivier Devillers, Marc Glisse.

In collaboration with Dominique Attali (Gipsa-lab), Ulrich Bauer (Göttingen Univ.), and André Lieutier (Dassault Systèmes).

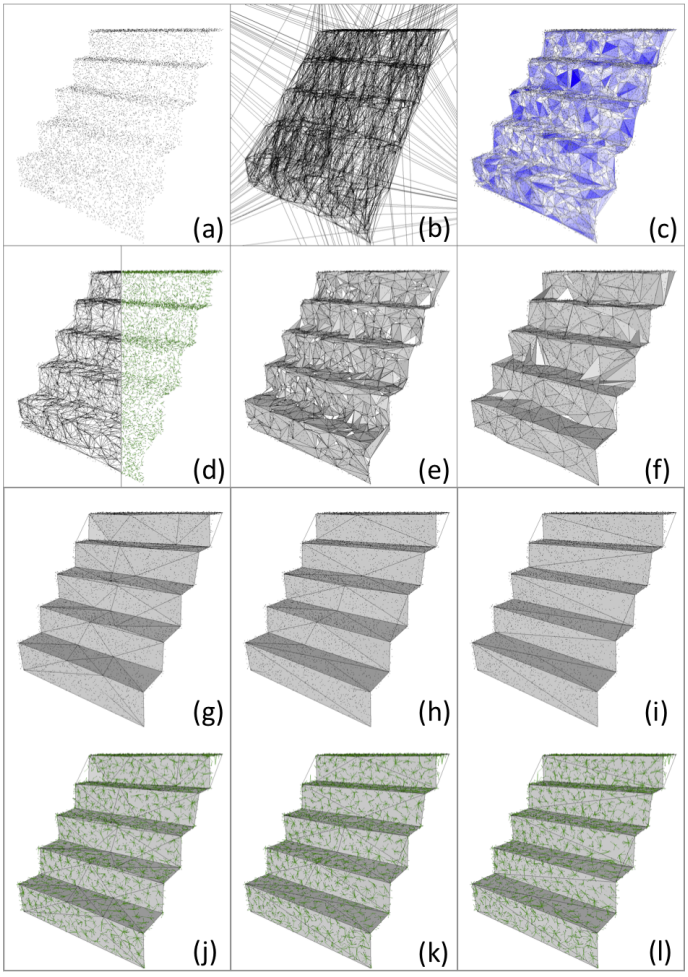

Figure 1. Steps of our algorithm: (a) Initial point set; (b) 3D Delaunay triangulation of a random subset containing 10% *of the input points; (c) Initial simplicial complex constructed from facets of the 3D triangulation with non-zero measure; (d) Initial transport plan assigning point samples to bin centroids (green arrows); (e-f) Intermediary decimation steps; (g-i) Reconstruction with 100, 50, and 22 vertices, respectively; (j-l) Final transport plan with 100, 50, and 22 vertices, respectively.*

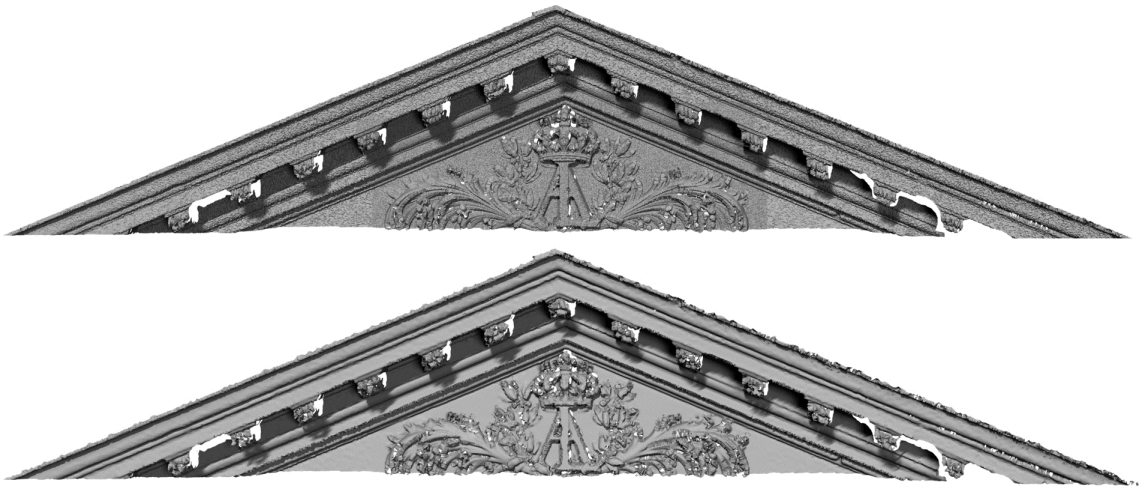

Figure 2. Similarity-based denoising. Top: input point set. Bottom: point set after denoising.

We consider the problem of deciding whether the persistent homology group of a simplicial pair (K, L) can be realized as the homology $H_*(X)$ of some space X with $L \subset X \subset K$. We show that this problem is NP-complete even if K is embedded in $\mathbb{R}^3$.

As a consequence, we show that it is NP-hard to simplify level and sublevel sets of scalar functions on $\mathbb{S}^3$ within a given tolerance constraint. This problem has relevance to the visualization of medical images by isosurfaces. We also show an implication to the theory of well groups of scalar functions: not every well group can be realized by some level set, and deciding whether a well group can be realized is NP-complete [43].

6.2.2. The structure and stability of persistence modules

Participants: Frédéric Chazal, Marc Glisse, Steve Oudot.

In collaboration with Vin de Silva (Pomona College)

We give a self-contained treatment of the theory of persistence modules indexed over the real line. We give new proofs of the standard results. Persistence diagrams are constructed using measure theory. Linear algebra lemmas are simplified using a new notation for calculations on quiver representations. We show that the stringent finiteness conditions required by traditional methods are not necessary to prove the existence and stability of the persistence diagram. We introduce weaker hypotheses for taming persistence modules, which are met in practice and are strong enough for the theory still to work. The constructions and proofs enabled by our framework are, we claim, cleaner and simpler [54].

6.2.3. Persistence stability for geometric complexes

Participants: Frédéric Chazal, Steve Oudot.

In collaboration with Vin de Silva (Pomona College)

We study the properties of the homology of different geometric filtered complexes (such as Vietoris–Rips, Čech and witness complexes) built on top of precompact spaces. Using recent developments in the theory of topological persistence [54] we provide simple and natural proofs of the stability of the persistent homology of such complexes with respect to the Gromov–Hausdorff distance. We also exhibit a few noteworthy properties of the homology of the Rips and Čech complexes built on top of compact spaces [53].

6.2.4. Zigzag zoology: rips zigzags for homology inference

Participants: Steve Oudot, Donald Sheehy.

For points sampled near a compact set X, the persistence barcode of the Rips filtration built from the sample contains information about the homology of X as long as X satisfies some geometric assumptions. The Rips filtration is prohibitively large, however zigzag persistence can be used to keep the size linear. We present several species of Rips-like zigzags and compare them with respect to the signal-to-noise ratio, a measure of how well the underlying homology is represented in the persistence barcode relative to the noise in the barcode at the relevant scales. Some of these Rips-like zigzags have been available as part of the Dionysus library for several years while others are new. Interestingly, we show that some species of Rips zigzags will exhibit less noise than the (non-zigzag) Rips filtration itself. Thus, the Rips zigzag can offer improvements in both size complexity and signal-to-noise ratio.

Along the way, we develop new techniques for manipulating and comparing persistence barcodes from zigzag modules. We give methods for reversing arrows and removing spaces from a zigzag. We also discuss factoring zigzags and a kind of interleaving of two zigzags that allows their barcodes to be compared. These techniques were developed to provide our theoretical analysis of the signal-to-noise ratio of Rips-like zigzags, but they are of independent interest as they apply to zigzag modules generally [60].

6.2.5. A space and time efficient implementation for computing persistent homology

Participants: Jean-Daniel Boissonnat, Clément Maria.

In collaboration with Tamal Dey (Ohio State University)

The persistent homology with Z_2-coefficients coincides with the same for cohomology because of duality. Recently, it has been observed that the cohomology based algorithms perform much better in practice than the originally proposed homology based persistence algorithm. We have implemented a cohomology based algorithm that attaches binary labels called annotations with the simplices. This algorithm fits very naturally with our recently developed data structure called simplex tree to represent simplicial complexes [49], [22]. By taking advantages of several practical tricks such as representing annotations compactly with memory words, using a union-find structure that eliminates duplicate annotation vectors, and a lazy evaluation, we save both space and time cost for computations. The complexity of the procedure, in practice, depends almost linearly on the size of the simplicial complex and on the variables related to the maximal dimension of the local homology groups we maintain during the computation, which remain small in practice. We provide a theoretical analysis as well as a detailed experimental study of our implementation. Experimental results show that our implementation performs several times better than the existing state-of-the-art software for computing persistent homology in terms of both time and memory requirements and can handle very large (several hundred million simplices in high-dimension) complexes efficiently [45].

6.2.6. Minimax rates for homology inference

Participant: Donald Sheehy.

In collaboration with Sivaraman Balakrishnan and Alessandro Rinaldo and Aarti Singh and Larry A. Wasserman (Carnegie Mellon University)

Often, high dimensional data lie close to a low-dimensional submanifold and it is of interest to understand the geometry of these submanifolds. The homology groups of a manifold are important topological invariants that provide an algebraic summary of the manifold. These groups contain rich topological information, for instance, about the connected components, holes, tunnels and sometimes the dimension of the manifold. We consider the statistical problem of estimating the homology of a manifold from noisy samples under several different noise models. We derive upper and lower bounds on the minimax risk for this problem. Our upper bounds are based on estimators which are constructed from a union of balls of appropriate radius around carefully selected points. In each case, we establish complementary lower bounds using Le Cam's lemma [15].

6.2.7. Linear-size approximations to the Vietoris-Rips filtration

Participant: Donald Sheehy.

The Vietoris-Rips filtration is a versatile tool in topological data analysis. Unfortunately, it is often too large to construct in full. We show how to construct an $O(n)$-size filtered simplicial complex on an n-point metric space such that the persistence diagram is a good approximation to that of the Vietoris-Rips filtration. The filtration can be constructed in $O(n \log n)$ time. The constants depend only on the doubling dimension of the metric space and the desired tightness of the approximation. For the first time, this makes it computationally tractable to approximate the persistence diagram of the Vietoris-Rips filtration across all scales for large data sets. Our approach uses a hierarchical net-tree to sparsify the filtration. We can either sparsify the data by throwing out points at larger scales to give a zigzag filtration, or sparsify the underlying graph by throwing out edges at larger scales to give a standard filtration. Both methods yield the same guarantees [34].

6.2.8. A multicover nerve for geometric inference

Participant: Donald Sheehy.

We show that filtering the barycentric decomposition of a Čech complex by the cardinality of the vertices captures precisely the topology of k-covered regions among a collection of balls for all values of k. Moreover, we relate this result to the Vietoris-Rips complex to get an approximation in terms of the persistent homology [33].

6.2.9. Computing well diagrams for vector fields on $\mathbb{R}^n$

Participant: Frédéric Chazal.

In collaboration with Primoz Skraba (Lubiana Univ.), Amit Patel (Rutgers Univ.)

Using topological degree theory, we present and prove correctness of a fast algorithm for computing the well diagram, a quantitative property, of a vector field on Euclidean space [17].

6.3. Data Structures and Robust Geometric Computation

6.3.1. Straight-line graph drawing on the torus

Participant: Olivier Devillers.

In collaboration with Luca Castelli Aleardi and Éric Fusy (LIX, Palaiseau).

We extend the notion of canonical orderings to cylindric triangulations. This allows us to extend the incremental straight-line drawing algorithm of de Fraysseix et al. to this setting. Our algorithm yields in linear time a crossing-free straight-line drawing of a cylindric triangulation T with n vertices on a regular grid $\mathbb{Z}/w\mathbb{Z} \times [0, h]$, with $w \leq 2n$ and $h \leq n(2d + 1)$, where d is the (graph-) distance between the two boundaries. As a by-product, we can also obtain in linear time a crossing-free straight-line drawing of a toroidal triangulation with n vertices on a periodic regular grid $\mathbb{Z}/w\mathbb{Z} \times \mathbb{Z}/h\mathbb{Z}$, with $w \leq 2n$ and $h \leq 1 + n(2c + 1)$, where c is the length of a shortest non-contractible cycle. Since $c \leq \sqrt{2n}$, the grid area is $O(n^{5/2})$ [24].

6.3.2. Qualitative symbolic perturbation

Participants: Olivier Devillers, Monique Teillaud.

In collaboration with Menelaos Karavelas (University of Crete).

In the literature, the generic way to address degeneracies in computational geometry is the *Symbolic Perturbation* paradigm: the input is made dependent of some parameter ε so that for ε positive and close to zero, the input is close to the original input, while at the same time, in non-degenerate position. A geometric predicate can usually be seen as the sign of some function of the input. In the symbolic perturbation paradigm, if the function evaluates to zero, the input is perturbed by a small positive ε, and the sign of the function evaluated at the perturbed input is used instead.

The usual way of using this approach is what we will call *Algebraic Symbolic Perturbation* framework. When the function to be evaluated is a polynomial of the input, its perturbed version is seen as a polynomial in ε, whose coefficients are polynomials in the input. These coefficients are evaluated by increasing degree in ε until a non-vanishing coefficient is found. The number of these coefficients can be quite large and expressing them in an easily and efficiently computable manner (e.g., factorized) may require quite some work.

We propose to address the handling of geometric degeneracies in a different way, namely by means of what we call the *Qualitative Symbolic Perturbation* framework. We no longer use a single perturbation that must remove all degeneracies, but rather a sequence of perturbations, such that the next perturbation is being used only if the previous ones have not removed the degeneracies. The new perturbation is considered as *symbolically smaller* than the previous ones. This approach allows us to use simple elementary perturbations whose effect can be analyzed and evaluated: (1) by geometric reasoning instead of algebraic development of the predicate polynomial in ε, and (2) independently of a specific algebraic formulation of the predicate.

We apply our framework to predicates used in the computation of Apollonius diagrams in 2D and 3D, as well as the computation of trapezoidal maps of circular arcs [57].

6.3.3. Covering spaces and Delaunay triangulations of the 2D flat torus

Participants: Mikhail Bogdanov, Monique Teillaud.

In collaboration with Gert Vegter (Johan Bernoulli Institute, Groningen University)

A previous algorithm was computing the Delaunay triangulation of the flat torus, by using a 9-sheeted covering space [64]. We propose a modification of the algorithm using only a 8-sheeted covering space, which allows to work with 8 periodic copies of the input points instead of 9. The main interest of our contribution is not only this result, but most of all the method itself: this new construction of covering spaces generalizes to Delaunay triangulations of surfaces of higher genus.

6.3.4. Hyperbolic Delaunay complexes and Voronoi diagrams made practical

Participants: Mikhail Bogdanov, Olivier Devillers, Monique Teillaud.

We study Delaunay complexes and Voronoi diagrams in the Poincaré ball, a confomal model of the hyperbolic space, in any dimension. We elaborate on our earlier work on the space of spheres [65], giving a detailed description of algorithms, and presenting a static and a dynamic variants. All proofs are based on geometric reasoning, they do not resort to any use of the analytic formula of the hyperbolic distance. We also study algebraic and arithmetic issues, observing that only rational computations are needed. This allows for an exact and efficient implementation in 2D. All degenerate cases are handled. The implementation will be submitted to the CGAL editorial board for future integration into the CGAL library [44].

6.3.5. The stability of Delaunay triangulations

Participants: Jean-Daniel Boissonnat, Ramsay Dyer.

In collaboration with Arijit Ghosh (Indian Statistical Institute, Kolkata, India)

We introduce a parametrized notion of genericity for Delaunay triangulations which, in particular, implies that the Delaunay simplices of δ-generic point sets are thick. Equipped with this notion, we study the stability of Delaunay triangulations under perturbations of the metric and of the vertex positions. We quantify the magnitude of the perturbations under which the Delaunay triangulation remains unchanged. We also present an algorithm that takes as input a discrete point set in $\mathbb{R}^m$, and performs a small perturbation that guarantees that the Delaunay triangulation of the resulting perturbed point set has quantifiable stability with respect to the metric and the point positions. There is also a guarantee on the quality of the simplices: they cannot be too flat. The algorithm provides an alternative tool to the weighting or refinement methods to remove poorly shaped simplices in Delaunay triangulations of arbitrary dimension, but in addition it provides a guarantee of stability for the resulting triangulation [21], [47].

6.3.6. Constructing intrinsic Delaunay triangulations of submanifolds

Participants: Jean-Daniel Boissonnat, Ramsay Dyer.

In collaboration with Arijit Ghosh (Indian Statistical Institute, Kolkata, India)

This work is the algorithmic counterpart of our previous paper [21]. We describe an algorithm to construct an intrinsic Delaunay triangulation of a smooth closed submanifold of Euclidean space. We also provide a counterexample to the results announced by Leibon and Letscher on Delaunay triangulations on Riemannian manifolds. In general the nerve of the intrinsic Voronoi diagram is not homeomorphic to the manifold. The density of the sample points alone cannot guarantee the existence of a Delaunay triangulation. To circumvent this issue, we use results established in our companion paper on the stability of Delaunay triangulations on δ-generic point sets. We establish sampling criteria which ensure that the intrinsic Delaunay complex coincides with the restricted Delaunay complex and also with the recently introduced tangential Delaunay complex. The algorithm generates a point set that meets the required criteria while the tangential complex is being constructed. In this way the computation of geodesic distances is avoided, the runtime is only linearly dependent on the ambient dimension, and the Delaunay complexes are guaranteed to be triangulations of the manifold [46].

6.3.7. Equating the witness and restricted Delaunay complexes

Participants: Jean-Daniel Boissonnat, Ramsay Dyer, Steve Oudot.

In collaboration with Arijit Ghosh (Indian Statistical Institute, Kolkata, India)

It is a well-known fact that the restricted Delaunay and witness complexes may differ when the landmark and witness sets are located on submanifolds of Rd of dimension 3 or more. Currently, the only known way of overcoming this issue consists of building some crude superset of the witness complex, and applying a greedy sliver exudation technique on this superset. Unfortunately, the construction time of the superset depends exponentially on the ambient dimension, which makes the witness complex based approach to manifold reconstruction impractical. This work provides an analysis of the reasons why the restricted Delaunay and witness complexes fail to include each other. From this, a new set of conditions naturally arises under which the two complexes are equal [37].

6.3.8. Simpler complexity analysis of random geometric structures

Participants: Olivier Devillers, Marc Glisse.

In collaboration with Xavier Goaoc (EPI VEGAS*).*

Average-case analysis of data-structures or algorithms is commonly used in computational geometry when the, more classical, worst-case analysis is deemed overly pessimistic. Since these analyses are often intricate, the models of random geometric data that can be handled are often simplistic and far from "realistic inputs". We present a new simple scheme for the analysis of geometric structures. While this scheme only produces results up to a polylog factor, it is much simpler to apply than the classical techniques and therefore succeeds in analyzing new input distributions related to smoothed complexity analysis.

We illustrate our method on two classical structures: convex hulls and Delaunay triangulations. Specifically, we give short and elementary proofs of the classical results that n points uniformly distributed in a ball in $\mathbb{R}^d$ have a convex hull and a Delaunay triangulation of respective expected complexities $\widetilde{\Theta}(n^{\frac{d-1}{d+1}})$ and $\widetilde{\Theta}(n)$. We then prove that if we start with n points well-spread on a sphere, e.g. an (ϵ,κ)-sample of that sphere, and perturb that sample by moving each point randomly and uniformly within distance at most δ of its initial position, then the expected complexity of the convex hull of the resulting point set is $\widetilde{\Theta}\left((\sqrt{n})^{1-\frac{1}{d}}\left(\frac{1}{\sqrt[4]{\delta}}\right)^{d-\frac{1}{d}}\right)$ [55].

6.3.9. Analysis of cone vertex walk in Poisson Delaunay triangulation

Participants: Olivier Devillers, Ross Hemsley.

In collaboration with Nicolas Broutin (EPI RAP*).*

Walking strategies are a standard tool for point location in a triangulation of size n. Although often claimed to be $\Theta(\sqrt{n})$ under random distribution hypotheses, this conjecture has only been formally proved by Devroye, Lemaire, and Moreau [*Comp Geom–Theor Appl*, vol. 29, 2004], in the case of the so called *straight walk* which has the very specific property that deciding whether a given (Delaunay) triangle belongs to the walk may be determined without looking at the other sites. We analyze a different walking strategy that follows vertex neighbour relations to move towards the query. We call this walk *cone vertex walk*. We prove that cone vertex walk visits $\Theta(\sqrt{n})$ vertices and can be constructed in $\Theta(\sqrt{n})$ time. We provide explicit bounds on the hidden constants [50].

6.3.10. The monotonicity of f-vectors of random polytopes

Participants: Olivier Devillers, Marc Glisse.

In collaboration with Xavier Goaoc and Guillaume Moroz (EPI VEGAS*) and Matthias Reitzner (Universität Osnabrück, Germany).*

Let K be a compact convex body in $\mathbb{R}^d$, let K_n be the convex hull of n points chosen uniformly and independently in K, and let $f_i(K_n)$ denote the number of i-dimensional faces of K_n.

We show that for planar convex sets, $E[f_0(K_n)]$ is increasing in n. In dimension $d \geq 3$, we prove that if $\lim_{n\to\infty} \frac{E[f_{d-1}(K_n)]}{An^c} = 1$ for some constants A and $c > 0$ then the function $n \mapsto E[f_{d-1}(K_n)]$ is increasing for n large enough. In particular, the number of facets of the convex hull of n random points distributed uniformly and independently in a smooth compact convex body is asymptotically increasing. Our proof relies on a *random sampling* argument [57].

6.3.11. Efficient Monte Carlo sampler for detecting parametric objects in large scenes

Participants: Florent Lafarge, Yannick Verdie.

Point processes have demonstrated efficiency and competitiveness when addressing object recognition problems in vision. However, simulating these mathematical models is a difficult task, especially on large scenes. Existing samplers suffer from average performances in terms of computation time and stability. We propose a new sampling procedure based on a Monte Carlo formalism. Our algorithm exploits Markovian properties of point processes to perform the sampling in parallel. This procedure is embedded into a data-driven mechanism such that the points are non-uniformly distributed in the scene. The performances of the sampler are analyzed through a set of experiments on various object recognition problems from large scenes, and through comparisons to the existing algorithms [35], [63].

6.4. Applications

6.4.1. Creating large-scale city models from 3D-point clouds: a robust approach with hybrid representation

Participant: Florent Lafarge.

We present a novel and robust method for modeling cities from 3D-point data. Our algorithm provides a more complete description than existing approaches by reconstructing simultaneously buildings, trees and topologically complex grounds. A major contribution of our work is the original way of modeling buildings which guarantees a high generalization level while having semantized and compact representations. Geometric 3D-primitives such as planes, cylinders, spheres or cones describe regular roof sections, and are combined with mesh-patches that represent irregular roof components. The various urban components interact through a non-convex energy minimization problem in which they are propagated under arrangement constraints over a planimetric map. Our approach is experimentally validated on complex buildings and large urban scenes of millions of points, and is compared to state-of-the-art methods [19].

6.4.2. The sticky geometry of the cosmic web

Participant: Monique Teillaud.

In collaboration with Johan Hidding, Rien van de Weygaert, Bernard J.T. Jones (Kapteyn Institute, Groningen University) and Gert Vegter (Johan Bernoulli Institute, Groningen University)

We highlight the application of Computational Geometry to our understanding of the formation and dynamics of the Cosmic Web. The emergence of this intricate and pervasive weblike structure of the Universe on Megaparsec scales can be approximated by a well-known equation from fluid mechanics, the Burgers' equation. The solution to this equation can be obtained from a geometrical formalism. We have extended and improved this method by invoking weighted Delaunay and Voronoi tessellations. The duality between these tessellations finds a remarkable and profound reflection in the description of physical systems in Eulerian and Lagrangian terms [28].

GRACE Team

5. New Results

5.1. Modular curves

F. Morain has been studying the theory and practice of modular curves associated with Weber's invariants. His paper ... is accepted for publication in *Acta Arithmetica*.

5.2. Computing discrete logarithms using codes

D. Augot and F. Morain have been working on the practical application of Reed–Solomon decoding to speed up discrete logarithm computations, following the work of Cheng and Wan. This work is available as a preprint [22], and a Magma implementation was written in support of the many experiments needed.

5.3. Interleaved codes and codes over rings

G. Quintin designed a decoding algorithm based on a lifting decoding scheme. He obtained a unique decoding algorithm with quasi-linear complexity in all parameters for Reed–Solomon codes over Galois rings. Using erasures, he improved the decoding radius with the same complexity. He then applied these techniques to interleaved linear codes over a finite field, and obtained a decoding algorithm that can recover more errors than half the minimum distance. This work has been presented at IEEE ISIT 2012 (Boston, USA).

5.4. Number fields codes

J.-F. Biasse and G. Quintin described an algorithm for list decoding algebraic number field codes in polynomial time in [24]. This is the first explicit procedure for decoding number field codes, whose construction were previously described by Lenstra [33] and Guruswami [32]. They rely on a new algorithm for computing the Hermite normal form of the basis of an $\mathcal{O}_K$-module due to Biasse and Fieker [31], where $\mathcal{O}_K$ is the ring of integers of a number field K. This work has been presented at IEEE ISIT 2012 (Boston, USA).

5.5. Point counting using p-adic methods

C. Gonçalvès designed a new algorithm to compute Zeta functions of cyclic covers of the projective line. This algorithm is a generalisation of the one for superelliptic curves provided by P. Gaudry and N. Gürel and has the same complexity. Moreover, optimal bounds for the precision have been proved. An alternative basis for computations has been studied and the resulting algorithm is faster, even if the asymptotic complexity is the same.

5.6. Codes and Cartier Operator

A. Couvreur proposed a new construction of codes from algebraic curves over a finite field in [25]. This class of codes is a natural geometric generalisation of classical Goppa codes. In particular, the nice equalities "$\Gamma(L, g^{q-1}) = \Gamma(L, g^q)$" satisfied by classical Goppa codes (for instance, see [30]) extend naturally to this larger class of codes. This article is to appear in *Proceeding of the American Mathematical Society*.

5.7. Quantum Codes

A. Couvreur, N. Delfosse and G. Zémor studied a construction of quantum LDPC codes proposed by McKay, Mitchison and Shokrollahi in a draft. This construction involves Cayley graphs of $GF(2)^n$. A general lower bound for the minimum distance of such codes has been found. In addition, a family of such codes whose parameters are proved to be $[[n, O(\sqrt{n}, O(\sqrt{N}))]]$ is exhibited. Notice that up to now, no construction of quantum LDPC codes is known to have a minimum distance better that $O(\sqrt{n})$. The obtained parameters beat many well–known constructions. This work has been presented at IEEE ISIT 2011 (St Petersburg, Russia), and a long version paper [26] has been submitted to an international journal.

5.8. Code-based McEliece like cryptology

A. Couvreur is working with P. Gaborit, V. Gauthier, A. Otmani, and J.-P. Tillich on distinguisher-based attacks on cryptosystems based on Generalised Reed–Solomon codes. Using the particular structure of the square of an evaluation code, they have been able to break some variants of McEliece's cryptosystem using Generalised Reed–Solomon codes, such as Wieschebrink's variant [34]. An article is in preparation.

5.9. Cyclic Codes

A. Zeh is working with A. Wachter-Zeh (University of Ulm and Institut de Recherche de Mathématique de Rennes) and Sergey Bezzateev (St. Petersburg State University of Aerospace Instrumentation) on a new bound for the minimum distance of q-ary cyclic codes [19], [18]. The connection to the BCH bound and the Hartmann–Tzeng (HT) bound was formulated explicitly. Furthermore, the bound was refined for several families of cyclic codes. We defined syndromes and formulated a Key Equation that allows an efficient decoding up to our bound with the Extended Euclidean Algorithm. It turned out that low-rate cyclic codes with small minimum distances are useful for our approach.

5.10. Iterative List Decoding

A. Zeh is working with J. S. R. Nielsen (Department of Mathematics, DTU) on an iterative list decoding algorithm for generalized Reed–Solomon codes. The method is parametrizable and allows variants of the usual list decoding approach. An article is in preparation.

LFANT Project-Team

6. New Results

6.1. Class groups and other invariants of number fields

Participants: Karim Belabas, Jean-François Biasse, Jean-Paul Cerri, Pierre Lezowski.

P. Lezowski extended J.-P. Cerri's algorithm, which was restricted to totally real number fields, to decide whether a generic number field is norm-Euclidean. His procedure allowed to find principal and non norm-Euclidean number fields of various signatures and degrees up to 8, but also to give further insight about the norm-Euclideanity of some cyclotomic fields. Besides, many new examples of generalised Euclidean and 2-stage Euclidean number fields were obtained. The article [31] will appear in *Mathematics of Computation*.

In another direction, norm-Euclidean ideal classes have been studied. They generalise the notion of norm-Euclideanity to non principal number fields. Very few such number fields were known before. A modification of the algorithm provided many new examples and allowed to complete the study of pure cubic fields equipped with a norm-Euclidean ideal class [15].

J.-F.Biasse has determined a class of number fields for which the ideal class group, the regulator, and a system of fundamental units of the maximal order can be computed in subexponential time $L(1/3, O(1))$ (whereas the best previously known algorithms have complexity $L(1/2, O(1))$). This class of number fields is analogous to the class of curves described in [10]. The article [22] has been submitted to *Mathematics of Computation*.

Assuming the GRH, Bach proved that one can calculate the residue of the Dedekind zeta function of a number field K from the knowledge of the splitting of primes $p < X$, with an error bounded explicitly in terms of X and the field discriminant. This is a crucial ingredient in all algorithms used to compute class groups and unit groups in subexponential time (under GRH). Using Weil's explicit formula, K. Belabas improved on Bach's bound, speeding up by a sizable constant factor this part of the class group algorithm. The article has been submitted to *Mathematics of Computation*.

6.2. Number and function fields

Participants: Athanasios Angelakis, Karim Belabas, Pieter Rozenhart.

In joint work with R. Scheidler and M. Jacobson, P. Rozenhart has generalized Belabas's algorithm for tabulating cubic number fields to cubic function fields [17]. This generalization required function field analogues of the Davenport-Heilbronn Theorem and of the reduction theory of binary cubic and quadratic forms. As an additional application, they have modified the tabulation algorithm to compute 3-ranks of quadratic function fields by way of a generalisation of a theorem due to Hasse. The algorithm, whose complexity is quasi-linear in the number of reduced binary cubic forms up to some upper bound X, works very well in practice. A follow-up article [35] describes how to use these results to compute 3-ranks of quadratic function fields, in particular yielding examples of unusually high 3-rank.

In 1976, Onabe discovered that, in contrast to the Neukirch–Uchida results that were proved around the same time, a number field K is not completely characterised by its absolute abelian Galois group A_K. The first examples of non-isomorphic K having isomorphic A_K were obtained on the basis of a classification by Kubota of idele class character groups in terms of their infinite families of Ulm invariants, and did not yield a description of A_K. In [21], A. Angelakis and P. Stevenhagen provide a direct "computation" of the profinite group A_K for imaginary-quadratic K, and use it to obtain many different K that all have the same minimal absolute abelian Galois group.

On March 29–April 2, 2010, a meeting was organized by J.-M. Couveignes, D. Bertrand, Ph. Boalch and P. Debes, at the Luminy CIRM (France) on geometric and differential Galois theories, witnessing the close ties these theories have woven in recent years. The volume [18] collects the proceedings of this meeting. The articles gathered in this volume cover the following topics: moduli spaces of connections, differential equations and coverings in finite characteristic, liftings, monodromy groups in their various guises (tempered fundamental group, motivic groups, generalised difference Galois groups), and arithmetic applications.

Using Galois theory of extension rings, J.-M. Couveignes, R. Lercier and T. Ezome have proposed a new pseudo-primality test in [13]. For every positive integer $k \leq logn$, this test achieves the security of k Miller-Rabin tests at the cost of $k^{1/2} + o(1)$ Miller-Rabin tests. The implementation in Magma shows that this test is competitive for primes with a few thousands digits.

6.3. Quaternion algebras

Participants: Jean-Paul Cerri, Pierre Lezowski, Aurel Page.

With J. Chaubert, J.-P. Cerri and P. Lezowski have studied whether some quaternion fields over number fields are Euclidean, that is to say whether they admit a left or right Euclidean order. In particular, they have established the complete list of totally definite and Euclidean quaternion fields over the rationals or a quadratic number field. Moreover, they have proved that every field in this list is in fact norm Euclidean. The proofs are both theoretical and algorithmic. The article [23] will appear in *International Journal of Number Theory*.

Starting with an order in a suitable quaternion algebra over a number field F with exactly one complex place, one can construct discrete subgroups of $\mathrm{PSL}_2(\mathbb{C})$. These groups, called arithmetic Kleinian groups, act properly discontinuously with finite covolume on the hyperbolic 3-space. In [34], A. Page designs an efficient algorithm which computes a fundamental domain and a presentation for such a group. It is a generalization to the dimension 3 of an algorithm of J. Voight's [44] together with a new, nondeterministic, but faster enumeration procedure. A public implementation is available in KLEINIANGROUPS (see 5.8).

6.4. Complex multiplication and modularity

Participants: Jean-Marc Couveignes, Andreas Enge, Nicolas Mascot, Aurel Page, Damien Robert.

The article by D. Lubicz and D. Robert which explains how to compute an isogeny between two abelian varieties given the kernel (but with different levels of theta structures) has been published [16]. The preprint [25] with R. Cosset and D. Robert extends these method to provide an algorithm constructing the corresponding isogeny without changing the level. This give the first algorithm allowing to compute in polynomial time an isogeny between abelian varieties, and a public implementation is available in AVISOGENIES. The drawback of this algorithm is that it needs the geometric points of the kernel. To compute an isogeny of degree ℓ^g over a finite field, working with geometric points requires to take an extension of degree up to $\ell^g - 1$, and the situation is much worse over a number field. Recently, D. Lubicz and D. Robert have explained how to compute the corresponding isogeny given only the equations of the kernel. This gives a quasi-linear algorithm (in the degree ℓ^g of the isogeny) when ℓ is congruent to 1 modulo 4.

With K. Lauter, D. Robert has worked on improving the computation of class polynomials in genus 2 by the CRT method. The main improvements come from using the above isogeny computation, both to find a maximal curve from a curve in the correct isogeny class, and to find all other maximal curves from one. Further improvements are in the endomorphism ring computation to detect if the curve is maximal, a better sieving of the primes used (and a dynamic selection of them), and the use of the CRT over the real quadratic field rather than over $\mathbb{Q}$ for the case of dihedral CM fields to find factors of the class polynomials. These results have been published at the ANTS conference [30].

With C. Ritzenthaler, Damien Robert has shown how to compute explicitly the Serre obstruction for abelian varieties isogenous to a product of three elliptic curves. This allows to find genus 3 curves with many points over a finite field. The corresponding code has been implemented in an (experimental) version of AVISOGENIES.

In [24], H. Cohen studies several methods for the numerical computation of Petersson scalar products. In particular he proves a generalisation of Haberland's formula to any subgroup of finite index G of $\Gamma = \mathrm{PSl}_2(Z)$, which gives a fast method to compute these scalar products when a Hecke eigenbasis is not necessarily available.

J.-M. Couveignes and B. Edixhoven explore in [19] the relevance of numerical methods in dealing with higher genus curves and their Jacobians. Fast exponentiation is crucial in this context as a stable substitute to Newton's method and analytic continuation. Arakelov theory provides the necessary complexity estimates.

With Reynald Lercier, J.-M. Couveignes has given in [26] a quasi-linear time randomised algorithm that on input a finite field $\mathbb{F}_q$ with q elements and a positive integer d outputs a degree d irreducible polynomial in $\mathbb{F}_q[x]$. The running time is $d^{1+o(1)} \times (\log q)^{5+o(1)}$ elementary operations. The $o(1)$ in $d^{1+o(1)}$ is a function of d that tends to zero when d tends to infinity. And the $o(1)$ in $(\log q)^{5+o(1)}$ is a function of q that tends to zero when q tends to infinity. The fastest previously known algorithm for this purpose was quadratic in the degree. The algorithm relies on the geometry of elliptic curves over finite fields (complex multiplication) and on a recent algorithm by Kedlaya and Umans for fast composition of polynomials.

In [32], N. Mascot shows how to compute modular Galois representations associated with a newform f and the coefficients of f modulo a small prime ℓ. To this end, he designs a practical variant of the complex approximation method presented in the book edited by B. Edixhoven and J.-M. Couveignes [8]. Its efficiency stems from several new ingredients. For instance, he uses fast exponentiation in the modular Jacobian instead of analytic continuations, which greatly reduces the need to compute abelian integrals, since most of the computation handles divisors. Also, he introduces an efficient way to compute arithmetically well-behaved functions on Jacobians. He illustrates the method on the newform Δ, and manages to compute for the first time the associated faithful representation modulo ℓ and the values modulo ℓ of Ramanujan's τ function at huge primes for $\ell \in \{11, 13, 17, 19\}$. In particular, he gets rid of the sign ambiguity stemming from the use of a non-faithful representation as in J. Bosman's work.

A. Enge and R. Schertz determine in [29] under which conditions singular values of multiple η-quotients of square-free level, not necessarily prime to 6, yield class invariants, that is, algebraic numbers in ring class fields of imaginary-quadratic number fields. It turns out that the singular values lie in subfields of the ring class fields of index $2^{k'-1}$ when $k' \geq 2$ primes dividing the level are ramified in the imaginary-quadratic field, which leads to faster computations of elliptic curves with prescribed complex multiplication. The result is generalised to singular values of modular functions on $X_0^+(p)$ for p prime and ramified.

With F. Morain, A. Enge has determined exhaustively under which conditions "generalised Weber functions", that is, simple quotients of η functions of not necessarily prime transformation level and not necessarily of genus 1, yield class invariants [28]. The result is a new infinite family of generators for ring class fields, usable to determine complex multiplication curves. We examine in detail which lower powers of the functions are applicable, thus saving a factor of up to 12 in the size of the class polynomials, and describe the cases in which the polynomials have integral rational instead of integral quadratic coefficients.

6.5. Elliptic curve cryptology

Participants: Jean-Marc Couveignes, Andreas Enge, Damien Robert.

With J.-G. Kammerer, J.-M. Couveignes has given in [14] an appropriate geometric method for studying and classifying encodings into elliptic curves in a cryptographic context. Such encodings were first proposed by Icart in 2009, and later on by Farashahi, Kammerer, Lercier, and Renault. But it was a little bit disappointing to see that it was no more than an application of Tartaglia's result without any geometrical explanations for the existence of such "parameterisations" of elliptic curves. Couveignes and Kammerer have filled this gap by giving exactly what can be expected from geometry: a clear explanation. Moreover, they unify all the recent "parameterisations" of elliptic curves under the same geometric point of view. The approach described in this article uses dual curves with some results coming from intersection theory. The main originality of this work is that these geometrical tools are employed to explain symbolic computations used in cryptography, that is, encoding on elliptic curves.

The survey [20], to be published in the *Handbook of Finite Fields*, presents the state of the art of the use of elliptic curves in cryptography.

6.6. Pairings

Participants: Andreas Enge, Damien Robert, Jérôme Milan.

In [27], A. Enge gives an elementary and self-contained introduction to pairings on elliptic curves over finite fields. For the first time in the literature, the three different definitions of the Weil pairing are stated correctly and proved to be equivalent using Weil reciprocity. Pairings with shorter loops, such as the ate, ate$_i$, R-ate and optimal pairings, together with their twisted variants, are presented with proofs of their bilinearity and non-degeneracy. Finally, different types of pairings are reviewed in a cryptographic context. The article can be seen as an update chapter to [42].

POLSYS Project-Team

6. New Results

6.1. The complexity of solving quadratic boolean systems is better than exhaustive search

A fundamental problem in computer science is to find all the common zeroes of m quadratic polynomials in n unknowns over $\mathbb{F}_2$. The cryptanalysis of several modern ciphers reduces to this problem. Up to now, the best complexity bound was reached by an exhaustive search in $4\log_2 n\, 2^n$ operations. In [4], we give an algorithm that reduces the problem to a combination of exhaustive search and sparse linear algebra. This algorithm has several variants depending on the method used for the linear algebra step. Under precise algebraic assumptions on the input system, we show in [4] that the deterministic variant of our algorithm has complexity bounded by $O(2^{0.841n})$ when $m = n$, while a probabilistic variant of the Las Vegas type has expected complexity $O(2^{0.792n})$. Experiments on random systems show that the algebraic assumptions are satisfied with probability very close to 1. We also give a rough estimate for the actual threshold between our method and exhaustive search, which is as low as 200, and thus very relevant for cryptographic applications.

6.2. Improving the Complexity of Index Calculus Algorithms in Elliptic Curves over Binary Fields

In [25], we study the index calculus method that was first introduced by Semaev for solving the ECDLP and later developed by Gaudry and Diem. In particular, we focus on the step which consists in decomposing points of the curve with respect to an appropriately chosen factor basis. This part can be nicely reformulated as a purely algebraic problem consisting in finding solutions to a multivariate polynomial. Our main contribution is the identification of particular structures inherent to such polynomial systems and a dedicated method for tackling this problem. We solve it by means of Gröbner basis techniques and analyze its complexity using the multi-homogeneous structure of the equations. We emphasize that the complexity obtained in the paper is very conservative in comparison to experimental results. We hope the new ideas provided here may lead to efficient index calculus based methods for solving ECDLP in theory and practice.

6.3. On the relation between the MXL family of algorithms and Gröbner basis algorithms

The computation of Gröbner bases remains one of the most powerful methods for tackling the Polynomial System Solving (PoSSo) problem. The most efficient known algorithms reduce the Gröbner basis computation to Gaussian eliminations on several matrices. However, several degrees of freedom are available to generate these matrices. It is well known that the particular strategies used can drastically affect the efficiency of the computations. In this work, we investigate a recently-proposed strategy, the so-called "Mutant strategy", on which a new family of algorithms is based (MXL, MXL2 and MXL3). By studying and describing the algorithms based on Gröbner basis concepts, we demonstrate in [3] that the Mutant strategy can be understood to be equivalent to the classical Normal Selection Strategy currently used in Gröbner basis algorithms. Furthermore, we show that the "partial enlargement" technique can be understood as a strategy for restricting the number of S-polynomials considered in an iteration of the F4 Gröbner basis algorithm, while the new termination criterion used in MXL3 does not lead to termination at a lower degree than the classical Gebauer–Möller installation of Buchberger's criteria. We claim that our results map all novel concepts from the MXL family of algorithms to their well-known Gröbner basis equivalents. Using previous results that had shown the relation between the original XL algorithm and F4, we conclude that the MXL family of algorithms can be fundamentally reduced to redundant variants of F4.

6.4. On the Complexity of the BKW Algorithm on LWE

In [35], we present a study of the complexity of the Blum-Kalai-Wasserman (BKW) algorithm when applied to the Learning with Errors (LWE) problem, by providing refined estimates for the data and computational effort requirements for solving concrete instances of the LWE problem. We apply this refined analysis to suggested parameters for various LWE-based cryptographic schemes from the literature and, as a result, provide new upper bounds for the concrete hardness of these LWE-based schemes.

6.5. On the Complexity of the Arora-Ge algorithm against LWE

Arora & Ge recently showed that solving LWE can be reduced to solve a high-degree non-linear system of equations. They used a linearization to solve the systems. We investigate in [34] the possibility of using Gröbner bases to improve Arora & Ge approach.

6.6. On enumeration of polynomial equivalence classes and their application to MPKC

The Isomorphism of Polynomials (IP) is one of the most fundamental problems in multivariate public key cryptography (MPKC). In [8], we introduce a new framework to study the counting problem associated to IP. Namely, we present tools of finite geometry allowing to investigate the counting problem associated to IP. Precisely, we focus on enumerating or estimating the number of isomorphism equivalence classes of homogeneous quadratic polynomial systems. These problems are equivalent to finding the scale of the key space of a multivariate cryptosystem and the total number of different multivariate cryptographic schemes respectively, which might impact the security and the potential capability of MPKC. We also consider their applications in the analysis of a specific multivariate public key cryptosystem. Our results not only answer how many cryptographic schemes can be derived from monomials and how big the key space is for a fixed scheme, but also show that quite many HFE cryptosystems are equivalent to a Matsumoto-Imai scheme.

6.7. Cryptanalysis of HFE, Multi-HFE and Variants for Odd and Even Characteristic

In [5], we investigate the security of HFE and Multi-HFE schemes as well as their minus and embedding variants. Multi-HFE is a generalization of the well-known HFE schemes. The idea is to use a multivariate quadratic system - instead of a univariate polynomial in HFE - over an extension field as a private key. According to the authors, this should make the classical direct algebraic (message-recovery) attack proposed by Faugère and Joux on HFE no longer efficient against Multi-HFE. We consider here the hardness of the key-recovery in Multi-HFE and its variants, but also in HFE (both for odd and even characteristic). We first improve and generalize the basic key recovery proposed by Kipnis and Shamir on HFE. To do so, we express this attack as matrix/vector operations. In one hand, this permits to improve the basic Kipnis-Shamir (KS) attack on HFE. On the other hand, this allows to generalize the attack on Multi-HFE. Due to its structure, we prove that a Multi-HFE scheme has much more equivalent keys than a basic HFE. This induces a structural weakness which can be exploited to adapt the KS attack against classical modifiers of multivariate schemes such as minus and embedding. Along the way, we discovered that the KS attack as initially described cannot be applied against HFE in characteristic 2. We have then strongly revised KS in characteristic 2 to make it work. In all cases, the cost of our attacks is related to the complexity of solving MinRank. Thanks to recent complexity results on this problem, we prove that our attack is polynomial in the degree of the extension field for all possible practical settings used in HFE and Multi-HFE. This makes then Multi-HFE less secure than basic HFE for equally-sized keys. As a proof of concept, we have been able to practically break the most conservative proposed parameters of multi-HFE in few days (256 bits security broken in 9 days).

6.8. Solving Polynomial Systems over Finite Fields: Improved Analysis of the Hybrid Approach

The Polynomial System Solving (PoSSo) problem is a fundamental NP-Hard problem in computer algebra. Among others, PoSSo have applications in area such as coding theory and cryptology. Typically, the security of multivariate public-key schemes (MPKC) such as the UOV cryptosystem of Kipnis, Shamir and Patarin is directly related to the hardness of PoSSo over finite fields. The goal of [22] is to further understand the influence of finite fields on the hardness of PoSSo. To this end, we consider the so-called *hybrid approach*. This is a polynomial system solving method dedicated to finite fields proposed by Bettale, Faugère and Perret (Journal of Mathematical Cryptography, 2009). The idea is to combine exhaustive search with Gröbner bases. The efficiency of the hybrid approach is related to the choice of a trade-off between the two methods. We propose here an improved complexity analysis dedicated to quadratic systems. Whilst the principle of the hybrid approach is simple, its careful analysis leads to rather surprising and somehow unexpected results. We prove that the optimal trade-off (i.e. number of variables to be fixed) allowing to minimize the complexity is achieved by fixing a number of variables proportional to the number of variables of the system considered, denoted n. Under some natural algebraic assumption, we show that the asymptotic complexity of the hybrid approach is $2^{(3.31-3.62\ \log_2(q)^{-1})\,n}$, where q is the size of the field (under the condition in particular that $\log(q) \ll n$). This is to date, the best complexity for solving PoSSo over finite fields (when $q > 2$). We have been able to quantify the gain provided by the hybrid approach compared to a direct Gröbner basis method. For quadratic systems, we show (assuming a natural algebraic assumption) that this gain is exponential in the number of variables. Asymptotically, the gain is $2^{1.49\,n}$ when both n and q grow to infinity and $\log(q)$.

6.9. Efficient Arithmetic in Successive Algebraic Extension Fields Using Symmetries

In [15] we present new results for efficient arithmetic operations in a number field $\mathbb{K}$ represented by successive extensions. These results are based on multi-modular and evaluation–interpolation techniques. We show how to use intrinsic symmetries in order to increase the efficiency of these techniques. Applications to splitting fields of univariate polynomials are presented.

6.10. Algebraic Crypanalysis with Side Channels Information

In [6] and [24] (see also the PhD thesis of C. Goyet [1]), we present new cryptanalyses of symmetric and asymmetric cryptosystems (e.g. AES and ECDSA). These analyses share the same fundamental hypotheses that some information are provided to the attacker by some oracle. In a practical point of view, such an oracle can be represented as a partial side channel attack realized in a first step (e.g. SPA, Fault attacks). The second step of the attack uses algorithms from computer algebra (e.g. Gröbner basis computation, LLL) in order to retrieve the secret key.

6.11. Worst case complexity of the Continued Fraction (CF) algorithm.

In [16] we consider the problem of isolating the real roots of a square-free polynomial with integer coefficients using the classic variant of the continued fraction algorithm (CF), introduced by Akritas. We compute a lower bound on the positive real roots of univariate polynomials using exponential search. This allows us to derive a worst case bound of $\widetilde{O}(d^4\tau^2)$ for isolating the real roots of a polynomial with integer coefficients using the *classic variant of CF*, where d is the degree of the polynomial and τ the maximum bitsize of its coefficients. This improves the previous bound of Sharma by a factor of d^3 and matches the bound derived by Mehlhorn and Ray for another variant of CF which is combined with subdivision; it also matches the worst case bound of the classical subdivision-based solvers STURM, DESCARTES, and BERNSTEIN.

6.12. Local Generic Position for Root Isolation of Zero-dimensional Triangular Polynomial Systems.

In [30] we present an algorithm based on local generic position (LGP) to isolate the complex or real roots and their multiplicities of a zero-dimensional triangular polynomial system. The Boolean complexity of the algorithm for computing the real roots is single exponential: $\widetilde{O}_B(N^{n^2})$, where $N = \max\{d, \tau\}$, d and τ, is the degree and the maximum coefficient bitsize of the polynomials, respectively, and n is the number of variables.

6.13. Univariate Real Root Isolation in Multiple Extension Fields

In [31] we present algorithmic, complexity and implementation results for the problem of isolating the real roots of a univariate polynomial in $K[x] \in L[y]$, where $L = \mathbb{Q}(\alpha\_1, \cdots, \alpha\_\ell)$ is an algebraic extension of the rational numbers. Our bounds are single exponential in ℓ and match the ones presented for the case $\ell = 1$. We consider two approaches. The first, indirect approach, using multivariate resultants, computes a univariate polynomial with integer coefficients, among the real roots of which are the real roots of B_α. The Boolean complexity of this approach is $\widetilde{O}_B(N^{4\ell+4})$, where N is the maximum of the degrees and the coefficient bitsize of the involved polynomials. The second, direct approach, tries to solve the polynomial directly, without reducing the problem to a univariate one. We present an algorithm that generalizes Sturm algorithm from the univariate case, and modified versions of well known solvers that are either numerical or based on Descartes' rule of sign. We achieve a Boolean complexity of $\widetilde{O}_B(\min\{N^{4\ell+7}, N^{2\ell^2+6}\})$ and $\widetilde{O}_B(N^{2\ell+4})$, respectively. We implemented the algorithms in C as part of the core library of Mathematica and we illustrate their efficiency over various data sets.

6.14. Mixed volume and distance geometry techniques for counting Euclidean embeddings of rigid graphs.

A graph G is called generically minimally rigid in $\mathbb{R}^d$ if, for any choice of sufficiently generic edge lengths, it can be embedded in $\mathbb{R}^d$ in a finite number of distinct ways, modulo rigid transformations. In [37] we deal with the problem of determining tight bounds on the number of such embeddings, as a function of the number of vertices. The study of rigid graphs is motivated by numerous applications, mostly in robotics, bioinformatics, and architecture. We capture embeddability by polynomial systems with suitable structure, so that their mixed volume, which bounds the number of common roots, yields interesting upper bounds on the number of embeddings. We explore different polynomial formulations so as to reduce the corresponding mixed volume, namely by introducing new variables that remove certain spurious roots, and by applying the theory of distance geometry. We focus on $\mathbb{R}^2$ and $\mathbb{R}^3$, where Laman graphs and 1-skeleta of convex simplicial polyhedra, respectively, admit inductive Henneberg constructions. Our implementation yields upper bounds for $n \leq 10$ in $\mathbb{R}^2$ and $\mathbb{R}^3$, which reduce the existing gaps and lead to tight bounds for $n \leq 7$ in both $\mathbb{R}^2$ and $\mathbb{R}^3$; in particular, we describe the recent settlement of the case of Laman graphs with 7 vertices. We also establish the first lower bound in $\mathbb{R}^3$ of about 2.52^n, where n denotes the number of vertices.

6.15. Variant Quantifier Elimination

In [10], we describe an algorithm (VQE) for a *variant* of the real quantifier elimination problem (QE). The variant problem requires the input to satisfy a certain *extra condition*, and allows the output to be *almost* equivalent to the input. The motivation/rationale for studying such a variant QE problem is that many quantified formulas arising in applications do satisfy the extra conditions. Furthermore, in most applications, it is sufficient that the output formula is almost equivalent to the input formula. The main idea underlying the algorithm is to substitute the repeated projection step of CAD by a single projection without carrying out a parametric existential decision over the reals. We find that the algorithm can tackle important and challenging problems, such as numerical stability analysis of the widely-used MacCormack's scheme. The problem has been practically out of reach for standard QE algorithms in spite of many attempts to tackle it. However the current implementation of VQE can solve it in about 12 hours.

6.16. Global optimization

Let $f_1, \cdots, f_p$ be in $\mathbb{Q}[\mathbf{X}]$, where $\mathbf{X} = (X_1, \cdots, X_n)^t$, that generate a radical ideal and let V be their complex zero-set. Assume that V is smooth and equidimensional. Given $f \in \mathbb{Q}[X]$ bounded below, consider the optimization problem of computing $f^\star = \inf_{x \in V \cap \mathbb{R}^n} f(x)$. For $\mathbf{A} \in GL_n(\mathbb{C})$, we denote by $f^{\mathbf{A}}$ the polynomial $f(\mathbf{AX})$ and by $V^{\mathbf{A}}$ the complex zero-set of $f_1^{\mathbf{A}}, ..., f_p^{\mathbf{A}}$. In [9], we construct families of polynomials $\mathsf{M}_0^{\mathbf{A}}, ..., \mathsf{M}_d^{\mathbf{A}}$ in $\mathbb{Q}[\mathbf{X}]$: each $\mathsf{M}_i^{\mathbf{A}}$ is related to the section of a linear subspace with the critical locus of a linear projection. We prove that there exists a non-empty Zariski-open set $O \subset GL_n(\mathbb{C})$ such that for all $\mathbf{A} \in O \cap GL_n(\mathbb{Q})$, $f(x)$ is non-negative for all $x \in V \cap \mathbb{R}^n$ if, and only if, $f^{\mathbf{A}}$ can be expressed as a sum of squares of polynomials on the truncated variety generated by the ideal $\langle \mathsf{M}_i^{\mathbf{A}} \rangle$, for $0 \leq i \leq d$. Hence, we can obtain algebraic certificates for lower bounds on $f^\star$ using semidefinite programs. Some numerical experiments are given. We also discuss how to decrease the number of polynomials in $\mathsf{M}_i^{\mathbf{A}}$.

6.17. Gröbner bases and critical points

We consider the problem of computing critical points of the restriction of a polynomial map to an algebraic variety. This is of first importance since the global minimum of such a map is reached at a critical point. Thus, these points appear naturally in non-convex polynomial optimization which occurs in a wide range of scientific applications (control theory, chemistry, economics,etc.). Critical points also play a central role in recent algorithms of effective real algebraic geometry. Experimentally, it has been observed that Gröbner basis algorithms are efficient to compute such points. Therefore, recent software based on the so-called Critical Point Method are built on Gröbner bases engines. Let $f_1, ..., f_p$ be polynomials in $\mathbb{Q}[x_1, ..., x_n]$ of degree D, $V \subset \mathbb{C}^n$ be their complex variety and π_1 be the projection map $(x_1, ..., x_n) \mapsto x_1$. The critical points of the restriction of π_1 to V are defined by the vanishing of $f_1, ..., f_p$ and some maximal minors of the Jacobian matrix Indus associated to $f_1, ..., f_p$. Such a system is algebraically structured: the ideal it generates is the sum of a determinantal ideal and the ideal generated by $f_1, ..., f_p$. In [26], we provide the first complexity estimates on the computation of Gröbner bases of such systems defining critical points. We prove that under genericity assumptions on $f_1, ..., f_p$, the complexity is polynomial in the generic number of critical points, i.e. $D^p(D-1)^{n-p} \begin{pmatrix} n-1 \\ p-1 \end{pmatrix}$ More particularly, in the quadratic case $D = 2$, the complexity of such a Gröbner basis computation is polynomial in the number of variables n and exponential in p. We also give experimental evidence supporting these theoretical results.

SECRET Project-Team

5. New Results

5.1. Symmetric cryptosystems

Participants: Christina Boura, Baudoin Collard, Anne Canteaut, Pascale Charpin, Gohar Kyureghyan, María Naya-Plasencia, Joëlle Roué, Valentin Suder.

From outside, it might appear that symmetric techniques become obsolete after the invention of public-key cryptography in the mid 1970's. However, they are still widely used because they are the only ones that can achieve some major features as high-speed or low-cost encryption, fast authentication, and efficient hashing. Today, we find symmetric algorithms in GSM mobile phones, in credit cards, in WLAN connections. Symmetric cryptology is a very active research area which is stimulated by a pressing industrial demand for low-cost implementations (in terms of power consumption, gate complexity...). These extremely restricting implementation requirements are crucial when designing secure symmetric primitives and they might be at the origin of some weaknesses. Actually, these constraints seem quite incompatible with the rather complex mathematical tools needed for constructing a provably secure system.

The specificity of our research work is that it considers all aspects of the field, from the practical ones (new attacks, concrete specifications of new systems) to the most theoretical ones (study of the algebraic structure of underlying mathematical objects, definition of optimal objects). But, our purpose is to study these aspects not separately but as several sides of the same domain. Our approach mainly relies on the idea that, in order to guarantee a provable resistance to the known attacks and to achieve extremely good performance, a symmetric cipher must use very particular building blocks, whose algebraic structures may introduce unintended weaknesses. Our research work captures this conflict for all families of symmetric ciphers. It includes new attacks and the search for new building blocks which ensure both a high resistance to the known attacks and a low implementation cost. This work, which combines cryptanalysis and the theoretical study of discrete mathematical objects, is essential to progress in the formal analysis of the security of symmetric systems.

In this context, the very important challenges are the designs of low-cost ciphers and of secure hash functions. Most teams in the research community are actually working on the design and on the analysis (cryptanalysis and optimisation of the performance) of such primitives.

5.1.1. Hash functions.

Following the recent attacks against almost all existing hash functions (MD5, SHA-0, SHA-1...), we have initiated a research work in this area, especially within the Saphir-2 ANR project and with several PhD theses. Our work on hash functions is two-fold: we have designed two new hash functions, named FSB and Shabal, which have been submitted to the SHA-3 competition, and we have investigated the security of several hash functions, including the previous standards (SHA-0, SHA-1...) and some other SHA-3 candidates.

Recent results:

- Upper bounds on the degree of an iterated permutation from the degree of the inverse of the inner transformation; this result has been applied both to hash functions and to block ciphers. Most notably, this work leads to the best (theoretical) analysis of the hash function Keccak, which has been selected for the new SHA-3 standard [12], [22], [9].
- Side-channel attacks on two SHA-3 candidates, Skein and Grøstl, when they are used with HMAC, and counter-measures [23], [50].
- Indifferentiability results for a broadened mode of operation including the modes based on block ciphers, and modes based on un-keyed functions [51].

5.1.2. Block ciphers.

Even if the security of the current block cipher standard, AES, is not threaten when it is used in a classical context, there is still a need for the design of improved attacks, and for the determination of design criteria which guarantee that the existing attacks do not apply. This notably requires a deep understanding of all previously proposed attacks. Moreover, there is a high demand from the industry of lightweight block ciphers for some constrained environments. Several such algorithms have been proposed in the last few years and their security should be carefully analysed. Most of our work in this area is related to an ANR Project named BLOC.

Recent results:

- Algebraic analysis of some recent lightweight block ciphers, including LED and Piccolo [24].
- Analysis of the security of the lightweight block cipher mCRYPTON [56].
- Design of a new block cipher, named PRINCE, with a very low-latency, leading to instantaneous encryption (i.e., within one clock cycle) with a very competitive chip area [21], [49].
- Analysis of the differential properties of the AES Superbox [58].
- Study of the significance of the related-key and known-key models for block ciphers [48].

5.1.3. Stream ciphers.

The project-team has been involved in the international project eSTREAM, which aimed at recommending some secure stream ciphers.

Recent results:

- Generalisation of several improvements of the so-called correlation attacks against stream ciphers and study of their complexities [13].
- Study of the bias of parity-check relations for combination generators used in stream ciphers [14].

5.1.4. Cryptographic properties and construction of appropriate building blocks.

The construction of building blocks which guarantee a high resistance to the known attacks is a major topic within our project-team, for stream ciphers, block ciphers and hash functions. The use of such optimal objects actually leads to some mathematical structures which may be the origin of new attacks. This work involves fundamental aspects related to discrete mathematics, cryptanalysis and implementation aspects. Actually, characterising the structures of the building blocks which are optimal regarding to some attacks is very important for finding appropriate constructions and also for determining whether the underlying structure induces some weaknesses or not.

For these reasons, we have investigated several families of filtering functions and of S-boxes which are well-suited for their cryptographic properties or for their implementation characteristics. For instance, bent functions, which are the Boolean functions which achieve the highest possible nonlinearity, have been extensively studied in order to provide some elements for a classification, or to adapt these functions to practical cryptographic constructions. We have also been interested in functions with a low differential uniformity (*e.g.*, APN functions), which are the S-boxes ensuring an (almost) optimal resistance to differential cryptanalysis.

Recent results:

- Study of the algebraic properties (e.g. the algebraic degree) of the inverses of APN power permutations [26].
- Study of the planarity of some mappings, including products of linearized polynomials [25], [16].
- Definition of a new criterion for Sboxes and link with some recent algebraic attacks on the hash functuion Hamsi [29], [9].
- Survey of PN and APN mappings [42].

5.2. Code-based cryptography

Participants: Grégory Landais, Rafael Misoczki, Nicolas Sendrier, Dimitrios Simos, Jean-Pierre Tillich.

Most popular public-key cryptographic schemes rely either on the factorization problem (RSA, Rabin), or on the discrete logarithm problem (Diffie-Hellman, El Gamal, DSA). These systems have evolved and today instead of the classical groups $(\mathbf{Z}/n\mathbf{Z})$ we may use groups on elliptic curves. They allow a shorter block and key size for the same level of security. An intensive effort of the research community has been and is still being conducted to investigate the main aspects of these systems: implementation, theoretical and practical security. It must be noted that these systems all rely on algorithmic number theory. As they are used in most, if not all, applications of public-key cryptography today (and it will probably remain so in the near future), cryptographic applications are thus vulnerable to a single breakthrough in algorithmics or in hardware (a quantum computer can break all those scheme).

Diversity is a way to dilute that risk, and it is the duty of the cryptographic research community to prepare and propose alternatives to the number theoretic based systems. The most serious tracks today are lattice-based cryptography (NTRU,...), multivariate cryptography (HFE,...) and code-based cryptography (McEliece encryption scheme,...). All these alternatives are referred to as *post-quantum cryptosystems*, since they rely on difficult algorithmic problems which would not be solved by the coming-up of the quantum computer.

The code-based primitives have been investigated in details within the project-team. The first cryptosystem based on error-correcting codes was a public-key encryption scheme proposed by McEliece in 1978; a dual variant was proposed in 1986 by Niederreiter. We proposed the first (and only) digital signature scheme in 2001. Those systems enjoy very interesting features (fast encryption/decryption, short signature, good security reduction) but also have their drawbacks (large public key, encryption overhead, expensive signature generation). Some of the main issues in this field are

- security analysis , implementation and practicality of existing solutions,
- reducing the key size, *e.g.*, by using rank metric instead of Hamming metric, or by using particular families of codes,
- address new functionalities, like hashing or symmetric encryption.

Recent results:

- A new variant of McEliece using Moderate Density Parity Check (MDPC) codes [55];
- An optimized software implementation of the code-based digital signature scheme CFS [27];
- An attack on a homomorphic encryption scheme [53];
- An attack on a variant of the McEliece cryptosystem based on Reed-Solomon codes [54].

5.3. Error-correcting codes and applications

Participants: Mamdouh Abbara, Marion Bellard, Denise Maurice, Nicolas Sendrier, Jean-Christophe Sibel, Jean-Pierre Tillich, Audrey Tixier.

We mainly investigate two new application domains for decoding algorithms: reverse engineering of communication systems, and quantum error correcting codes for which we have shown that some of them can be decoded successfully with iterative decoding algorithms.

5.3.1. Quantum codes.

The knowledge we have acquired in iterative decoding techniques has also led to study whether or not the very same techniques could also be used to decode quantum codes. Part of the old ACI project “RQ” in which we were involved and the new ANR project “COCQ” are about this topic. It is worth noticing that protecting quantum information from external noise is an issue of paramount importance for building a quantum computer. It also worthwhile to notice that all quantum error-correcting code schemes proposed up to now suffer from the very same problem that the first (classical) error-correcting codes had: there are constructions of good quantum codes, but for the best of them it is not known how to decode them in polynomial time. Our approach for overcoming this problem has been to study whether or not the family of turbo-codes and LDPC codes (and the associated iterative decoding algorithms) have a quantum counterpart.

Recent results:

- Construction of quantum LDPC codes obtained by transforming a quantum CSS LDPC code into a code over a larger alphabet which improves substantially the performances under iterative decoding [18];
- Construction of spatially coupled quantum LDPC codes which performs well under iterative decoding almost up to the coherent capacity of the quantum channel [19].

5.3.2. Reverse engineering of communication systems.

To evaluate the quality of a cryptographic algorithm, it is usually assumed that its specifications are public, as, in accordance with Kerckhoffs principle [1], it would be dangerous to rely, even partially, on the fact that the adversary does not know those specifications. However, this fundamental rule does not mean that the specifications are known to the attacker. In practice, before mounting a cryptanalysis, it is necessary to strip off the data. This reverse engineering process is often subtle, even when the data formatting is not concealed on purpose. A typical case is interception; some raw data, not necessarily encrypted, is observed out of a noisy channel. To access the information, the whole communication system has first to be disassembled and every constituent reconstructed. Our activity within this domain, whose first aim is to establish the scientific and technical foundations of a discipline which does not exist yet at an academic level, has been supported by some industrial contracts driven by the DGA and the French Ministry for Defense.

Recent results:

- Recontruction of the constellation labeling (i.e. used in the modulator of a communication system) in presence of error and when the underlying code is convolutional [20].

[1] Kerckhoffs stated that principle in a paper entitled *La Cryptographie militaire*, published in 1883.

VEGAS Project-Team

5. New Results

5.1. Classical computational geometry

5.1.1. Complexity analysis of random geometric structures made simpler

Average-case analysis of data-structures or algorithms is commonly used in computational geometry when the more classical worst-case analysis is deemed overly pessimistic. Since these analyses are often intricate, the models of random geometric data that can be handled are often simplistic and far from "realistic inputs".

In a joint work with Olivier Devillers and Marc Glisse (Inria GEOMETRICA) [20], we presented a new simple scheme for the analysis of geometric structures. While this scheme only produces results up to a polylog factor, it is much simpler to apply than the classical techniques and therefore succeeds in analyzing new input distributions related to smoothed complexity analysis. We illustrated our method on two classical structures: convex hulls and Delaunay triangulations. Specifically, we gave short and elementary proofs of the classical results that n points uniformly distributed in a ball in R^d have a convex hull and a Delaunay triangulation of respective expected complexities $\widetilde{\Theta}(n^{((d+1)/(d-1))})$ and $\widetilde{\Theta}(n)$. We then prove that if we start with n points well-spread on a sphere, e.g. an (ϵ, κ)-sample of that sphere, and perturb that sample by moving each point randomly and uniformly within distance at most δ of its initial position, then the expected complexity of the convex hull of the resulting point set is $\widetilde{\Theta}(\sqrt(n)^{(1-1/d)}\delta^{-(d-1)/(4d)})$.

5.1.2. On the monotonicity of the expected number of facets of a random polytope

Let K be a compact convex body in R^d, let K_n be the convex hull of n points chosen uniformly and independently in K, and let $f_i(K_n)$ denote the number of i-dimensional faces of K_n.

In a joint work with Olivier Devillers and Marc Glisse (Inria GEOMETRICA) and Matthias Reitzner (Univ. Osnabruck) [21], we showed that for planar convex sets, $E(f_0(K_n))$ is increasing in n. In dimension $d \geq 3$ we prove that if $lim_{n\to\infty} \frac{E(f_{d-1}(K_n))}{An^c} = 1$ for some constants A and $c > 0$ then the function $E(f_{d-1}(K_n))$ is increasing for n large enough. In particular, the number of facets of the convex hull of n random points distributed uniformly and independently in a smooth compact convex body is asymptotically increasing. Our proof relies on a random sampling argument.

5.1.3. Embedding geometric structures

We continued working this year on the problem of embedding geometric objects on a grid of $\mathbb{R}^3$. Essentially all industrial applications take, as input, models defined with a fixed-precision floating-point arithmetic, typically doubles. As a consequence, geometric objects constructed using exact arithmetic must be embedded on a fixed-precision grid before they can be used as input in other software. More precisely, the problem is, given a geometric object, to find a similar object representable with fixed-precision floating-point arithmetic, where similar means topologically equivalent, close according to some distance function, etc. We are working on the problem of rounding polyhedral subdivisions on a grid of $\mathbb{R}^3$, where the only known method, due to Fortune in 1999, considers a grid whose refinement depends on the combinatorial complexity of the input, which does not solve the problem at hand. This project is joint work with Olivier Devillers (Inria Geometrica) and William Lenhart (Williams College, USA) who was in sabbatical in our team in 2012.

5.2. Non-linear computational geometry

5.2.1. Geometry of robotic mechanisms

Parallel manipulators are a family of mechanisms, the geometry of which is difficult to compute in general. The use of algebraic methods allowed us to describe precisely the geometry of the configurations of different specific parallel manipulators, in collaboration with researchers from the IRCCyN laboratory in Nantes.

More precisely, moving a parallel robot toward specific parametric values can break it. A challenge is to describe this set of singularities. This was adressed for a planar mechanism with three degrees of freedom in [16] and a spatial mechanism with six degrees of freedom in [12].

Then, a more challenging question arises naturally. Given a familly of mechanisms parametrized by some construction variables, is it possible to find a mechanism that has no singularities? A method based on Gröbner bases was proposed in [17] for a specific family of planar parallel robot with two degrees of freedom.

5.2.2. Solving bivariate systems and topology of algebraic curves

In the context of our algorithm Isotop for computing the topology of algebraic curves [28], we study the bit complexity of solving a system of two bivariate polynomials of total degree d with integer coefficients of bitsize τ. We focus on the problem of computing a Rational Univariate Representation (RUR) of the solutions, that is, roughly speaking, a univariate polynomial and two rational functions which map the roots of the polynomial to the two coordinates of the solutions of the system.

We work on an algorithm for computing RURs with worst-case bit complexity in $O(d^8 + d^7\tau + d^5\tau^2)$ (where polylogarithmic factors are omitted). In addition, we show that certified approximations of the real solutions can be computed from this representation with $O(d^8 + d^7\tau)$ bit operations. It should be stressed that our algorithm is deterministic and that it makes no genericity assumption.

When $\tau \in O(d^2)$, this complexity decreases by a factor d^2 the best known upper bound for computing Rational Univariate Representations of such systems and it matches the recent best known complexity (Emeliyanenko and Sagraloff, 2012) for "only" computing certified approximations of the solutions. This shows, in particular, that computing RURs of bivariate systems is in a similar class of (known) complexity as computing certified approximations of *one* of the variables of its real solutions.

This work is on-going and is done in collaboration with Fabrice Rouillier (Inria Ouragan).

5.3. Combinatorics and combinatorial geometry

5.3.1. Multinerves and Helly numbers of acyclic families

The nerve of a family of sets is a simplicial complex that records the intersection pattern of its subfamilies. Nerves are widely used in computational geometry and topology, because the nerve theorem guarantees that the nerve of a family of geometric objects has the same topology as the union of the objects, if they form a good cover.

In a joint work with Éric Colin de Verdière (CNRS-ENS) and Grégory Ginot (Univ. Paris 6) we relaxed the good cover assumption to the case where each subfamily intersects in a disjoint union of possibly several homology cells, and we proved a generalization of the nerve theorem in this framework, using spectral sequences from algebraic topology. We then deduced a new topological Helly-type theorem that unifies previous results of Amenta, Kalai and Meshulam, and Matoušek. This Helly-type theorem is used to (re)prove, in a unified way, bounds on transversal Helly numbers in geometric transversal theory.

This work was presented at SoCG 2012 [18], where it received one of the two "best paper" awards.

5.3.2. Set systems and families of permutations with small traces

In a joint work with Otfried Cheong (KAIST, South Korea) and Cyril Nicaud (Univ. Marne-La-Vallée), we studied two problems of the following flavor: how large can a family of combinatorial objects defined on a finite set be if its number of distinct "projections" on any small subset is bounded? We first consider set systems, where the "projections" is the standard notion of trace, and for which we generalized Sauer's Lemma on the size of set systems with bounded VC-dimension. We then studied families of permutations, where the "projections" corresponds to the notion of containment used in the study of permutations with excluded patterns, and for which we delineated the main growth rates ensured by projection conditions. One of our motivations for considering these questions is the "geometric permutation problem" in geometric transversal theory, a question that has been open for two decades.

This work was published in the European Journal of Combinatorics [13].

5.3.3. Simplifying inclusion-exclusion formulas

Let $F = \{F_1, F_2, ..., F_n\}$ be a family of n sets on a ground set X, such as a family of balls in R^d. For every finite measure μ on X, such that the sets of F are measurable, the classical inclusion-exclusion formula asserts that $\mu(F_1 \cup F_2 \cup \bullet\bullet\bullet \cup F_n) = \sum_{I:\varnothing \neq I \subseteq [n]} (-1)^{|I|+1} \mu(\cap_{i \in I} F_i)$; that is, the measure of the union is expressed using measures of various intersections. The number of terms in this formula is exponential in n, and a significant amount of research, originating in applied areas, has been devoted to constructing simpler formulas for particular families F.

In a joint work with Jiří Matoušek, Pavel Paták, Zuzana Safernová and Martin Tancer (Charles Univ., Prague) [22] we provided the apparently first upper bound valid for an arbitrary F: we showed that every system F of n sets with m nonempty fields in the Venn diagram admits an inclusion-exclusion formula with $m^{O((\log n)^2)}$ terms and with ± 1 coefficients, and that such a formula can be computed in $m^{O((\log n)^2)}$ expected time. We also constructed systems of n sets on n points for which every valid inclusion-exclusion formula has the sum of absolute values of the coefficients at least $\Omega(n^{3/2})$.

ALF Project-Team

6. New Results

6.1. Processor Architecture within the ERC DAL project

Participants: Pierre Michaud, Nathanaël Prémillieu, Luis Germán Garcia Morales, Bharath Narasimha Swamy, Sylvain Collange, André Seznec, Arthur Pérais, Surya Narayanan, Sajith Kalathingal, Kamil Kedzierski.

Processor, cache, locality, memory hierarchy, branch prediction, multicore, power, temperature

Multicore processors have now become mainstream for both general-purpose and embedded computing. Instead of working on improving the architecture of the next generation multicore, with the DAL project, we deliberately anticipate the next few generations of multicores. While multicores featuring 1000s of cores might become feasible around 2020, there are strong indications that sequential programming style will continue to be dominant. Even future mainstream parallel applications will exhibit large sequential sections. Amdahl's law indicates that high performance on these sequential sections is needed to enable overall high performance on the whole application. On many (most) applications, the effective performance of future computer systems using a 1000-core processor chip will significantly depend on their performance on both sequential code sections and single threads.

We envision that, around 2020, the processor chips will feature a few complex cores and many (may be 1000's) simpler, more silicon and power effective cores.

In the DAL research project, http://www.irisa.fr/alf/dal, we explore the microarchitecture techniques that will be needed to enable high performance on such heterogeneous processor chips. Very high performance will be required on both sequential sections, —legacy sequential codes, sequential sections of parallel applications—, and critical threads on parallel applications, —e.g. the main thread controlling the application. Our research focuses essentially on enhancing single processes performance.

6.1.1. Microarchitecture exploration of control flow reconvergence

Participants: Nathanaël Prémillieu, André Seznec.

After continuous progress over the past 15 years [14], [13], the accuracy of branch predictors seems to be reaching a plateau. Other techniques to limit control dependency impact are needed. Control flow reconvergence is an interesting property of programs. After a multi-option control-flow instruction (i.e. either a conditional branch or an indirect jump including returns), all the possible paths merge at a given program point: the reconvergence point.

Superscalar processors rely on aggressive branch prediction, out-of-order execution and instruction level parallelism for achieving high performance. Therefore, on a superscalar core, the overall speculative execution after the mispredicted branch is cancelled, leading to a substantial waste of potential performance. However, deep pipelines and out-of-order execution induce that, when a branch misprediction is resolved, instructions following the reconvergence point have already been fetched, decoded and sometimes executed. While some of this executed work has to be cancelled since data dependencies exist, cancelling the control independent work is a waste of resources and performance. We have proposed a new hardware mechanism called SYRANT, SYmmetric Resource Allocation on Not-taken and Taken paths, addressing control flow reconvergence at a reasonable cost. Moreover, as a side contribution of this research we have shown that, for a modest hardware cost, the outcomes of the branches executed on the wrong paths can be used to guide branch prediction on the correct path [17].

As a follower work, we are now focusing on exploiting control flow reconvergence in the special case of predication. When the target ISA has predicated instruction, it is possible to transform control dependencies into data dependencies. This process is called if-conversion. As a result, the two paths of a conditional branch is merge into one path. Hence exploiting the principles developed in SYRANT is much easier than for a standard ISA.

6.1.2. Memory controller

Participant: André Seznec.

The memory controller has become one of the performance enablers of a computer system. Its impact is even higher on multicores than it was on uniprocessor systems. We propose the sErvice Value Aware memory scheduler (EVA) to enhance memory usage. EVA builds on two concepts, the request weight and the per-thread traffic light. For a read request on memory, the request weight is an evaluation of the work allowed by the request. Per-thread traffic lights are used to track whether or not in a given situation it is worth to service requests from a thread, e.g. if a given thread is blocked by refreshing on a rank then it is not worth to serve requests from the same thread on another rank. The EVA scheduler bases its scheduling decision on a service value which is heuristically computed using the request weight and per-thread traffic lights. Our EVA scheduler implementation relies on several hardware mechanisms, a request weight estimator, per-thread traffic estimators and a next row predictor. Using these components, our EVA scheduler estimates scores to issue scheduling decisions. EVA was shown to perform efficiently and fairly compared with previous proposed memory schedulers [21]

6.1.3. Performance and power models for heterogeneous muticores

Participants: Kamil Kedzierski, André Seznec.

In the DAL project, we expect architectures to be a combination of many simple cores for parallel execution and sequential accelerators [8] built on top of complex cores for ILP intensive tasks. For evaluating these architectures, we need performance and power models. We design a parallel manycore simulator, built with pthread implementation. Such an approach allows us to maintain flexibility and scalability: our goal is to scale well both when we vary the number of cores used to perform simulation, and as we vary the number of cores being simulated. Our implementation also allows to configure each core independently for the heterogeneous architectures. Preliminary results show that the simulator uses with very small memory footprint, which is crucial for the manycore studies with number of cores constantly increasing.

A new power management approach is needed for these future manycore processors that employ both sequential accelerators and simple cores. This is due to the fact that the frequency at which a given core operates is highly correlated with the cores' size (and thus a task that the core performs). Therefore, we built Dynamic Voltage Frequency Scaling model for the on-chip voltage regulator (VR) case, as we believe that future architectures will incorporate VRs on chip.

6.1.4. Designing supercores

Participants: Pierre Michaud, Luis Germán García Morales, André Seznec.

In the framework of the DAL project, we study super-cores that could achieve very high clock frequency and a high instruction per cycle rate (IPC). The current objective is to explore the design space of possible configurations for the microarchitecture that are suitable in terms of performance, area and power for the super-core. In particular, we focus on the back-end of the microarchitecture. A way to increase the IPC is to allow the core processing more instructions simultaneously e.g. increasing the issue width. This can be done for example by replicating the functional units (FU) inside the core. However keeping the same frequency could become very challenging. Clustering of FUs is a technique that helps designers to overcome this problem, even though other problems might appear e.g. IPC loss compared to an ideal monolithic back-end due to inter-cluster delays. We have started exploring different cluster schemes and instruction steering policies with the purpose of having a wide-issue clustered microarchitecture with a high IPC, a high frequency and the problem of inter-cluster delay minimized.

6.1.5. Helper threads

Participants: Bharath Narasimha Swamy, André Seznec.

Improving sequential performance will be key to both performance on single threaded codes and scalability on parallel codes. Complex out-of-order execution processors that aggressively exploit instruction level parallelism are the obvious design direction to improve sequential performance. However, ability of these complex cores to deliver performance will be undermined by performance degrading events such as branch mis-predictions and cache misses that limit the achievable instruction throughput. As an alternative to the monolithic complex core approach, we propose to improve sequential performance on emerging heterogeneous many core architectures by harnessing (unutilized) additional cores to work as helper cores for the sequential code. Helper cores can be employed to mitigate the impact of performance degrading events and boost sequential performance, for example by prefetching data for the sequential code ahead of time.

We are currently pursuing two directions to utilize helper cores. (1) We explore the use of helper cores to emulate prefetch algorithms in software. We will adapt and extend existing prefetch mechanisms for use on the helper cores and evaluate mechanisms to utilize both compute and cache resources on the helper cores to prefetch for the main thread. We intend to target delinquent load/store instructions that cause most of the cache misses and prefetch data ahead of time, possibly even before the hardware prefetchers on the main core. (2) We explore the use of helper cores to execute pre-computation code and generate prefetch requests for the main thread. Pre-computation code is constructed from the main thread and targets to capture the data access behavior of the main thread, particularly for irregular data access patterns in control-flow dominated code. We will explore algorithms to generate pre-computation code and evaluate mechanisms for communication and synchronization between the main thread and the helper cores, specifically in the context of a heterogenous many core architecture.

6.1.6. What makes parallel code sections and sequential code sections different?

Participants: Surya Natarajan, André Seznec.

In few years from now, single die processor components will feature many cores. They can be symmetric/asymmetric or homogeneous/heterogeneous cores. The utilization of these cores depends on the application and the programming model used. We have initiated a study on understanding the difference in nature between the parallel and sequential code sections in parallel applications. Initial experiments show that instruction mix of the serial and parallel parts are different. For example, contribution of the conditional branches are dominant in serial part and data transfer instructions are dominant in the parallel part. By experimentation, we infer that the conditional branch prediction in serial part needs a bigger branch predictor compared to the parallel part. Later, we would like to define the hardware mechanisms that are needed for cost effective execution of parallel sections; cost-effective meaning silicon and energy effective since parallelism can be leveraged.

On the other hand, the shared memory model has critical sections in the parallel sections, which makes the parallel sections sequential at times. We will try to characterize the nature of these sequential code sections and particularly identify their potential bottlenecks. The objective is to address the performance bottlenecks on sequential sections through new microarchitecture and/or compiler mechanisms.

6.1.7. Revisiting Value Prediction

Participants: Arthur Pérais, André Seznec.

Value prediction was proposed in the mid 90's to enhance the performance of high-end microprocessors. The research on Value Prediction techniques almost vanished in the early 2000's as it was more effective to increase the number of cores than to dedicate silicon to Value Prediction. However high end processor chips currently feature 8-16 high-end cores and the technology will allow to implement 50-100 of such cores on a single die in a foreseeable future. Amdahl's law suggests that the performance of most workloads will not scale to that level. Therefore, dedicating more silicon area to value prediction in high-end cores might be considered as worthwhile for future multicores.

We introduce a new value predictor VTAGE harnessing the global branch history [32]. VTAGE directly inherits the structure of the indirect jump predictor ITTAGE[11]. VTAGE is able to predict with a very high accuracy many values that were not correctly predicted by previously proposed predictors, such as the FCM predictor and the stride predictor. Three sources of information can be harnessed by these predictors: the global branch history, the differences of successive values and the local history of values. Moreover we show that the predictor components using these sources of information are all amenable to very high accuracy at the cost of some prediction coverage.

Compared with these previously proposed solutions, VTAGE can accommodate very long prediction latencies. The introduction of VTAGE opens the path to the design of new hybrid predictors. Using SPEC 2006 benchmarks, our study shows that with a large hybrid predictor, in average 55-60 % of the values can be predicted with more than 99.5 % accuracy. Evaluation of effective performance benefit is an on-going work.

6.1.8. Augmenting superscalar architecture for efficient many-thread parallel execution

Participants: Sylvain Collange, Sajith Kalathingal, André Seznec.

Heterogeneous multi-core architectures create many issues for test, design and optimizations. They also necessitate costly data transfer from the complex cores to the simple cores when switching from the parallel to sequential sections and vice-versa. We have initiated research on designing a unique core that efficiently run both sequential and massively parallel sections. It will explore how the architecture of a complex superscalar core has to be modified or enhanced to be able to support the parallel execution of many threads from the same application (10's or even 100's a la GPGPU on a single core). The overall objective is to support both sequential codes and very parallel execution, particularly data parallelism, on the same hardware core.

6.2. Other Architecture Studies

Participants: Damien Hardy, Pierre Michaud, Ricardo Andrés Velásquez, Sylvain Collange, André Seznec, Junjie Lai.

GPU, performance, simulation, vulnerability

6.2.1. Analytical model to estimate the performance vulnerability of caches and predictors to permanent faults

Participant: Damien Hardy.

This research was partially undertaken during Damien Hardy's stay in the Computer Architecture group of the University of Cyprus (January-August 2012).

Technology trends suggest that in tomorrow's computing systems, failures will become a commonplace due to many factors, and the expected probability of failure will increase with scaling. Faults can result in execution errors or simply in performance loss. Although faults can occur anywhere in the processor, the performance implications of a faulty cell vary depending on how the array is used in a processor.

Virtually all previous micro-architectural work aiming to assess the performance implications of permanently faulty cells relies on simulations with random fault-maps, assumes that faulty blocks are disabled, and focuses on architectural arrays such as caches.

These studies are, therefore, limited by the fault-maps they use that may not be representative for the average and distributed performance. Moreover, they are incomplete by ignoring faults in non-architectural arrays, such as predictors, that do not affect correctness but can degrade performance.

In [20], an analytical model is proposed for understanding the implications on performance of permanently faulty cells in caches and predictors. The model for a given program execution, micro-architectural configuration, and probability of cell failure, provides rapidly the *Performance Vulnerability Factor (PVF)*. PVF is a direct measure of the performance degradation due to permanent faults. In particular, the model can determine the expected PVF as well as the PVF probability distribution bounds without using an arbitrary number of random fault-maps.

The model, once derived, can be used to explore processor behavior with different cell probability of failures. This can be helpful to forecast how processor performance may be affected by faults in the future. Additionally, this information can be useful to determine which arrays have significant PVF and make design decisions to reduce their PVF, for example through a protection mechanism, using larger cells, or even by selecting a different array organization.

6.2.2. GPU-inspired throughput architectures

Participant: Sylvain Collange.

This research was partially undertaken while Sylvain Collange was with Universidade Federal de Minas Gerais, Belo Horizonte - Brazil, (January-September 2012).

In an heterogeneous architecture where power is the primary performance constraint, parallel sections of applications need to run on throughput-optimized cores that focus on energy efficiency. The Single-Instruction Multiple Thread (SIMT) execution model introduced for Graphics Processing Units (GPUs) provides inspiration to design such future energy-efficient throughput architectures. However, the performance of SIMT architectures is vulnerable to control and data flow divergences across threads. It limits its applicability to regular data-parallel applications. We work on making SIMT architectures more efficient, and generalizing the SIMT model to general-purpose architectures.

First, hybrids between multi-thread architectures and SIMT architectures can achieve a tradeoff between energy efficiency and flexibility [35]. Second, the same concepts that benefit GPUs may be applied to vectorize dynamically single-program, multi-thread applications. Indeed, data-parallel multi-thread workloads, such as OpenMP applications, expose parallelism by running many threads executing the same program. These threads may be synchronized to run the same instructions at the same time. SPMD threads also commonly perform the same computation on the same value. We take advantage from these correlations by sharing instructions between threads. It promises to save energy and frees processing resources on multi-threaded cores [26].

Besides architecture-level improvements, the efficiency of SIMT architectures can be improved through compiler-level code optimization. By maintaining a large number of threads in flight (in the order of tens of thousands), GPUs suffer from high cache contention as the local working set of each thread increases. This raises challenges as memory accesses are costly in terms of energy. Divergence analysis is a compiler pass that identifies similarities in the control flow and data flow of concurrent threads. In particular, it detects program variables that are affine functions of the thread identifier. Register allocation can benefit from divergence analysis to unify affine variables across SIMT threads and re-materialize them when needed. It reduces the volume of register spills, relieving pressure on the memory system [28].

6.2.3. Behavioral application-dependent superscalar core modeling

Participants: Ricardo Andrés Velásquez, Pierre Michaud, André Seznec.

Behavioral superscalar core modeling is a possible way to trade accuracy for processor simulation speed in situations where the focus of the study is not the core itself but what is outside the core, i.e., the *uncore*. In this modeling approach, a superscalar core is viewed as a black box emitting requests to the uncore at certain times. A behavioral core model can be connected to a cycle-accurate uncore model. Behavioral core models are built from detailed simulations. Once the time to build the model is amortized, significant simulation speedups are achieved.

We have proposed a new method for defining behavioral models for modern superscalar cores. Our method, *behavioral application-dependent superscalar core* (**BADCO**) modeling, requires two traces generated with cycle-accurate simulations to build a model. After the model is built, it can be used for simulating uncores. BADCO predicts the execution time of a thread running on a modern superscalar core with an error typically under 5%. From our experiments, we found that BADCO is qualitatively accurate, being able to predict how performance changes when we change the uncore. The simulation speedups obtained with BADCO are typically greater than 10 [29].

In a later work [33], we have shown that fast approximate microarchitecture models such as BADCO can also be very useful for selecting multiprogrammed workloads for evaluating the throughput of multicore processors. Computer architects usually study multiprogrammed workloads by considering a set of benchmarks and some combinations of these benchmarks. However, there is no standard method for selecting such sample, and different authors have used different methods. The choice of a particular sample impacts the conclusions of a study. Using BADCO, we propose and compare different sampling methods for defining multiprogrammed workloads for computer architecture [33]. We evaluate their effectiveness on a case study, the comparison of several multicore last-level cache replacement policies. We show that random sampling, the simplest method, is robust to define a representative sample of workloads, provided the sample is big enough. We propose a method for estimating the required sample size based on fast approximate simulation. We propose a new method, workload stratification, which is very effective at reducing the sample size in situations where random sampling would require large samples.

6.2.4. Performance Upperbound Analysis of GPU applications

Participants: Junjie Lai, André Seznec.

In the framework of the ANR Cosinus PetaQCD project, we are modeling the demands of high performance scientific applications on hardware. GPUs have become popular and cost-effective hardware platforms. In this context, we have been addressing the gap between theoretical peak performance on GPU and the effective performance [22]. There has been many studies on optimizing specific applications on GPU as well as and also a lot of studies on automatic tuning tools. However, the gap between the effective performance and the maximum theoretical performance is often huge. A tighter performance upperbound of an application is needed in order to evaluate whether further optimization is worth the effort. We designed a new approach to compute the CUDA application's performance upperbound through intrinsic algorithm information coupled with low-level hardware benchmarking. Our analysis [30] allows us to understand which parameters are critical to the performance and therefore to get more insight on the performance result. As an example, we analyzed the performance upperbound of SGEMM (Single-precision General Matrix Multiply) on Fermi and Kepler GPUs. Through this study, we uncover some undocumented features on Kepler GPU architecture. Based on our analysis, our implementations of SGEMM achieve the best performance on Fermi and Kepler GPUs so far (5 % improvement on average).

6.2.5. Multicore throughput metrics

Participant: Pierre Michaud.

Several different metrics have been proposed for quantifying the throughput of multicore processors. There is no clear consensus about which metric should be used. Some studies even use several throughput metrics. We have shown several new results concerning multicore throughput metrics [16]. We have exhibited the relation between single-thread average performance metrics and throughput metrics, emphasizing that throughput metrics inherit the meaning or lack of meaning of the corresponding single-thread metric [16]. In particular, two of the three most frequently used throughput metrics in microarchitecture studies, the weighted speedup and the harmonic mean of speedups, are inconsistent: they do not give equal importance to all benchmarks. We have demonstrated that the weighted speedup favors unfairness. We have shown that the harmonic mean of IPCs, a seldom used throughput metric, is actually consistent and has a physical meaning. We have explained under which conditions the arithmetic mean or the harmonic mean of IPCs can be used as strong indicators of throughput increase.

In a subsequent work [31], we have pointed out a problem with commonly used multiprogram throughput metrics, which is that they are based on the assumption that all the jobs execute for a fixed and equal time. We argue that this assumption is not realistic. We have proposed and characterized some new throughput metrics based on the assumption that jobs execute a fixed and equal quantity of work. We have shown that using such equal-work throughput metric may change the conclusion of a microarchitecture study [31].

6.3. Compiler, vectorization, interpretation

Participants: Erven Rohou, Emmanuel Riou, Arjun Suresh, André Seznec.

The usage of the bytecode-based languages such as Java has been generalized in the past few years. Applications are now very large and are deployed on many different platforms, since they are highly portable. With the new diversity of multicore platforms, functional, but also performance portability will become the major issue in the next 10 years. Therefore our research effort focuses on efficiently compiling towards bytecodes and on efficiently executing the bytecodes through JIT compilation or through direct interpretations.

6.3.1. Vectorization Technology To Improve Interpreter Performance

Participant: Erven Rohou.

Recent trends in consumer electronics have created a new category of portable, lightweight software applications. Typically, these applications have fast development cycles and short life spans. They run on a wide range of systems and are deployed in a target independent bytecode format over Internet and cellular networks. Their authors are untrusted third-party vendors, and they are executed in secure managed runtimes or virtual machines. Furthermore, due to security policies, these virtual machines are often lacking just-in-time compilers and are reliant on interpreter execution.

The main performance penalty in interpreters arises from instruction dispatch. Each bytecode requires a minimum number of machine instructions to be executed. In this work we introduce a powerful and portable representation that reduces instruction dispatch thanks to vectorization technology. It takes advantage of the vast research in vectorization and its presence in modern compilers. Thanks to a split compilation strategy, our approach exhibits almost no overhead. Complex compiler analyses are performed ahead of time. Their results are encoded on top of the bytecode language, becoming new SIMD IR (i.e., intermediate representation) instructions. The bytecode language remains unmodified, thus this representation is compatible with legacy interpreters.

This approach drastically reduces the number of instructions to interpret and improves execution time. SIMD IR instructions are mapped to hardware SIMD instructions when available, with a substantial improvement. Finally, we finely analyze the impact of our extension on the behavior of the caches and branch predictors.

These results are published in ACM TACO [18], and will be presented at the HiPEAC 2013 conference.

6.3.2. Tiptop

Participant: Erven Rohou.

Hardware performance monitoring counters have recently received a lot of attention. They have been used by diverse communities to understand and improve the quality of computing systems: for example, architects use them to extract application characteristics and propose new hardware mechanisms; compiler writers study how generated code behaves on particular hardware; software developers identify critical regions of their applications and evaluate design choices to select the best performing implementation.

We propose [27] that counters be used by all categories of users, in particular non-experts, and we advocate that a few simple metrics derived from these counters are relevant and useful. For example, a low IPC (number of executed instructions per cycle) indicates that the hardware is not performing at its best; a high cache miss ratio can suggest several causes, such as conflicts between processes in a multicore environment.

We propose tiptop: a new tool, similar to the UNIX top utility, that requires no special privilege and no modification of applications. Tiptop provides more informative estimates of the actual performance than existing UNIX utilities, and better ease of use than current tools based on performance monitoring counters. With several use cases, we have illustrated possible usages of such a tool.

Tiptop has been extended to display any user-defined arithmetic expression based on constants and counter values. A new configuration file lets users defined their default parameters as well as custom expressions.

6.3.3. Code obfuscation and JIT Compilers

Participant: Erven Rohou.

This project proposes to leverage JIT compilation to make software tamper-proof. The idea is to constantly generate different versions of an application, even while it runs, to make reverse engineering hopeless. A strong random number generator will guarantee that generated code is not reproducible, though the functionality is the same. Performance will not be sacrificed thanks to multi-core architectures: the JIT runs on separate cores, overlapping with the execution of the application.

The following directions are investigated:

1. We proposed a "change metric" that evaluates how different each new version of a function differs from the previous one, and hence contributes to the robustness of the system. The metric is based on string matching (such as in bioinformatics).
2. To increase the frequency of code switching, we consider on-stack-replacement. For performance, compilation is performed on a separate thread and pre-copying of the stack state to the new function version, thereby saving switching time.
3. We decompose a thread control-flow graph into many control-flow graphs such that the result of execution would be the same. The control-flow complexity is substantial as there are in the order of $O(n^n)$ possible combinations (where n is the number of threads and compilation units).

This is done in collaboration with the group of Prof. Ahmed El-Mahdy at E-JUST, Alexandria, Egypt.

6.3.4. Dynamic Analysis and Re-Optimization of Executables

Participants: Erven Rohou, Emmanuel Riou.

The objective of the ADT PADRONE beginning in November 2012 is to design and develop a platform for re-optimization of binary executables at run-time. We reviewed available support in hardware (such as performance monitoring unit, trap instructions), and in the Linux operating system (such as the ptrace system call). We started working on the platform, with an initial focus on analysis techniques.

6.3.5. Improving single core execution in the many-core era

Participants: Erven Rohou, André Seznec, Arjun Suresh.

In the framework of the DAL research project, we have initiated compiler research on using available unused resources in multicores to improve the performance of sequential code segments. Helper threads, driven by automated compiler infrastructure, can alleviate potential performance degradation due to resource contention. For example, loop based applications experiencing bad memory locality can be re-optimized by a just-in-time compiler to adjust to actual hardware characteristics.

6.4. WCET estimation

Participants: Damien Hardy, Benjamin Lesage, Hanbing Li, Isabelle Puaut, Erven Rohou, André Seznec.

Predicting the amount of resources required by embedded software is of prime importance for verifying that the system will fulfill its real-time and resource constraints. A particularly important point in hard real-time embedded systems is to predict the Worst-Case Execution Times (WCETs) of tasks, so that it can be proven that tasks temporal constraints (typically, deadlines) will be met. Our research concerns methods for obtaining automatically upper bounds of the execution times of applications on a given hardware. Our focus this year is on (i) multi-core architectures (ii) preemption delay analysis (iii) traceability of flow information in compilers for WCET estimation.

6.4.1. WCET estimation and multi-core systems

6.4.1.1. Predictable shared caches for mixed-criticality real-time systems

Participants: Benjamin Lesage, Isabelle Puaut, André Seznec.

The general adoption of multi-core architectures has raised new opportunities as well as new issues in all application domains. In the context of real-time applications, it has created one major opportunity and one major difficulty. On the one hand, the availability of multiple high performance cores has created the opportunity to mix on the same hardware platform the execution of a complex critical real-time workload and the execution of non-critical applications. On the other hand, for real-time tasks timing deadlines must be met and enforced. Hardware resource sharing inherent to multicores hinders the timing analysis of concurrent tasks. Two different objectives are then pursued: enforcing timing deadlines for real-time tasks and achieving highest possible performance for the non-critical workload.

In this work [23], we suggest a hybrid hardware-based cache partitioning scheme that aims at achieving these two objectives at the same time. Plainly considering inter-task conflicts on shared cache for real-time tasks yields very pessimistic timing estimates. We remove this pessimism by reserving private cache space for real-time tasks. Upon the creation of a real-time task, our scheme reserves a fixed number of cache lines per set for the task. Therefore uniprocessor worst case execution time (WCET) estimation techniques can be used, resulting in tight WCET estimates. Upon the termination of the real-time task, this private cache space is released and made available for all the executed threads including non-critical ones. That is, apart the private spaces reserved for the real-time tasks currently running, the cache space is shared by all tasks running on the processor, i.e. non-critical tasks but also the real-time tasks for their least recently used blocks. Experiments show that the proposed cache scheme allows to both guarantee the schedulability of a set of real-time tasks with tight timing constraints and enable high performance on the non-critical tasks.

6.4.1.2. WCET-oriented cache partitioning for multi-core systems

Participant: Isabelle Puaut.

Multi-core architectures are well suited to fulfill the increasing performance requirements of embedded real-time systems. However, such systems also require the capacity to estimate the timing behavior of their critical components. Interference between tasks, as they occur on standard multi-core micro-architectures due to cache sharing are still difficult to predict accurately. An alternative is to remove these indirect interferences between tasks through partitioning of the shared cache and through the use of partitioned task scheduling.

In this work [19], we have proposed a new algorithm for joint task and cache partitioning in multi-core systems scheduled using non-preemptive Earliest Deadline First policy. The main novelty of the algorithm is to take into account the tasks' period repartition in the task partitioning problem, which is critical in a non-preemptive context. Other task properties such as task cache requirements are also considered to optimize cache partitioning. Experiments show that our algorithm outperforms the state-of-the-art algorithm for tasks and cache partitioning, named IA3 [43], in terms of schedulability, specially when the spectrum of tasks periods is wide.

6.4.2. Preemption delay analysis for floating non-preemptive region scheduling

Participant: Isabelle Puaut.

This is joint work with Stefan M. Petters, Vincent Nélis and José Marinho, ISEP Porto, Portugal.

In real-time systems, there are two distinct trends for scheduling task sets on unicore systems: non-preemptive and preemptive scheduling. Non-preemptive scheduling is obviously not subject to any preemption delays but its schedulability may be quite poor, whereas fully preemptive scheduling is subject to preemption delays, but benefits from a higher flexibility in the scheduling decisions.

The time-delay involved by task preemptions is a major source of pessimism in the analysis of the task Worst-Case Execution Time (WCET) in real-time systems. Cache related preemption delays (CRPD) are the most important ones, and are caused by the preempting tasks that modify the cache; the preempted task then suffers an indirect delay after the preemption to reload the cache with useful information.

Preemptive scheduling policies including non-preemptive regions are a hybrid solution between non-preemptive and fully preemptive scheduling paradigms, which enables to conjugate both worlds benefits. In this work [25], we exploit the connection between the progression of a task in its operations, and the knowledge of the preemption delays as a function of its progression. Thus the pessimism in the preemption delay estimation is reduced, in comparison to state of the art methods, due to the increase in information available in the analysis. The method proposed in [25] was later improved in [24], to extract more information on the code and further decrease the CRPD estimation.

6.4.3. Traceability of flow information for WCET estimation

Participants: Hanbing Li, Isabelle Puaut, Erven Rohou.

Control-flow information is mandatory for WCET estimation, to guarantee that programs terminate (e.g. provision of bounds for the number of loop iterations) but also to obtain tight estimates (e.g. identification of infeasible or mutually exclusive paths). Such flow information is expressed though annotations, that may be calculated automatically by program/model analysis, or provided manually.

The objective of this work is to address the challenging issue of the mapping and transformation of the flow information from high level down to machine code. In a first step, we will consider only the issue of conveying information through the compilation flow, without any optimization. Then, we will study the impact of optimizations on the traceability of annotations.

This research started in October 2012 and is part of the ANR W-SEPT project.

CAIRN Project-Team

6. New Results

6.1. Reconfigurable Architecture Design

6.1.1. Reconfiguration Controller

Participants: Robin Bonamy, Daniel Chillet, Sébastien Pillement.

Dynamically reconfigurable architectures, which can offer high performance, are increasingly used in different domains. Unfortunately, lots of applications cannot benefit from this new paradigm due to large timing overhead. Even for partial reconfiguration, modifying a small region of an FPGA takes few *ms* using the 14.5MB/s IP from Xilinx based on an embedded micro blaze processor. To cope with this problem by increasing performance, we have developed an ultra-fast power-aware reconfiguration controller (UPaRC) to boost the reconfiguration throughput up to 1.433 GB/s. UPaRC cannot only enhance the system performance, but also auto-adapt to various performance and consumption conditions. This could enlarge the range of supported applications and can optimize power-timing trade-off of reconfiguration phase for each selected application during run-time. The energy-efficiency of UPaRC over state-of-the-art reconfiguration controllers is up to 45 times more efficient [66].

6.1.2. Low-Power Reconfigurable Arithmetic Operators

Participants: Vivek D. Tovinakere, Olivier Sentieys, Arnaud Tisserand.

Arithmetic operators with fixed input data sizes are a source of unnecessary power consumption when data of lower precision have to be processed for significant amount of time. Configuring the arithmetic operator for lower precision when adequate and suppressing standby power in unused logic gates of the circuit can provide the benefit of reduced power consumption. In this work a logic clustering approach to partition arithmetic circuits as a function of reconfigurable input data widths is presented. Unused clusters at a specific precision are power-gated to achieve aggressive leakage power reduction that is a source of significant power consumption in nanoscale technologies. Application of this method to two types of 32-bit adders, reconfigurable to four precisions of data in 65nm CMOS technology shows a possible reduction in power consumption by a factor of 8 to 13 with an area overhead of 15% and 9.2% respectively. The variation of energy savings with respect to standby time of unused logic and frequency of precision adaptation was also analyzed.

6.1.3. Ultra-Low-Power Reconfigurable Controllers

Participants: Vivek D. Tovinakere, Olivier Sentieys, Steven Derrien.

Most digital systems use controllers based on a finite state machine (FSM) and datapath model. For specific control tasks, this model gives an energy efficient ASIC-like implementation compared to a microcontroller. This is especially true when the controller is required to execute a pre-specified task flow graph consisting of several basic tasks in applications like wireless sensor network (WSN) nodes. Previously design flows have been proposed to generate FSMs along with datapaths for tasks specified at a high level of abstraction and hence combine them with a scheduler to realize the overall controller. The generated controller was found to be efficient compared to its microcontroller counterpart by over two orders of magnitude in energy per operation metric, but a significant limitation of such controllers is the lack of flexibility. In this work, flexible controllers based on reconfigurable FSMs are considered at an expense of hardware area. Scalable architectures for reconfigurable FSMs based on lookup tables (LUTs) whose complexity may be parameterized by a high level specification of number of states, primary inputs and outputs of an FSM are proposed. Power gating as a low power technique is used to achieve aggressive leakage power reduction by shutting-off power to unused parts of logic at any given time. It is well known that in nanoscale CMOS circuits, the increase in static power density as a cost far exceeds the impact of area due to increased logic integration. The feedback and feedforward structures of a FSM are exploited to reduce programmable interconnections - a key issue in reconfigurable logic like FPGAs. Power estimation results show good performance of proposed architectures on different metrics when compared with other solutions in the design space of controllers for WSN nodes.

6.1.4. Models for Dynamically Reconfigurable Systems

6.1.4.1. Power Models

Participants: Robin Bonamy, Daniel Chillet, Olivier Sentieys.

Including a reconfigurable area in a heterogeneous system-on-chip is considered as an interesting solution to reduce area and increase performance. But the key challenge in the context of embedded systems is currently the power budget of the system, and the designer needs some early estimations of the power consumption of its system. Power estimation for reconfigurable systems is a difficult problem because several parameters need to be taken into account to define an accurate model.

In this work, we considered dynamic reconfiguration that makes possible to partially reconfigure a specific part of the circuit while the rest of the system is running. This technique has two main effects on power consumption. First, thanks to the area sharing ability, the global size of the device can be reduced and the static (leakage) power consumption can thus be also reduced. Secondly, it is possible to delete the configuration of a part of the device which reduces the dynamic power consumption when a task is no longer used. We have defined several models of power consumption for the dynamic reconfiguration on a Virtex 5 board and a first model of the power consumption of the reconfiguration. This model shows that the power consumption not only depends on the bitstream file size but also on the content of the reconfiguration region. Finally three models of the partial and dynamic reconfiguration with different complexities/accuracy tradeoffs are extracted [52].

6.1.4.2. High-Level Modeling of Reconfigurable Architectures

Participants: Robin Bonamy, Daniel Chillet.

To model complex multiprocessor SoCs, the Architecture Analysis & Design Language (AADL) has been adopted. We have proposed an extension of AADL towards reconfigurable systems to support power consumption and dynamic reconfiguration modeling. As different power/energy/time/cost tradeoffs can be achieved for a given application, we proposed to represent as Pareto frontiers the set of values of power/energy vs. execution time or cost to model the execution of an application on the reconfigurable system. These Pareto frontiers are computed from analysis functions which extract and combine component characteristics from AADL models. These functions, developed in OCL (Object Constraint Language), are well suited for design space exploration and they can be used to extract the energy/power properties from the model to compute and to verify user's constraints.

To complete these levels of description, we started the development of techniques for constraint verifications. These developments are based on the OCL language, which allows one to extract characteristics on the AADL model, compute mathematical expressions and finally verify mathematical constraints. These verifications have been developed for power and energy consumption, they include static and dynamic power estimation, the power consumption during the dynamic reconfiguration process and the reconfiguration speed. They handle all energy/power parameters related to reconfigurable architectures for an energy estimation of a complete application and heterogeneous system. We currently work on the link between the design space exploration explained in the previous section and the AADL models developed in collaboration with the LEAT laboratory, and to be included in the Open-People Platform [27], [54], [76], [71].

6.1.5. Fault-Tolerant Reconfigurable Architectures

Participants: Sébastien Pillement, Manh Pham, Stanislaw Piestrak [Univ. Metz].

In terms of complex systems implementation, reconfigurable FPGAs circuits are now part of the mainstream thanks to their flexibility, performance and high number of integrated resources. FPGAs enter new fields of applications such as aeronautics, military, automotive or confined control thanks to their ability to be remotely updated. However, these fields of applications correspond to harsh environments (cosmic radiation, ionizing, electromagnetic noise) and with high fault-tolerance requirements. We proposed a complete framework to design reconfigurable architecture supporting fault-tolerance mitigation schemes. The proposed framework enables simulation, validation of mitigation operations, but also the scaling of architecture resources. The

proposed model was validated thanks to a physical implementation of the fault-tolerant reconfigurable platform. Results have shown the effectiveness of the framework [39] and confirmed the potential of dynamically reconfigurable architectures for supporting fault-tolerance in embedded systems.

6.1.6. Low-Power Architectures

6.1.6.1. Wakeup Time and Wakeup Energy Estimation in Power-Gated Logic Clusters

Participants: Olivier Sentieys, Vivek D. Tovinakere.

Run-time power gating for aggressive leakage reduction has brought into focus the cost of mode transition overheads due to frequent switching between sleep and active modes of circuit operation. In order to design circuits for effective power gating, logic circuits must be characterized for overheads they present during mode transitions. We have proposed a method to determine steady-state virtual-supply voltage in active mode and hence present a model for virtual-supply voltage in terms of basic circuit parameters. Further, we derived expressions for the estimation of two mode transition overheads: wakeup time and wakeup energy for a power-gated logic cluster using the proposed model. Experimental results of application of the model to ISCAS85 benchmark circuits show that wakeup time may be estimated within a low average error across large variation in sleep transistor sizes and variation in circuit sizes with significant speedup in computation time compared to transistor-level circuit simulations [73].

6.1.7. Arithmetic Operators for Cryptography

Participants: Arnaud Tisserand, Emmanuel Casseau, Thomas Chabrier, Danuta Pamula, Karim Bigou, Franck Bucheron, Jérémie Métairie.

6.1.7.1. Arithmetic Operators for Fast and Secure Cryptography

Electrical activity variations in a circuit are one of the information leakage used in side channel attacks. In [65], we present $\mathbb{F}_{2^m}$ finite-field multipliers with reduced activity variations for asymmetric cryptography. Useful activity of typical multiplication algorithms is evaluated. The results show strong shapes, which can be used as a small source of information leakage. We propose modified multiplication algorithms and architectures to reduce useful activity variations. Useful activity has been evaluated using accurate FPGA emulation and activity counters at every operation cycle. Measurement analysis shows that the implemented multiplication algorithms (classical, Montgomery and Mastrovito) lead to specific shapes for the curve of activity variations which may be used as a small source of information leakage for some side channel attacks. We proposed modifications of selected $\mathbb{F}_{2^m}$ multipliers to reduce this information leakage source at two levels: architecture level by removing activity peaks due to control (e.g. reset at initialization) and algorithmic level by modifying the shape of the activity variations curve. Due to very low-level optimizations there is no significant area and delay overhead.

Paper [64] presents overview of the most interesting $\mathbb{F}_{2^m}$ multiplication algorithms and proposes efficient hardware solutions applicable to elliptic curve cryptosystems. It focuses on fields of size $m = 233$, one of the sizes recommended by NIST (National Institute of Standards and Technology). We perform an analysis of most popular algorithms used for multiplication over finite fields; suggest efficient hardware solutions and point advantages and disadvantages of each algorithm. The article overviews and compares classic, Mastrovito and Montgomery multipliers. Hardware solutions presented here, implement their modified versions to gain on efficiency of the solutions. Moreover we try to present a fair comparison with existing solutions. The designs presented here are targeted to FPGA devices.

6.1.7.2. ECC Processor with Protections Against SCA

A dedicated processor for elliptic curve cryptography (ECC) is under development. Functional units for arithmetic operations in $\mathbb{F}_{2^m}$ and $\mathbb{F}_p$ finite fields and 160–600-bit operands have been developed for FPGA implementation. Several protection methods against side channel attacks (SCA) have been studied. The use of some number systems, especially very redundant ones, allows one to change the way some computations are performed and then their effects on side channel traces.

6.1.8. 3D Heterogeneous SoC Design

Participants: Quang-Hai Khuat, Hoa Le, Sébastien Pillement, Emmanuel Casseau, Antoine Courtay, Daniel Chillet, Olivier Sentieys.

A three-dimensional system-on-chip is an SoC in which two or more layers of dies are stacked vertically into a single circuit and integrated within a single package. 3D stacking is an emerging solution that provides a new dimension in performance by reducing the distances that signals need to travel between the different blocks of a system. Interconnects in future technologies are known to be a major bottleneck for performance and power. In this context, 3D implementations can help alleviate the performance and power overheads of on-chip wiring.

In the context of 3D SoC, we have developed a spatio-temporal scheduling algorithm for 3D architecture composed of two layers: i) a homogenous Chip MultiProcessor (CMP) layer and ii) a homogeneous embedded Field-Programmable Gate Array (eFPGA) layer, interconnected by through-silicon vias (TSVs), thus ensuring tight coupling between software tasks on processors and associated hardware accelerators on the eFPGA. We extended the Proportionate-fair (Pfair) algorithm to tackle 3D heterogeneous multiprocessors. Unlike Pfair, our algorithm copes with task dependencies and global communication cost. Communication cost is computed by summing not only point-to-point/direct communication cost, but also memory cost. Our algorithm favours direct communication onto the eFPGA layer, but uses shared memory when direct communications are not possible [61], [75], [74].

6.2. Compilation and Synthesis for Reconfigurable Platform

Participants: Steven Derrien, Emmanuel Casseau, Daniel Menard, François Charot, Christophe Wolinski, Olivier Sentieys, Patrice Quinton.

6.2.1. Polyhedral-Based Loop Transformations for High-Level Synthesis

Participants: Steven Derrien, Antoine Morvan, Patrice Quinton.

After almost two decades of research effort, there now exists a large choice of robust and mature C to hardware tools that are used as production tools by world-class chip vendor companies. Although these tools dramatically slash design time, their ability to generate efficient accelerators is still limited, and they rely on the designer to expose parallelism and to use appropriate data layout in the source program. We believe this can be overcome by tackling the problem directly at the source level, using source-to-source optimizing compilers. More precisely, our aim is to study how polyhedral-based program analysis and transformation can be used to address this problem. In the context of the PhD of Antoine Morvan, we have studied how it was possible to improve the efficiency and applicability of nested loop pipelining (also known as nested software pipelining) in C to hardware tools. Loop pipelining is a key transformation in high-level synthesis tools as it helps maximizing both computational throughput and hardware utilization. Nevertheless, it somewhat looses its efficiency when dealing with small trip-count inner loops, as the pipeline latency overhead quickly limits its efficiency. Even if it is possible to overcome this limitation by pipelining the execution of a whole loop nest, the applicability of nested loop pipelining has so far been limited to a very narrow subset of loops, namely perfectly nested loops with constant bounds. In this work, we have extended the applicability of nested-loop pipelining to imperfectly nested loops with affine dependencies. We have shown how such loop nest can be analyzed and, under certain conditions, how one can modify the source code in order to allow nested loop pipeline to be applied using a method called polyhedral bubble insertion. The approach has been implemented in the Gecos source-to-source toolbox and was validated using two leading-edge HLS commercial tools. It helps improving performance for a minor area overhead. This work has been accepted for publication in late 2012 to IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. In addition, the complete Gecos source-to-source toolbox was presented at the DAC university booth in June 2012.

In addition to our work on nested loop pipelining, we also started investigating how to extend existing polyhedral code generation technique to enable the synthesis of area-efficient control-logic for nested loops hardware accelerators.

6.2.2. Compiling for Embedded Reconfigurable Multi-Core Architectures

Participants: Steven Derrien, Olivier Sentieys, Maxime Naullet.

Current and future wireless communication and video standards have huge processing power requirements, which cannot be satisfied with current embedded single processor platforms. Most platforms now therefore integrate several processing core within a single chip, leading to what is known as embedded multi-core platforms. This trend will continue, and embedded system design will soon have to implement their systems on platforms comprising tens if not hundred of high performance processing cores. Examples of such architectures are the Xentium processor from by Recore or the Kahrisma processor, a radically new concept of morphable processor from Karlsruhe Institute of Technology (KIT). This evolution will pose significant design challenges, as parallel programming is notoriously difficult, even for domain experts. In the context of the FP7 European Project Alma (Architecture-oriented parallelization for high performance embedded Multi-core systems using scilAb), we are studying how to help designers programming these platforms by allowing them to start from a specification in Matlab and/or Scilab, which are widely used for prototyping image/video and wireless communication applications. Our research work in this field revolves around two topics. The first one aims at exploring how floating-point to fixed-point conversion can be performed jointly with the SIMD instruction selection stage to explore performance/accuracy trade-off in the software final implementation. The second one aims at exploring how program transformation techniques (leveraging the polyhedral model and/or based on the domain specific semantics of scilab built-in functions) can be used to enable an efficient coarse grain parallelization of the target application on such multi-core machines.

6.2.3. Reconfigurable Processor Extensions Generation

Participants: Christophe Wolinski, François Charot, Antoine Floc'h.

Most proposed techniques for automatic instruction sets extension usually dissociate pattern selection and instruction scheduling steps. The effects of the selection on the scheduling subsequently produced by the compiler must be predicted. This approach is suitable for specialized instructions having a one-cycle duration because the prediction will be correct in this case. However, for multi-cycle instructions, a selection that does not take into account scheduling is likely to privilege instructions which will be, a posteriori, less interesting than others in particular in the case where they can be executed in parallel with the processor core.

The originality of our research work is to carry out specialized instructions selection and scheduling in a single optimization step. This complex problem is modeled and solved using constraint programming. This approach allows the features of the extensible processor to be taken into account with a high degree of flexibility. Two architecture models are envisioned. The first one is an extensible processor tightly coupled to an hardware extension having internal registers used to store intermediate results. The second model is VLIW-oriented, a specialized instruction is able to configure several processing using working in parallel. Our experimental results show that these approaches are able to handle graphs of several hundred of nodes in a reasonable time (less than ten seconds for most cases). Speedups obtained are particularly interesting for applications having a high degree of instruction-level parallelism.

More details on constraint programming approach applied to reconfigurable processor extension generation can be found in [32] and in the Ph.D. thesis of Antoine Floc'h [20].

During this year, we have also studied a novel technique that addresses the interactions between code optimization and instruction set extension. The idea is to automatically transform the original loop nests of a program (using the polyhedral model) to select specialized and vectorizable instructions. These instructions may use local memories of the hardware extension to store intermediates data produced at a given loop iteration. Details can be found in the Ph.D. thesis of Antoine Floc'h [20].

6.2.4. Custom Operator Identification for High-Level Synthesis

Participants: Emmanuel Casseau, François Charot, Chenglong Xiao.

In this work, our goal is to propose an automated design flow based on custom operator identification for high-level synthesis. Custom operators that can be implemented in special hardware units make it possible to improve performance and reduce area of the design. The key issues involved in the design flow are: automatic enumeration and selection of custom operators from a given high-level application code and re-generation of the source code incorporating the selected custom operators. This new source code is then provided to the high-level synthesis tool. The application is first translated into an internal representation based on a graph representation. Then the problem is to enumerate and select subgraphs that will be implemented as custom operators. However, enumerating all the subgraphs is a computationally difficult problem. In Xiao's PhD thesis [25] and [42], three enumeration algorithms for exact enumeration of subgraphs under various constraints were proposed. Compared to a previously proposed well-known algorithm, the proposed enumeration algorithms can achieve orders of magnitude speedup. Selecting a most profitable subset from the enumerated subgraphs is also a time-consuming job. [25] proposed three different selection heuristics targeting different objectives. Based on these algorithms, experimental results show that the approach achieves on average 19% area reduction, compared to a traditional high-level synthesis with CtoS tool from Cadence. Meanwhile, the latency is reduced on average by 22%.

6.3. Interaction between Algorithms and Architectures

6.3.1. Numerical Accuracy Analysis and Optimization

Participants: Daniel Menard, Karthick Parashar, Olivier Sentieys, Romuald Rocher, Pascal Scalart, Aymen Chakhari, Jean-Charles Naud, Emmanuel Casseau.

Most of analytical methods for numerical accuracy evaluation use perturbation theory to provide the expression of the quantization noise at the output of a system. Existing analytical methods do not consider a correlation between noise sources. This assumption is no longer valid when a unique datum is quantized several times. In [34], an analytical model of the correlation between quantization noises is provided. The different quantization modes are supported and the number of eliminated bits is taken into account. The expression of the power of the output quantization noise is provided when the correlation between the noise sources is considered. The proposed approach allows improving significantly the estimation of the output quantization noise power compared to the classical approach, with a slight increase of the computation time.

An analytical approach is studied to determine accuracy of systems including unsmooth operators. An unsmooth operator represents a function which is not derivable in all its definition interval (for example the sign operator). The classical model is no valid yet since these operators introduce errors that do not respect the Widrow assumption (their values are often higher than signal power). So an approach based on the distribution of the signal and the noise is proposed. It is applied to the sphere decoding algorithm to determine analytically the error probability due to quantization [53]. We also focus on recursive structures where an error influences future decision. So, the Decision Feedback Equalizer is also considered. In that case, numerical analysis method (as Newton Raphson algorithm) can be used. Moreover, an upper bound of the error probability can be analytically determined. A method to determine the distribution of the noise due to quantization at the output of a system made of smooth operators has been developed [70]. It is based on Generalized Gaussian Distribution and allows take under consideration all possible distributions (uniform, gaussian, laplacian, etc.).

6.3.2. Multi-Antenna Systems

Participants: Olivier Berder, Pascal Scalart, Quoc-Tuong Ngo, Viet-Hoa Nguyen.

Still considering the maximization of the minimum Euclidean distance, we proposed a new linear precoder obtained by observing the SNR-like precoding matrix. An approximation of the minimum distance is derived, and its maximum value was obtained by maximizing the minimum diagonal element of the SNR-like matrix. The precoding matrix is first parameterized as the product of a diagonal power allocation matrix and an input-shaping matrix acting on rotation and scaling of the input symbols on each virtual subchannel. We demonstrated that the minimum diagonal entry of the SNR-like matrix is obtained when the input-shaping matrix is a DFT-matrix. The major advantage of this design is that the solution can be available for all

rectangular QAM-modulations and for any number of datastreams [35], [36], [37]. To reduce the decoding complexity of linearly precoded MIMO systems, the sphere decoder was applied instead of maximum likelihood and the performance complexity trade-off was investigated. The sphere decoding (SD) algorithm, proposed as a sub-optimal ML-decoding, just considers a subset of lattice points that drop into the sphere centered by the received point to obtain the decoded solution, thus reducing significantly the complexity. Because the structure of our precoder is complicated and strongly depends on the channel, it exists the case when all power is poured only on the best sub-channel. Some adjustments, therefore, of traditional sphere decoding algorithm were mandatory to adapt to the precoded MIMO systems.

6.3.3. *Impact of RF Front-End Nonlinearity on WSN Communications*

Participants: Amine Didioui, Olivier Sentieys, Carolynn Bernier [CEA Leti].

6.3.4. *HarvWSNet: A Co-Simulation Framework for Energy Harvesting Wireless Sensor Networks*

Participants: Amine Didioui, Olivier Sentieys, Carolynn Bernier [CEA Leti].

Recent advances in energy harvesting (EH) technologies now allow wireless sensor networks (WSNs) to extend their lifetime by scavenging the energy available in their environment. While simulation is the most widely used method to design and evaluate network protocols for WSNs is simulation, existing network simulators are not adapted to the simulation of EH-WSNs and most of them provide only a simple linear battery model. To overcome these issues, we have proposed HarvWSNet, a co-simulation framework based on WSNet and Matlab that provides adequate tools for evaluating EH-WSN lifetime [56]. Indeed, the framework allows for the simulation of multi-node network scenarios while including a detailed description of each node's energy harvesting and management subsystem and its time-varying environmental parameters. A case study based on a temperature monitoring application has demonstrated HarvWSNet?s ability to predict network lifetime while minimally penalizing simulation time.

6.3.5. *Cooperative Strategies for Low-Energy Wireless Networks*

Participants: Olivier Berder, Olivier Sentieys, Le-Quang-Vinh Tran, Duc-Long Nguyen.

Recently, cooperative relay techniques (e.g. repetition-based or distributed space-time code based (DSTC-based) protocols) are increasingly of interest as one of the advanced techniques to mitigate the fading effects of transmission channel. We proposed a novel cooperative scheme with data exchange between relays before using distributed space-time coding. This fDSTC (full Distributed Space-Time Code) was compared with the conventional distributed space-time coded (cDSTC) protocol. Then, the thorough comparison of the fDSTC and cDSTC protocols in case of non-regenerative relays (NR-relays) and regenerative relays (R-relays) were considered in terms of error performance, outage probability, diversity order and energy consumption via both numerical simulations and mathematical analysis [24]. The previous works consider the energy efficiency of the cooperative relays techniques under the view of ideal medium access control (MAC) protocol. However, MAC protocol is responsible for regulating the shared wireless medium access of the networks, therefore, it has great influences on the total energy consumption of the networks. That lead us to a big motivation to design a cooperative MAC protocol, RIC-MAC (Receiver Initiated Cooperative MAC), by combining preamble sampling and cooperative relay techniques. The analytic results still confirm the interest of using cooperative relay techniques. However, the energy efficiency of the cooperative relay systems may be affected by MAC protocol design, the traffic loads of the networks and the desired latency [24].

6.3.6. *Opportunistic Routing*

Participants: Olivier Berder, Olivier Sentieys, Ruifeng Zhang.

However, the aforementioned approaches introduce an overhead in terms of information exchange, increasing the complexity of the receivers. A simpler way of exploiting spatial diversity is referred to as opportunistic routing. In this scheme, a cluster of nodes still serves as relay candidates but only a single node in the cluster forwards the packet [80]. Energy efficiency and transmission delay are very important parameters for wireless multihop networks. Numerous works that study energy efficiency and delay are based on the assumption of reliable links. However, the unreliability of channels is inevitable in wireless multihop networks. We investigated the tradeoff between the energy consumption and the latency of communications in a wireless multihop network using a realistic unreliable link model [43]. It provided a closed-form expression of the lower bound of the energy-delay tradeoff and of energy efficiency for different channel models (additive white Gaussian noise, Rayleigh fast fading and Rayleigh block-fading) in a linear network. These analytical results are also verified in 2-dimensional Poisson networks using simulations. The closed-form expression provides a framework to evaluate the energy-delay performance and to optimize the parameters in physical layer, MAC layer and routing layer from the viewpoint of cross-layer design during the planning phase of a network.

6.3.7. *Adaptive Techniques for WSN Power Optimization*

Participants: Olivier Berder, Daniel Menard, Olivier Sentieys, Mahtab Alam, Trong-Nhan Le.

We proposed a self-organized asynchronous medium access control (MAC) protocol for wireless body area sensor (WBASN). A body sensor network exhibits a wide range of traffic variations based on different physiological data emanating from the monitored patient. In this context, we exploit the traffic characteristics being observed at each sensor node and propose a novel technique for latency-energy optimization at the MAC layer [48], [26]. The protocol relies on dynamic adaptation of wake-up interval based on a traffic status register bank. The proposed technique allows the wake-up interval to converge to a steady state for variable traffic rates, which results in optimized energy consumption and reduced delay during the communication. The results show that our protocol outperforms the other protocols in terms of energy as well as latency under the variable traffic of WBASN.

System lifetime is the crucial problem of Wireless Sensor Networks (WSNs), and exploiting environmental energy provides a potential solution for this problem. When considering self-powered systems, the Power Manager (PM) plays an important role in energy harvesting WSNs. Instead of minimizing the consumption energy as in the case of battery powered systems, it makes the harvesting node converge to Energy Neutral Operation (ENO) to achieve a theoretically infinite lifetime and maximize the system performance. In [62], a low complexity PM with a Proportional Integral Derivative (PID) controller is introduced. This PM monitors the buffered energy in the storage device and performs adaptation by changing the wake-up period of the wireless node. This shows the interest of our approach since the impractical monitoring harvested energy as well as consumed energy is not required as it is the case in other previously proposed techniques. Experimental results are performed on a real WSN platform with two solar cells in an indoor environment. The PID controller provides a practical strategy for long-term operations of the node in various environmental conditions.

6.3.8. *WSN for Health Monitoring*

Participants: Patrice Quinton, Olivier Sentieys.

Applications of wireless sensor devices were also considered in the domain of health monitoring. Together with researchers from CASA team of IRISA-UBS, we investigated the possibility of using ECG-sensors to remotely monitor the cardiac activity of runners during a marathon race, using off-the shelf sensing devices and a limited number of base stations deployed along the marathon route. Preliminary experiments showed that such a scenario is indeed viable, although special attention must be paid to balancing the requirements of ECG monitoring with the constraints of episodic, low-rate transmissions.

The proliferation of private, corporate and community Wi-Fi hotspots in city centers and residential areas opens up new opportunities for the collection of biomedical data produced by sensors carried by mobile non-hospitalized subjects. Using disruption-tolerant networks, it was shown that biomedical data could be recorded using nearby hotspot. A scenario involving a subject wearing an ECG-enabled sensor walking in the streets of a residential area was reported.

These researches, combined with new sensor devices developed by the BOWI project, open up a large range of applications where high-performance sensor devices would allow health monitoring, or sport events organization.

6.3.9. Reconfigurable Video Coding

Participants: Emmanuel Casseau, Hervé Yviquel.

In the field of multimedia coding, standardization recommendations are always evolving. To reduce design time taking benefit of available SW and HW designs, Reconfigurable Video Coding (RVC) standard allows defining new codec algorithms. The application is represented by a network of interconnected components (so called actors) defined in a modular library and the behaviour of each actor is described in the specific RVC-CAL language. Dataflow programming, such as RVC applications, express explicit parallelism within an application. However general purpose processors cannot cope with both high performance and low power consumption requirements embedded systems have to face. Hence we are investigating the mapping of RVC specifications on hardware accelerators or on many tiny core platforms. Actually, our goal is to propose an automated co-design flow based on the Reconfigurable Video Coding framework. The designer provides the application description in the RVC-CAL dataflow language, after which the co-design flow automatically generates a network of processors that can be synthesized on FPGA platforms. We are currently focussing on a many-core platform based on the TTA processor (Very Long Instruction Word -style processor). Hervé Yviquel did a 4-months stay (Spring 2012) at Tampere University of Technology, Finland, in the group of Jarmo Takala who is developing a co-design toolset for TTA processor automated generation. Such a methodology permits the rapid design of a many-core signal processing system which can take advantage of all levels of parallelism. This work is done in collaboration with Mickael Raulet from IETR INSA Rennes and has been implemented in the Orcc open-source compiler. At present time the mapping of the RVC-CAL actor network is straightforward: every actor is mapped on a TTA processor based on our collaboration with Jani Boutellier from the University of Oulu (Finland). To reduce the area of the platform, TTA processor usage rate has to be improved, i.e. several actors are to be mapped onto a single processor. Work in progress is about this. It requires an actor partitioning step to define the set of actors that will be executed on the same processor. Due to the dynamic behaviour of the application, we expect we will be able to use profiling to get some feedbacks for the partitioning.

6.3.10. A Low-Complexity Synchronization Method for OFDM Systems

Participants: Pramod P. Udupa, Olivier Sentieys, Pascal Scalart.

A new hierarchical synchronization method was proposed for initial timing synchronization in orthogonal frequency-division multiplexing (OFDM) systems. Based on the proposal of new training symbol, a threshold based timing metric is designed for accurate estimation of start of OFDM symbol in a frequency selective channel. Threshold is defined in terms of noise distributions and false alarm which makes it applicable independent of type of channel it is applied. Frequency offset estimation is also done for the proposed training symbol. The performance of the proposed timing metric is evaluated using simulation results. The proposed method achieves low mean squared error (MSE) in timing offset estimation at five times lower computational complexity compared to cross-correlation based method in a frequency selective channel. It is also computationally efficient compared to hybrid approaches for OFDM timing synchronization.

6.3.11. Flexible hardware accelerators for biocomputing applications

Participants: Steven Derrien, Naeem Abbas, Patrice Quinton.

It is widely acknowledged that FPGA-based hardware acceleration of compute intensive bioinformatics applications can be a viable alternative to cluster (or grid) based approach as they offer very interesting MIPS/watt figure of merits. One of the issues with this technology is that it remains somewhat difficult to use and to maintain (one is rather designing a circuit rather than programming a machine). Even though there exists C-to-hardware compilation tools (Catapult-C, Impulse-C, etc.), a common belief is that they do not generally offer good enough performance to justify the use of such reconfigurable technology. As a matter of fact, successful hardware implementations of bio-computing algorithms are manually designed at RT-level

and are usually targeted to a specific system, with little if any performance portability among reconfigurable platforms. This research work, funded by the ANR BioWic project, aims at providing a framework for helping semi-automatic generation of high-performance hardware accelerators. This research work builds upon the CAIRN research group expertise on automatic parallelization for application specific hardware accelerators and has been targeting mainstream bioinformatics applications (HMMER, ClustalW and BLAST). The Biowic project ended in early 2012. Naeems Abbas, a PhD student funded by the project defended his PhD in May 2012.

CAMUS Team

6. New Results

6.1. VMAD

Participants: Alexandra Jimborean, Philippe Clauss, Jean-François Dollinger, Aravind Sukumaran-Rajam, Juan Manuel Martinez Caamaño, Vincent Loechner.

The goal of the VMAD project is to provide a set of annotations (pragmas) that the user can insert in the source code to perform advanced analyses and optimizations, for example dynamic speculative parallelization.

VMAD contains a modified LLVM compiler and a runtime system. The program binary files are first generated by our compiler to include necessary data, instrumentation instructions, parallel code templates, and callbacks to the runtime system. External modules associated to specific analyses and transformations are dynamically loaded when required at runtime. Dynamic information, such as memory locations of the modules entries, are patched at startup in the loaded executable.

VMAD uses sampling and multi-versioning to limit the runtime overhead (profiling, analysis, and code generation). At runtime, targeted codes are launched by successive chunks that can be either original, instrumented or optimized/parallelized versions. After each chunk execution, decisions can be taken relatively to the current optimization strategy. VMAD is handling advanced memory access profiling [17] through linear interpolation of the addresses, dynamic dependence analysis, version selection [17] and speculative polyhedral parallelization [19], [16].

Alexandra Jimborean defended her PhD thesis on this topic in 2012 [12]. In 2012, Aravind Sukumaran-Rajam started a PhD in our team to continue this work, especially on extending the dependence analysis to make it handle more general programs, keeping it fast and accurate. Jean-François Dollinger will extend the framework to handle heterogeneous architectures (GPGPUs). Juan Manuel Martinez Caamaño, a master student of University of Buenos Aires (associate team EA-Ancome) is also working on VMAD to make the code generation support tiling.

6.2. The Multifor programming construct

Participants: Philippe Clauss, Imèn Fassi, Yosr Slama, Matthieu Kuhn.

We have proposed a new programming control structure called “multifor”, allowing to take advantage of parallelization models that were not naturally attainable with the polytope model before. In a multifor-loop, several loops whose bodies are run simultaneously can be defined. Respective iteration domains are mapped onto each other according to a run frequency – the grain – and a relative position – the offset –. Execution models like dataflow, stencil computations or MapReduce can be represented onto one referential iteration domain, while still exhibiting traditional polyhedral code analysis and transformation opportunities. Moreover, this construct provides ways to naturally exploit hybrid parallelization models, thus significantly improving parallelization opportunities in the multicore era. Traditional polyhedral software tools are used to generate the corresponding code. Additionally, a promising perspective related to non-linear mapping of iteration spaces has also been developed, yielding to run a loop nest inside any other one by solving the problem of inverting “ranking Ehrhart polynomials”.

This work is the PhD work of Imèn Fassi, who started her work this year and who is co-advised by Yosr Slama, Assistant Professor at the University El Manar in Tunis, Tunisia, and Philippe Clauss. A first publication of this topic has been accepted at the IMPACT workshop that will be held in conjunction with the HIPEAC conference in Berlin, Germany, January 2013.

6.3. Parwiz: dynamic data dependence analysis

Participants: Alain Ketterlin, Philippe Clauss.

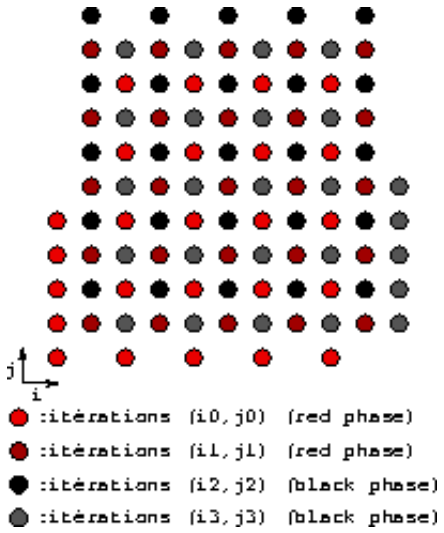

Figure 3. Red-Black Gauss-Seidel Multifor Iteration Space

We have continued working on dynamic data-dependence analysis during this year, especially on increasing the scope of our tool (called Parwiz). For instance, Parwiz is now able to suggest several program transformations (like loop distribution) that enable loop vectorization. It uses an algorithm known as *codegen* (developed by Allen & Kennedy), but the novelty is that it applies the algorithm to dependence graphs that are built empirically, by running the program on selected input data sets. As far as we know, Parwiz is the first tool able to suggest loop transformations.

We have also developed several other empirical analysis. One of these focuses on loops that are not parallel, but whose iterations present significant parallelism provided the program explicitly schedules the various iterations. This still lacks a suitable cost model to estimate the potential gain, but gives significant insight into the behavior of a given non-parallel loop.

This work has been presented at the MICRO-45 conference held in Vancouver on december 1–5 2012 [18].

6.4. Modeling the behavior of parallel traces

Participants: Alain Ketterlin, Stéphane Genaud.

We have started this year a project aiming at developing algorithms and tools to capture the behavior of parallel programs. Our initial goal is automatically obtain formal models of communicating MPI processes, in terms of message sends and receives and of synchronization events. Such models have various uses, the first of them being the visualization of the system's communications, for debugging, or plain understanding (see below, Figure 4). However, we expect to develop other applications, for example in optimizing the communication infrastructure (or routing algorithm) for specific applications.

Our modeling algorithm works in two phases. The first phase is local to each node, using our work on nested loop recognition [7]. This builds a sequence of loop nests providing a compact representation of all local communication events. At the end of the run, the various local models are merged, typically through a parallel reduction operation, to build the global model.

We plan to publish the first part of this work in the first half of 2013. Several experimental data have been collected already, but we would like to evaluate the overall task on significantly sized programs.

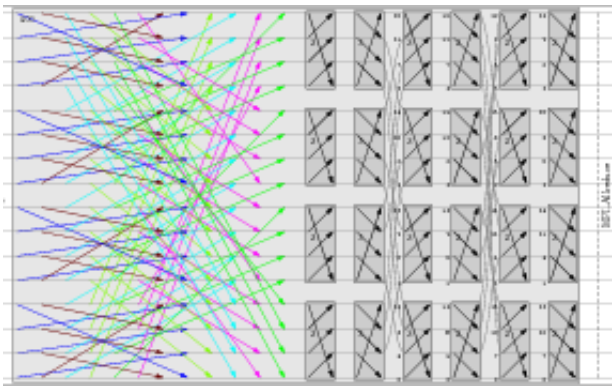

Figure 4. Visualizing parallel traces

Currently, the whole process is restricted to communication events. However, it can be immediately extended to trace including other kinds of events, like the addresses and sizes of memory buffers transmitted from process to process. This would provide a complete, run time description of the program, which could be used to evaluate the potential gain of various re-parallelization techniques. This aspect is the next goal on our agenda.

6.5. Certified polyhedral transformations into more and more concrete languages

Participants: Nicolas Magaud, Julien Narboux, Éric Violard.

We continued our work to complete the proof of polyhedral based transformations in the language *Loops* designed by Alexandre Pilkiewicz (see the proof scheme on Fig. 5). Our idea is to use once again a validator. The validation here consists in comparing two polyhedrons: the one (**pprogopt**) obtained from the non-optimized Loops program (**prog**), by translation to the polyhedral language (*Plang*) (**pprog**), and then optimization in *Plang*; and the one (**interprogopt**) obtained from the optimized Loops program (**progopt**) by translation into *Plang*. If these two polyhedrons are the same, then the validator returns true, otherwise it returns false. The proof that the non-optimized and optimized programs have the same behaviour lies on the deterministic property of the function that translates a program *Loops* into *Plang*. We obtained the proof in Coq that our scheme is correct. Now, we have to complete the implementation of our optimizing compiler for *Loops* by connecting our validator with the off the shell tools for polyhedral transformations. We will use the tool PLuTo[10] to find efficient code transformations and CLooG[11] to generate the loops from the polyhedral representation (we proposed an internship for this purpose).

We now have to connect the language *Loops* with more concrete languages (whose features and semantics have to be defined). We already showed how to deal with arithmetic overflows in a more concrete language where each loop variable is a machine integer [20]. Our approach is thus to incrementally add concrete features until joining an intermediate language of CompCert.

Since the members of our team have some skill in defining new languages and their semantics, we thought that it could be a good idea to exploit this and to define a formal semantics for the **Multifor** syntactic sugar proposed by Philippe Clauss. We aims at associating a rigorous mathematical meaning with this syntactic construct: first a denotational semantics and then an operational one. This work will serve as a base to prove correct the compilation process that translates this construct into intermediate code.

[10] http://pluto-compiler.sourceforge.net/
[11] http://www.cloog.org/

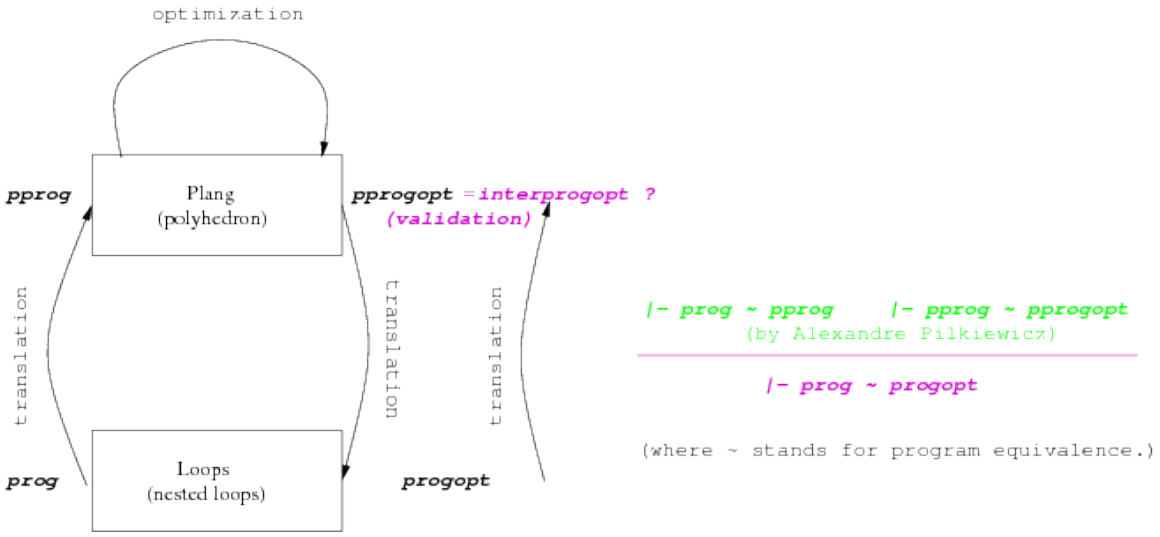

Figure 5. Our proof scheme for a certified compiler of Loops

COMPSYS Project-Team

6. New Results

6.1. Enhancing the Compilation of Synchronous Data-Flow Languages

Participants: Paul Feautrier, Abdoulaye Gamatié [LIFL], Laure Gonnord [Compsys/LIFL].

In [25] a new (light) numerical-Boolean abstraction was proposed for an efficient static analysis of synchronous programs that describe multi-clock embedded systems in the language Signal. In this abstraction, relations between clocks and numerical variables are modeled by Boolean-affine formulas. These formulas can easily be extracted from the program text. From the results of a satisfiability test of these formulas, clock properties can be deduced, which, when submitted to the Signal compiler, may improve the resulting target program.

In collaboration with Abdoulaye Gamatié, we proposed an extension of the previous approach to modular programs. This extension necessitates the use of an extended SMT (satisfiability modulo theory) solver – able for instance to deal with quantifier elimination – which has been implemented by Paul Feautrier by reusing some of the Syntol tools. This work is still unpublished but will be soon submitted to a journal.

6.2. Dataflow Analysis of Polyhedral X10 Programs

Participants: Paul Feautrier, Sanjay Rajopadhye [Colorado State Universty], Vijay Saraswat [IBM Research], Tomofumi Yuki [Colorado State University].

X10 is a recent parallel language, developed by IBM Research, whose aim is to increase programmers productivity. It is a descendant of Java and it includes several new parallel constructs, such as `async` and `finish`, which generalize `fork` and `join`, clocks, which generalize barriers, and `at`, which enables the remote execution of program fragments. X10 programs are guaranteed to be deadlock-free, but may exhibit non-deterministic behaviors or *races*.

We have devised a verifier for the `async/finish` fragment of X10 and polyhedral programs. The approach consists in computing the source of each array access. In the presence of parallel constructs, the sequencing predicate is no longer a total order and a read may have several sources, which indicates a race. A proof-of-concept tool has been implemented. This work will be presented at the next *Principles and Practice of Parallel Programming* conference (PPoPP'13 in Shenzen, China) [13].

6.3. Data-Aware Process Networks

Participants: Christophe Alias, Alexandru Plesco [Compsys/Zettice].

New techniques were introduced to generate and compile optimized data-aware process networks, from a C program annotated with pragmas (see Section 5.9 and the software tool Dcc). These techniques are essential for the Zettice start-up and are not made publicly available for the moment.

6.4. Optimizing Remote Accesses for HLS

Participants: Christophe Alias, Alain Darte, Alexandru Plesco [Compsys/Zettice].

Some data- and compute-intensive applications can be accelerated by offloading portions of codes to platforms such as GPGPUs or FPGAs. However, to get high performance for these kernels, it is mandatory to restructure the application, to generate adequate communication mechanisms for the transfer of remote data, and to make good usage of the memory bandwidth. In the context of the high-level synthesis (HLS), from a C program, of hardware accelerators on FPGA, we showed how to automatically generate optimized remote accesses for an accelerator communicating to an external DDR memory. Loop tiling is used to enable block communications, suitable for DDR memories. Pipelined communication processes are generated to overlap communications and computations, thereby hiding some latencies, in a way similar to double buffering. Finally, not only intra-tile but also inter-tile data reuse is exploited to avoid remote accesses when data are already available in the local memory.

We showed how to generate the sets of data to be read from (resp. written to) the external memory just before (resp. after) each tile so as to reduce communications and reuse data as much as possible in the accelerator. The main difficulty arises when some data may be (re)defined in the accelerator and should be kept locally. We proposed an automatic optimized code generation scheme, entirely at source-level, i.e., in C, that allows us to compile all the necessary glue (the communication processes) with the same HLS tool as for the computation kernel. Our method, implemented in the tool Chuba (see Section 5.7) uses advanced polyhedral techniques for program analysis and transformation. Experiments with Altera HLS tools demonstrate how to use our techniques to efficiently map C kernels to FPGA.

This work, astride two different fields (compilation for high-performance computing and high-level synthesis) turned out to be very difficult to publish. It was finally accepted at PPoPP'12 [6], but only as a short paper (2 pages). We requested to retain the copyright of this work to be able to publish a longer version. It was accepted at the IMPACT'12 workshop [7], which makes paper available on the web, but with no copyright. It was finally accepted as a full publication at the DATE'13 conference [8].

6.5. Parametric Inter-Tile Reuse for Kernel Offloading

Participants: Alain Darte, Alexandre Isoard.

The method described in Section 6.4 is not parametric in terms of the tile size, i.e., the tile size needs to be fixed before compiling the program. Furthermore, the size of the required local memory depends on the tile size and is available only after program analysis. As a result, to select the tile size with respect to the size of the local memory, the program first needs to be compiled (actually analyzed) for all tile sizes. A parametric program analysis would be much more convenient. The situation is even worse to get the runtime performances in terms of the tile size. Indeed, so far, Chuba generates a C code that is generic and cannot be immediately compiled by C2H. A few modifications by hand are still needed, such as inserting the adequate pragmas for C2H, transforming array accesses to linearized addresses with the right base addresses, changing some arrays into non-aliasing pointers so that C2H, whose dependence analyzer and software pipeliner are weak, can generate codes with the right initiation intervals, etc. These changes are minor and systematic and take time when performed by hand. A fully-parametric compilation scheme would be a plus.

The formulation proposed in Section 6.4 is unfortunately quadratic in terms of the tile size, which prevents to parameterize it. Indeed, it relies on parametric linear programming, which works only with a linear use of parameters. As part of the Master internship of Alexandre Isoard, we nevertheless succeeded to design a fully-parametric scheme for inter-tile reuse and buffer size computation. The method is much more involved but is still compatible with approximations. These results have still to be implemented and submitted for publication.

6.6. Semantic Program Transformations

Participants: Christophe Alias, Guillaume Iooss, Sanjay Rajopadhye [Colorado State University].

Traditionally, a program transformation is considered to be *correct* if each data dependence of the original program is respected. In that case, both original and transformed programs perform *exactly* the same computation. We can relax this condition by expecting both programs to perform the same computation, modulo the *semantic* properties of the operators (e.g., associativity, commutativity). Semantic program transformations extend the traditional corpus of program transformations and can reveal new optimization opportunities.

More specifically, we are interested in *semantic loop tiling*, a special case of loop tiling, where the input arrays are tiled, and the program is restructured to use high-level matrix operations between data tiles, instead of the original scalar operations. Surprisingly, it turns out that in most cases, the semantic tiling is simply obtained by substituting the scalar variables by the tiles (matrices), and the original operators by the corresponding matrix operators (e.g., a/b by MatMul(A,Inv(B))). The approach currently investigated consists in two steps: (i) guess the semantic tiling, and (ii) prove the (semantic) equivalence with the original program.

Our current contribution is an heuristic to check the equivalence of two programs modulo associativity/commutativity so as to achieve the step (ii). The two programs should fit in the polyhedral model but can involve explicit reductions. This work is currently under submission, and is part of the PhD thesis of Guillaume Iooss.

6.7. Modular Termination of Large Programs

Participants: Christophe Alias, Guillaume Andrieu [LIFL], Laure Gonnord [Compsys/LIFL].

Program termination is an essential step in program verification. In [16], we showed how to check the termination of programs whose control can be summarized by an integer interpreted automaton. This was done by computing a *ranking function* (kind of schedule) by means of integer linear programming techniques. This approach, though powerful, clearly lacks scalability and cannot handle large programs.

We overcame this limitation by proceeding into two steps. First, we extract, from the program to be analyzed, the part useful for termination, i.e., the smaller program slice with the same control behavior. Then, we show that proving the termination of the whole program (slice) boils down to prove the termination of small programs, which can be handled by the technique of [16]. Experimental results show that many large programs can be handled this way.

This work was part of the engineer internship of Guillaume Andrieu. Our technique has been implemented in a tool called SToP (see Section 5.12) and presented at the workshop TAPAS'12 [9].

6.8. Lower Bounds for the Inherent Data Locality Properties of Computations

Participants: Venmugil Elango [OSU, Columbus, USA], Louis-Noël Pouchet [UCLA, Los Angeles, USA], P. Sadayappan [OSU, Columbus, USA], J. (Ram) Ramanujam [LSU, Houston, USA], Fabrice Rastello.

Data movement will account for most of the energy as well as execution time on upcoming exascale architectures, including data movement between processors as well as data movement across the memory hierarchy within each processor. Therefore a fundamental characterization of the data access complexity of algorithms is increasingly important.

We addressed the problem of data access or I/O complexity in a two-level memory hierarchy, as studied in the seminal work of Hong and Kung [26]. We developed a novel approach based on graph min-cut for deriving lower bounds on I/O complexity with two significant advantages over the S-partitioning model of Hong and Kung: (1) the approach can be used to develop analytical expressions with tighter lower bounds for I/O, and (2) unlike any previous model, our new lower bound approach can be automated for analyzing an arbitrary computational directed acyclic graph. We show tighter analytically-derived lower bounds as well as very promising experimental results thanks to a prototype tool that implements our fully-automated analysis.

This work has been submitted and is part of an informal collaboration with P. Sadayappan from the University of Columbus (CSU).

6.9. A Polynomial Spilling Heuristic: Layered Allocation

Participants: Albert Cohen [Inria, Parkas], Boubacar Diouf [Université Paris Sud, Parkas], Fabrice Rastello.

Register allocation is subdivided into two sub-problems: first, the *allocation* (or its dual problem the *spilling*) selects the set of variables that will reside in registers (resp. in memory) at each point of the program. Then, the *assignment* or *coloring* picks a specific register where a variable will reside. Building on some properties of the static single assignment form (SSA), it is now possible to decouple the allocation from the assignment. Indeed, the interference graph of a program in SSA form is a chordal graph. In this context, MAXLIVE, the maximal number of variables simultaneously live at a program point, is used during the spilling phase as a criterion to guarantee that the forthcoming assignment will be performed without any spill. If MAXLIVE is lower than or equal to R, the number of available registers, then all the variables will be assigned without any spill. This *decoupled* approach was advocated by Fabri, Appel and George, Darte et al., and others.

Existing spilling heuristics rely on a sufficient condition to guarantee register assignment, and incrementally spill until the condition holds. As we just mentioned, for programs under SSA, the condition is necessary and sufficient: MAXLIVE has to be lower than or equal to R. Incremental spilling decisions to satisfy this condition tend to be overly local and suboptimal. Indeed, incremental spilling itself is NP-complete, and heuristics based upon it trade too much their optimality for polynomiality. In contrast to incremental spilling, we proposed to adopt the symmetric approach: incremental allocation. Intuition for it emerges from two observations allowing for more global spilling decisions:

1. Register allocation is pseudo-polynomial in the number of registers, suggesting a heuristic that solves (optimally) roughly $R/step$ allocation problems on $step$ registers each. The final allocation is the layered composition of the stepwise allocations.
2. Stepwise optimality does not guarantee an overall optimal allocation, but experiments show that it comes very close to optimal, even with $step = 1$. Intuition for this comes from recent work by Diouf et al., observing that allocation decisions tend to be a monotonic function of the number of registers.

This work, which will be presented at CGO'13 [11], proposes a new graph-based allocation heuristic, based on a maximum clique cover formulation to define the profitability of spilling variables. It exploits the pseudo-polynomial complexity in the number of registers of the allocation problem under SSA — as opposed to the symmetric, spilling problem, which remains strongly NP-complete. It addresses the spill-everywhere problem in a decoupled context and also proposes an extension to non-decoupled approaches. It introduces *layered allocation* a new strategy that incrementally allocates variables instead of incrementally spilling variables. The evaluation performed on standard benchmarks shows that this new approach is near-optimal.

6.10. Interaction Between Spilling and Scheduling

Participants: Quentin Colombet, Alain Darte, Fabrice Rastello.

As explained in Section 6.9 , it is possible to decouple the register allocation problem in two successive phases: a first *spilling* phase places `load` and `store` instructions so that the register pressure at all program points is small enough, a second *assignment* and *coalescing* phase maps the remaining variables to physical registers and reduces the number of `move` instructions among registers. At CASES'11 [18], we presented a new integer linear programming (ILP) formulation, for load-store architectures, to capture "optimal" spilling in a more accurate and more expressive way than previous approaches. Among other features, we can express SSA ϕ-functions, memory-to-memory copies, and the fact that a value can be stored simultaneously in a register and in memory.

We used this ILP formulation to experimentally analyze the impact of the different heuristic strategies and compare them with optimal solutions. While "optimal" solutions show significant improvements for static spill costs, it turned out that runtime performances were disappointing (if not random). We conducted various experiments to understand this behavior and discovered that the interaction with scheduling is actually higher than expected. Micro-architectural features (e.g., memory latencies that can be hidden by prefetching, bundling that can hide cycles) have to be accounted for in the model, which is never done. These experiments and analysis are described in Chapter 4 of Quentin Colombet's PhD thesis [1].

6.11. Elimination of Parallel Copies Using Code Motion on Data Dependence Graphs

Participants: Florian Brandner, Quentin Colombet.

Traditional approaches to copy elimination during register allocation are based on interference graphs and register coalescing. Variables are represented as nodes in a graph, which are coalesced, if they can be assigned the same register. However, decoupled approaches strive to avoid interference graphs and thus often resort to local recoloring.

A common assumption of existing coalescing and recoloring approaches is that the original ordering of the instructions in the program is not changed. We developed an extension of a local recoloring technique called Parallel Copy Motion. We perform code motion on data dependence graphs in order to eliminate useless copies and reorder instructions, while at the same time a valid register assignment is preserved. Our results show that even after traditional register allocation with coalescing our technique is able to eliminate an additional 3% (up to 9%) of the remaining copies and reduce the weighted costs of register copies by up to 25% for the SPECINT 2000 benchmarks. In comparison to Parallel Copy Motion, our technique removes 11% (up to 20%) more copies and up to 39% more of the copy costs.

These results have been accepted for publication at SAC'12 [10] and, in a longer version, in the journal Computer Languages, Systems, and Structures [5].

AOSTE Project-Team

6. New Results

6.1. Logical time in Model-Driven Engineering embedded design

Participants: Charles André, Frédéric Mallet, Julien Deantoni, Marie-Agnès Peraldi Frati, Arda Goknil, Nicolas Chleq.

6.1.1. TimeSquare

We progressed our work on the foundations of logical time modeling as present in MARTE Time Model and our CCSL clock constraint specification language, while continuing the development of the TimeSquare tool environment which supports this in practice. A technical position paper was presented to the international TOOLS conference [22].

6.1.2. ECL (Event Constraint Language

Our contributions on CCSL and Time Model to the MARTE profile are part of the standard, but so far expressed in a syntax that is clearly distinct of the former UML notations. On the other hand, UML provides a textual language, named OCL, to express well-formedness constraints on diagram models and metamodels. While the original objectives were quite different, it seemed tempting to extend or adapt the general OCL philosophy, and to apply it then to timing and performance constraints as targeted by CCSL. The goal is to able the description of MoCs in an appropriate syntax, at metamodeling level. The result was a new syntax, called ECL for event contraint language, endowed with the well-established, sound timing interpretation as in CCSL. This work was reported in [40].

6.1.3. Logical time clocks to schedule data-flow models

Data-flow models can be used to capture data dependencies from applications, execution platforms and allocations. Most of the time such data dependencies impose only a partial order on the execution of application elements onto the execution platform and allow several allocation schemes. In [38], we have shown how to use logical time and CCSL constraints to capture explicitly the partial order imposed by the data-dependencies without imposing a total order. This work of representation expressivity then paved the way for analysis studies on time refinement, described in 6.3 .

6.1.4. Timing requirement modeling

One of the weak points of UML regarding a complete system design flow is its poor treatment of requirement capture (although this is partly corrected in the SysML profile). When requirements are made on timing aspects and logical time (as in our advocated approach), the relevant syntactic expressivity must be provided. We worked on the definition of a Domain-Specific Language (DSL for Timing Requirements engineering. The results were presented in [24], then applied to system specification in the context of the work described in section 6.6 .

6.2. Semantic translation of CCSL constraints into appropriate Büchi automata for trace recognition

Participants: Frédéric Mallet, Julien Deantoni, Robert de Simone, Ling Yin.

Our CCSL language expresses timing and scheduling constraints for a system, based on the notion of abstract logical clocks providing time events, and contraints linking them with relations of "asynchronous" nature (precedence, faster than) or of "synchronous" origin (subclocking, included in). Of course in a large system design both types coexist, and functional definitions also live next to declarative specifications to allow several timing solutions. Such a solution, called a schedule, must enforce that each logical clock either ticks endlessly, or terminates properly, in a way that globally respects the constraints. In previous works we have shown how a large variety of semantic scheduling constraints from the literature could soundly be represented in CCSL.

This year we focused on the semantic foundation of our CCSL language, by defining a structral operational semantic translation into a specific type of transition systems. Because we deal with infinite traces we had to consider acceptance mechanisms such as Büchi repeated states (as already used for translation of LTL temporal logic formulae in classical model-checking). Next we found out that, while state-labeled acceptance conditions were fine to obtain a direct and intuitive translation of individual constraints, building the composition of such models when dealing with multiple constraints was much easier in the case of *transition-labeled* Büchi automata (with repeated acceptance criteria now on transitions); the theory carries over to such case quite naturally, and has already been studied in the past. Finally, because traces must include infinite occurrences for *each* clock, we had to move to so-called *extended* Büchi automata, again a model already studied previously. We provided a complete semantic translation for all CCSL kernel constructs. Most importantly, we provided an efficient and simple fix-point algorithm to check the existance of a valid schedule, based on the type of automata just defined. This is (we believe) a genuine improvement on existing results, with potential applications outside our direct scope. These results are presented in a technical report, submitted for publication [46].

6.3. Timing refinement for multidimensional dataflow models using MARTE Time Model

Participants: Frédéric Mallet, Julien Deantoni, Jean-Vivien Millo.

Extensions of dataflow process networks have been proposed (as multidimensional SDF) to combine task parallelism (as in traditional process networks) with intensive data parallelism (as proposed in the Array-OL/Gaspard2 formalism developed in the DaRT EPI, for instance). The prospect of scheduling (seen as precise time cycle allocation) is here more complex, because of possible trade-offs between the granularity of treatments at task level *vs.* the size of data arrays that are handled uniformally in parallel inside each task. We considered how these phenomena could be represented (if not solved) inside the framework of MARTE Time Model and logical clocks, so as to handle such design issues in a well-defined MDE approach. Additionally, we used the MARTE platform description to specify how the previous models are refined through mapping allocation. The resulting modeling framework was presented in a journal article [19]. This work was conducted jointly with P. Boulet, from DaRT EPI, and C. Glitia, former DaRT PhD and Aoste postdoc student.

6.4. Process Network analysis

Participants: Robert de Simone, Jean-Vivien Millo.

6.4.1. K-periodic routing schemes for Network-on-Chip data traffic

This year we considered more specifically the issue of exploiting the predictable routing schemes of our KRG models, expressed as infinite binary words to indicate the sucessive branching directions at merge/select switch nodes, in order to encode data traffic patterns expanded at compile time, when mapping applications expressed under the form of dataflow process networks onto processor arrays in manycore architectures based on network-on-chip interconnects. To show the potential impact of such predicatble compile-time routing patterns, we stdudied as a typical example a fulll (all-to-all) broadcast algorithm on a mesh topology, connecting mode-less computation nodes as in the theory of cellular automata. This resulted in a precise recursive definition of routing patterns, which achieve an optimal data propagation (broadcast implemented as multicast), given the availability of actual links in the NoC topology. This result was presented at the Autamata'2012 conference [30], and an expanded version is available as technical report [44].

A wider view of the approach, and its potential benefits, are described in a technical report [43], submitted for publication.

6.4.2. Optimal data placement for process network scheduling

The topic of efficient scheduling of dataflow process network traffic to optimize both throughput and buffer queue sizing has given rise to a huge literature starting with seminal works in [49], [47], [56]. It has recently been given new impulse due to the advent of manycore architectures (see above). We conducted a number of theoretical works, to establish how such optimal computation scheduling can be best achieved in configurations where data are evenly distributed and streched in time across the (process) network. While this result is intuitively obvious, we formalized precisely what evenly distributed technically means, with the notion of balanced/mechanical words going a long way back in formal language theory, and we demonstrated that under such assumptions optimal schedules could be constructed *in a fully analytical way*, without any symbolic simulation steps or behavior expansion. The result was accepted for publication in a journal article [20].

6.5. Transformation from MARTE Time Model and CCSL to formal analysis models

Participants: Frédéric Mallet, Ling Yin.

This work was conducted in the context of an on-going collaboration with the Software Engineering Institute (SEI) of East Normal China University (ECNU) at Shanghai, which led altogether in part to the DAESD Associated-team, followed by a LIAMA joint project proposal recently submitted (HADES), and the co-supervision by Frédéric Mallet (together with Professor Jing Liu from ECNU) of the PhD thesis of Yin Ling. Yin Ling spent a one-year visit in our team, funded on a chinese governemental grant.

We studied the efficient and sound formal translation of a subset of CCSL contraints into the PROMELA/SPIN formalism, to benefit from model-checking formal analysis features in this environment. The translation is not completely direct, as synchronous simultaneity is not a native notion of PROMELA, and has to be encoded as atomicity. The motivating principles and translation details are provided in [42]. A similar attempt could be considered in the future, this time with the synchronous model-checker SMV, which allows coumpound instantaneous atomic behaviors.

Another line of research was initiated at ECNU to consider *logical continuous time*, while most of our current work considers only discrete time (while MARTE Time Model considers both). Considerations on *hybrid state diagrams* , inviting the expressive power of formal hierarchical hybrid automata models into the MDE design space of UML MARTE, were investigated in [27].

6.6. Use of MARTE Time Model and Logical Time in automotive design and AUTOSAR/TADL

Participants: Marie-Agnès Peraldi Frati, Julien Deantoni, Arda Goknil.

Precise timing constraint modeling and analysis [26], [33] is a key point for the correct development of automotive electronics. EAST-ADL and AUTOSAR has been adopted as standards in automotive industry. The timing model (TADL :Time augmented Description Language) of these standards raises different issues, mainly concerning the precise modeling of the multi clock characteristics of distributed systems together with parameterized timing expressions. In the ITEA TIMMO-2-USE project [35] 8.3.2.1 , we conducted a work [34], [35], on extending TADL with an explicit notion of multiple time bases for modeling the various temporal referentials used in an automotive design (clocks from different ECUs, motor position, etc). Additionally, timing constraints are augmented with parameters, which can be free at the highest abstraction level and then progressively defined during the design process. As a result, a symbolic timing expression in TADL2 is possibly made of a suitable set of arithmetic operators mixing symbolic identifiers (not necessarily set variables) and referring to different time bases. One typical use of this feature is to capture unknown configuration parameters for time budgeting; another one is to relate constraints in different time-bases to each other. Inherent to this work is also the study of the allowable ranges for symbolic values that are dictated by a set of constraints.

6.7. Multiview modeling and power intent in Systems-on-chip

Participants: Carlos Gomez Cardenas, Ameni Khecharem, Jean-François Le Tallec, Frédéric Mallet, Julien Deantoni, Robert de Simone.

6.7.1. High-level power management modeling

One of the concern of the UML MARTE profile is to allow non-functional property modeling, so that the same system bare description can be annotated in a number of views. In our case, combined with our logical time framework, such properties can be made as time-depending, inside potentially distinct views. We examplified this approach by dealing to a large extent with the example of low-power design and energy modeling in the case of Systems-on-Chip (SoC) in the mobile phone domain. Pure power/thermal modeling can be realized, based on the system global architecture, then made operational with the use of logical time controllers triggering power management functionalities.
Thermal/power simulation models are usually relying on continuous time. Therefore we considered the issue of *logical continuous* time, in an early attempt at combining simulation of continuous time power/thermal models with intrinsically discrete functional aspects. A prototype was realized in Scicos, as part of Ameni Khecharem master internship.
This work was conducted in the context of Carlos Gomez PhD thesis, and in collaboration with several partners inside the ANR HeLP project. It should be continued in the forthcoming PhD thesis of Ameni Khecharem, just started in the context of the follow-up ANR HOPE project, which will consider specific issues of hierachical power modeling and compositional power management (as an example of incremental multiview aspects).

6.7.2. IP-XACT

In this context of high-level power modeling and multiview concerns, we considered the emerging Accelera standard IP-XACT, made to provide easy-to-plug interfaces and Architecture Description Language (ADL) to allow simple assembly of hardware IP components into well-behaved SoCs. More specifically we provided means to annotate such interface with extra informations, directly borrowed from UML MARTE NFP properties, to handle power and thermal aspects. A number of model transformations back and forth between MARTE and (extended) IP-XACT were realized, and extraction of IP-XACT compliant interfaces from proprietary SystemC code describing the elementary IP component tehmselves has been defined and implemented as well.
This work was initiated as part of a project with STMicroelectronics, inside the nano2012 programme (ended 2011), and continued as part of the ANR HeLP collaboration. It resulted in the PhD thesis of Jean-François Le Tallec (who remained in the team for a couple of months later to complete the prototype implementation) [16].

6.8. Correct and efficient implementation of polychronous formalisms

Participants: Thomas Carle, Manel Djemal, Dumitru Potop Butucaru, Robert de Simone, Yves Sorel.

We extended our work on extending the AAA methodology for polychronous processes, by providing a better integration of clock analysis in the various phases of the implementation process (allocation, scheduling, pipelining, etc.). We also considered a wider range of implementation targets (time-triggered, MPSoC) and non-functional constraints (partitioning).

6.8.1. Time-Triggered Platform targets

Our first result this year concerns the automatic scheduling and code generation for time-triggered platforms. We extended our previous results in two significant ways. First, we designed a novel approach for specification of real-time features of time-triggered systems, with deadlines longer than periods; this allows a faithful representation of complex end-to-end flow requirements. Second, we provided new algorithms for off-line pipelined scheduling of these specifications onto partitioned time-triggered architectures *à la* ARINC 653; allocation of time slots/windows to partitions can be either complete or partially provided, or synthesized by our tool. Automatic allocation and scheduling onto multi-processor (distributed) systems with a global time base becomes feasible, taking into account communication costs. For single processors, we allow the generation of fully compliant ARINC653/APEX implementation code.

This work was mainly carried out inside the FUI Parsec 8.2.2.2 (which funds the PhD thesis of T. Carle) and P 8.2.2.1 projects, as well as a collaboration with ASTRIUM Space Transportation. First results are presented in a technical report, submitted for publication [39].

6.8.2. Multi-Processor System-on-Chip (MP-SoC) targets

Our second contribution concerns the automatic allocation and real-time scheduling over MPSoC (multi-processor on chip) architectures with NoC (network-on-chip) interconnect. One must take into account the specific 2D mesh network-on-chip topology, and synthesize the NoC routing patterns. This work provides operational execution support for the contributions described in 6.9 .

6.8.3. The LoPhT tool

Our recent work on extending the AAA methodology with better handling of execution conditions, with pipelining and pipelined scheduling, and with specific real-time scheduling and code generation techniques for time-triggered/partitioned and MPSoC platforms resulted in the development of a new scheduling and code generation toolbox, called LoPhT (for Logical to Physical Time Compiler).

6.9. Programmable On-Chip Networks

Participants: Thomas Carle, Manel Djemal, Dumitru Potop Butucaru, Robert de Simone, Zhen Zhang.

Modern computer architectures are increasingly relying on multi-processor systems-on-chip (MPSoCs), with data transfers between cores and memories managed by on-chip networks (NoC). This reflects in part a convergence between embedded, general-purpose PC, and high-performance computing (HPC) architecture designs.

Efficient compilation of applications onto MPSoCs remains largely an open problem, with the issue of best mapping of computation parts (threads, tasks,...) onto processing resources amply recognized, while the issue of best use of the interconnect NoC to route and transfer data still less commonly tackled. In the most general case, dynamic allocation of applications and channel virtualization can be guided by user-provided information under various forms, as in OpenMP, CUDA, OpenCL and so on. But then there is no clear guarantee of optimality, and first attempts by non-experts often show poor performances in the use of available computing power. Conversely there are consistent efforts, in the domains of embedded and HPC computing, aiming at automatic parallelization, compile-time mapping and scheduling optimization. They rely on the fact that applications are often known in advance, and deployed without disturbance from foreign applications, and without uncontrolled dynamic creation of tasks. Our contribution follows this “static application mapping” approach.

An optimal use of the NoC bandwidth should authorize data transfers to be realized according to (virtual) channels that are temporarily patterned to route data “just-in-time”. Previous works have identified the need for Quality of Service (QoS) in “some” data connections across the network (therefore borrowing notions from macroscopic networks, say internet and its protocols). But our experience with the AAA methodology strongly suggests that optimal NoC usage should result from a global optimization principle (embodied in a form of the AAA methodology), as opposed to a collection of local optimizations of individual connections. Indeed, various data flows with distinct sources and targets will nevertheless be highly concerted, both in time and space, like in a classical pipelined CPU, where the use of registers (replaced in our case with a complex NoC) is strongly synchronized with that of the functional units.

One main problem in applying such a global optimization approach is to provide the proper hardware infrastructures allowing the implementation of optimal computation and communication mappings and schedules. Our thesis is that optimal data transfer patterns should be encoded using simple programs configuring the router nodes (each router being then programmed to act its part in the global concerted computation and communication scheme).

We addressed this problem in the framework of our collaboration with the "Embedded Systems- on-Chips" department of the LIP6 laboratory, one of the main site of expertise for SoC/NoC design and Hardware/software codesign. This collaboration first materialized with the co-supervision of M. Djemal's PhD thesis. We concretely supported our proposed approach by extending the DSPIN 2D mesh network-on-chip (NoC) developed at UPMC- LIP6. In this NoC, we replace the fair arbitration modules of the NoC routers with static, micro-programmable modules that can enforce a given packet routing sequence, as specified by small programs. The design of such simple routing schemes can, for instance, be extracted from our results in section 6.4 .
We advocate the desired level of expressiveness/complexity for such simple configuration programs, and provide experimental data (cycle-accurate simulations) supporting our choices. We also wrote an architecture synthesis tool that allows simple architectural exploration of MPSoCs using the new DSPINPro NoC. First results in this direction have been presented in the DASIP 2012 conference, where our paper [23] has been short-listed for best paper award.

6.10. Uniprocessor Real-Time Scheduling

Participants: Laurent George, Mohamed Marouf, Daniel De Rauglaudre, Yves Sorel.

6.10.1. Combination of Non-Preemptive and Preemptive Tasks

We focused on fixed priority scheduling for a combination of non-preemptive strict periodic tasks in conjunction with preemptive sporadic tasks, that we extended to software fault tolerance [29]. We first investigated the transient phase for non-preemptive strict periodic tasks and we proved that its length is smaller than the transient phase for preemptive periodic tasks. Then, we determined the worst case scenario for preemptive sporadic tasks where the Worst Case Response Time (WCRT) can be obtained in the presence of strict periodic tasks. We proved that these release times belong only to the permanent phase of strict periodic tasks, and thus that the schedulability analysis for sporadic tasks can be restricted to the permanent phase. For preemptive sporadic tasks, we extended the classical necessary and sufficient schedulability condition based on the worst case response time computation to take into account non-preemptive strict periodic tasks. Finally, we considered software fault tolerance in the particular case where each primary strict periodic task has an alternate sporadic task which is released when the primary task fails. The schedulability analysis guarantees that even if all strict periodic tasks fail then all their respective alternate tasks will meet their deadlines.

6.10.2. Formal Proofs of Real-Time Scheduling Theorems

We completed two formal proofs of theorems in Coq on scheduling of fixed priority real-time preemptive tasks: one dealing with the sizes of busy periods (about 3500 lines of Coq), and another one dealing with response time (about 5200 lines of Coq). A monograph about these proofs, together with the formal check in Coq of scheduling conditions of strict periodicity, presented in the conference JFLA 2012 [37], have been started (currently about 70 pages).

6.11. Multiprocessor Real-Time Scheduling

Participants: Abderraouf Benyahia, Laurent George, Mohamed Marouf, Falou Ndoye, Simon Nivault, Yves Sorel, Cécile Stentzel, Meriem Zidouni.

6.11.1. Non-Preemptive Partitioned Fault Tolerant Scheduling

We addressed partitioned multiprocessor scheduling of non-preemptive strict periodic tasks which is extended thereafter to hardware fault tolerance [17].

In order to schedule a task set of non-preemptive strict periodic tasks on a multiprocessor platform, we partitioned this task set into subsets of tasks, each one is scheduled on a single processor using our proposed uniprocessor scheduling algorithm. The partition is carried out according to an enhanced "First Fit" algorithm that balances the load of the tasks on all the processors. However, inter-processors communications can lead to delay task execution. Thus, we determined the start time of each task taking into account the communication delay between this latter task and its predecessor tasks. Also, as inter-processor communications may generate a transient phase, we computed the length of the transient phase.

We proposed a fault tolerant real-time scheduling algorithm which allows hardware processors and/or buses faults, and conserves the strict periodicity of each task. We also proposed a graph transformation algorithm, applied on the task graph, which generates redundancies of tasks as well as dependencies. The transformation adds also selector tasks which choose data coming from the non failing processors and buses. That algorithm is based on exclusion relations to assign redundant tasks (resp. dependencies) to different processors (resp. busses). Then, we extended the previous partitioned multiprocessor scheduling algorithm to manage fault tolerance taking into account these exclusion relations.

This approach was successfully implemented on a CyCabs electric vehicle in a real-time fault tolerant tracking application where some processor or some bus could fail without any consequence on the proper execution of the application, i.e. same functional behaviour and real-time constraints satisfied.

6.11.2. Partitioned Scheduling with Exact Preemption Cost

Preemption allows a better scheduling success ratio but has a cost that must not be neglected in safety critical applications of domains such as avionic, automotive, etc. We focused on partitioned multiprocessor scheduling of independent preemptive periodic real-time tasks, while taking into account the exact preemption cost with the $\oplus$ operation formerly proposed by Meumeu and Sorel [10]. We improved the "greedy" heuristic proposed last year and compared it with the "Best-Fit" (BF) and "Worst-Fit" (WF) heuristics classically used in partitioned multiprocessor scheduling, but extended to take into account the exact preemption cost. We also compared our heuristic with an exact "Branch and Bound" algorithm with the same extension. The first comparison shows that the task allocation found by our heuristic gives a better response time than those found by WF and BF. This is due to the fact that the execution of the tasks is better parallelized. On the other hand, BF and WF heuristics execute a bit faster than our heuristic because they do not use all the available processors contrary to our heuristic which has the advantage to improve the load balancing of the tasks on all the processors.

Then, we addressed the scheduling of preemptive periodic real-time tasks with dependence constraints involving task precedences and data dependences. We considered harmonic tasks, i.e. periods of tasks are multiple or equal, to avoid loss of data. In order to satisfy data dependence constraints, we modified the release dates and deadlines of the dependent tasks according to the reception date of the data. In addition, data dependences between tasks mean to share data between dependent tasks which can cause deadlock and priority inversion problems. In order to solve these problems while taking into account the preemption cost, we proposed a new schedulability condition based on an extension of the $\oplus$ operation. We plan to propose a mutiprocessor scheduling heuristic based on that condition applied on tasks with modified release dates and deadlines.

6.11.3. Semi-partitioned Scheduling

Semi-partitioned multiprocessor scheduling stands between partitioned and global scheduling, the latter allowing migrations. We mainly addressed the semi-partitioned scheduling approach where the Worst Case Execution Time (WCET) of a job can be portioned, each portion being executed on a dedicated processor, according to a static pattern of migration. A job is migrated at its local deadline, computed from the deadline of the task it belongs to. We have studied this approach in the context of a fork/join task model with thread parallelism. A task is composed of a sequence of segments that can be parallelized in threads, if needed. The local deadlines depends on the number of parallel threads assigned to each segment.

6.11.4. Code Generation for Multicore

This work was carried out in the OPENPROD ITEA project 8.3.2.2 . xMod developed by IFPEN (IFP Energies Nouvelles), is an heterogeneous model integration environment that allows model importation from specific tools such as Simulink, AMSIM, etc. It also provides as a virtual instrumentation laboratory. In order to make xMod being able to run simulations with hardware-in-the-loop environment, we developed a new SynDEx executive kernel based on the kernel, dedicated to Windows/RTX, developed last year. That executive kernel is used with the macro-code generated by SynDEx to produce a real-time executable code that can drive the execution (real-time multi-core distribution and synchronized execution) of the models imported by xMod

and simulated in the virtual instrumentation laboratory. This prototype as well as the report describing the corresponding achieved works, are the final deliverable of the OPENPROD project.

Furthermore, a French and English SynDEx code generation reference manual has been written to help future SynDEx users and maintainers to generate real-time code for already supported architectures or new ones.

6.11.5. Gateway with Modeling Languages for Certified Code Generation

This work was carried out in the P FUI project 8.2.2.1 . We provide inside the project expertise mainly on schedulability analysis and automatic generation of distributed real-time code. In this context, we developed a gateway between UML/MARTE and SynDEx. From a model specified with UML (Activity Diagram to specify algorithms and Composite Structure Diagram to specify multicomponent architectures) and refined with the UML profile MARTE (Modeling and Analysis of Real-Time Embedded Systems), we use the gateway to generate automatically distributed real-time application specified in the SynDEx format. Currently, we intend to provide a gateway between the GeneAuto language and SynDEx. The GeneAuto language is a subset of the future pivot P language. We presently deal with the part of the GeneAuto language corresponding to Simulink for data-flow modeling and we plan to deal soon with the part corresponding to Stateflow for control-flow modeling (composition of automata).

6.11.6. SynDEx Updates

We continued the software developments for the future version 8 of SynDEx which will feature a new software architecture to allow better functionality evolutions and maintenance. On the other hand in the COTROS ADT ("Génération de code temps réel distribué optimisé et sûr"), we completed the tests on the new automatic code generator for the current version 7 of SynDEx. This new generator produces code for mono-periodic and multi-periodic applications with condition and repetitive control structures, for the different hardware architectures supported by SynDEx. We developed a checker for the generated code that was integrated in the new generator. This checker verifies the correct use of semaphores and consequently the absence of deadlocks in the real-time code. Deadlocks are the most difficult part when dealing with distributed architectures. We achieved also a maintenance report describing the structure and the main features of code generator, as well as the technical choices we did.

6.12. Variability of program execution times on multicore processors

Participants: Sid-Ahmed-Ali Touati, Matias Vara Larsen, Abdelhafid Mazouz.

The activity described here represents the finalization of previous efforts conducted by Sid Touati and members of his groups, initiated before he joined the AOSTE EPI, and which are progressively merged with our own objectives, for results to be reported hopefully next year).

With the massive introduction of multicore platforms on embedded systems, parallel applications gained in performance. However, we showed in previous studies that the performance gain comes with high instability: program execution times vary in important way. We investigated the reasons for this variations and tried to understand the factors that influence program performance variability, that we decompose intro multiple families: factors from the application itself (implemented algorithms, coding technique, synchronization barriers, etc.), factors from the execution environment (OS effects, thread scheduling, Input/Ouput operations) and factors from the underlying hardware (micro-architecture, memory hierarchy, speculative execution, hardware data prefectching, etc.). Now, we have better understanding to these factors thanks to the work of two students:

1. Mr. Abdelhafid Mazouz who defended his PhD under the direction of Sid Touati at the university of Versailles in 11th of December 2012. The title of his PhD is "An Empirical Study of Program Performance of OpenMP Applications on Multicore Platforms".
2. Mr. Matias Vara Larsen, intern under the supervision of Sid Touati from February to June 2012, inside the Aoste EPI in Sophia-Antipolis, co-funded under a grant from Inria international internship program). The topic of his internship was to study the influence of he Linux kernels (multiple versions) on the stability of parallel applications.

Last, we published a rigorous statistical protocol in [21] called the Speedup-Test. It is used to analyze valid speedups (performance gain) in presence of performance instability: The Speedup-Test protocol is implemented and distributed as an open source tool based on R software. Our statistical methodology defines a consistent improvement compared with the usual performance analysis method in high-performance computing.

CONVECS Team

6. New Results

6.1. New Formal Languages and their Concurrent Implementations

LNT is a next generation formal description language for asynchronous concurrent systems, which attempts to combine the best features of imperative programming languages and value-passing process algebras. LNT is increasingly used by the CONVECS team for industrial case studies and applications (see § 6.5) and serves also in university courses on concurrency, in particular at ENSIMAG (Grenoble) and at the Saarland University.

6.1.1. Translation from LNT to LOTOS

Participants: Hubert Garavel, Frédéric Lang, Wendelin Serwe.

The LNT2LOTOS, LNT.OPEN, and LPP tools convert LNT code to LOTOS, thus allowing the use of CADP to verify LNT descriptions. These tools have been used successfully for many different systems (see § 6.5 and § 9.1).

In 2012, in addition to 12 bug fixes, the following enhancements have been brought to these tools:

- We improved the ergonomy of the LNT2LOTOS translator by refining certain command-line options and by making some warning messages more user-friendly.
- We optimized the generated LOTOS code of the "disrupt" and "parallel" composition operators, so as to reduce the number of spurious warnings about impossible synchronizations and, more importantly, to meet the subset of LOTOS supported by the CAESAR compiler (static bound on the number of parallel processes).
- We improved the support for LNT programs that contain several modules by allowing the main process to be defined in any module (not only in the main module).
- We added new predefined functions for the generic data types (lists, sorted lists, and sets), and we updated accordingly the reference manual of LNT. The set types are now implemented correctly by avoiding duplicate elements.

6.1.2. Distributed Code Generation for Process Algebras

Participants: Hugues Evrard, Frédéric Lang.

One goal of CONVECS is to build a tool that generates automatically a distributed implementation of a system specified in LNT. This requires a protocol to realize process synchronization. As far as possible, this protocol must itself be distributed, so as to avoid the bottleneck that would inevitably arise if a unique process would have to manage all synchronizations in the system. A particularity of such a protocol is its ability to support *branching synchronizations*, corresponding to situations where a process may offer a choice of synchronizing actions (which themselves may nondeterministically involve several sets of synchronizing processes) instead of a single one. Therefore, a classical barrier protocol is not sufficient and a more elaborate synchronization protocol is needed.

In 2012, we explored the bibliography on synchronization protocols. Among almost twenty references studied, we selected three existing distributed synchronization protocols that seemed appropriate to our problem. In order to validate these protocols, we designed a tool chain that, given a system described as a parallel composition of LNT processes, generates an LNT specification of an implementation of the system (called the *implementation model*), by incorporating the protocol in the specification to realize the synchronizations. We then used CADP to check for livelocks and deadlocks possibly introduced in the implementation model by the protocol (using MCL and EVALUATOR 4.0), and to verify that the implementation model mimicks the behaviour of the system by equivalence checking (using BISIMULATOR).

Among the three protocols considered, we selected the most promising one [57], which is suitable for generalization to implement synchronization vectors (and hence, the generalized parallel composition operator of LNT). Using the methodology mentioned above, we discovered a previously unknown error in this protocol, which leads to deadlocks in certain situations, and we proposed a correction. An article has been submitted to an international conference.

6.1.3. Translation from an Applied Pi-Calculus to LNT

Participants: Radu Mateescu, Gwen Salaün.

The π-calculus is a process algebra defined by Milner, Parrow, and Walker two decades ago for describing concurrent mobile processes. So far, only a few verification tools have been designed for analyzing π-calculus specifications automatically. Our objective is to provide analysis features for π-calculus specifications by reusing the verification technology already available for value-passing process algebras without mobility. Our approach is based on a novel translation from the finite control fragment of π-calculus to LNT. To the best of our knowledge, this is the first π-calculus translation having a standard process algebra as target language.

In this work, we have also extended the original polyadic π-calculus with data-handling features. This results in a general-purpose applied π-calculus, which offers a good level of expressiveness for specifying mobile concurrent systems, and therefore for widening its possible application domains. As language for describing data types and functions, a natural choice was LNT itself: in this way, the data types and functions used in the π-calculus specification can be directly imported into the LNT code produced by translation.

The translation is fully automated by the tool PIC2LNT 2.0. This enables the analysis of applied π-calculus specifications using all verification tools of CADP, in particular the EVALUATOR 4.0 on-the-fly model checker, which evaluates temporal properties involving channel names and data values. PIC2LNT 2.0 was used for teaching mobile concurrency at Saarland University. A paper describing this work was accepted for publication in an international conference [16].

6.1.4. Translation from EB3 to LNT

Participants: Frédéric Lang, Radu Mateescu.

In collaboration with Dimitris Vekris (University Paris-Est Créteil), we have considered a translation from the EB3 language [39] for information systems to LNT. EB3 is inspired from a process algebra, but has the particularity to contain so-called *attribute functions*, whose semantics depend on the history of events. Therefore, the history of events becomes part of the state of an EB3 specification, which is unusual in process algebras.

Since EB3 is not equipped with native verification tools, we have proposed a translation from EB3 to LNT, which would enable EB3 specifications to be formally verified using CADP. Our formal translation scheme ensures the strong equivalence between the LTS corresponding to an EB3 specification and the LTS corresponding to the LNT code generated. The history of events is encoded as a particular LNT process "*memory*" synchronized on all EB3 events with the rest of the system. The memory process thus acts as a monitor that changes its state according to the occurring events and answers requests emitted by the attribute functions when needed. A prototype translator has been developed at University Paris-Est Créteil and a paper describing this work has been submitted to an international conference.

6.1.5. Coverage Analysis for LNT

Participants: Gwen Salaün, Lina Ye.

In the classic verification setting, we have an LNT specification of a system, a set of temporal properties to be verified on the LTS model corresponding to the LNT specification, and a data set of examples (test cases) we use for validation purposes. At this stage, building the set of validation examples and debugging the specification is a complicated task, in particular for non-experts.

Coverage analysis aims at proposing and developing techniques for automatically detecting parts of an LNT specification not (yet) covered during verification. Such LNT coverage analysis techniques would be very helpful for (i) extending the set of test cases with new inputs covering parts of the LNT specification that have not been analyzed yet, (ii) eliminating dead code in the LNT specification, and (iii) extending the set of temporal properties with new ones.

We have already identified four criteria (action, decision, block, property) and developed a prototype tool that automatically returns coverage values for these four criteria. We have applied our tool to LNT specifications of existing protocols, such as a reconfiguration protocol for component-based architectures [34], and found several cases of dead code and missing test cases.

6.1.6. Other Software Developments

Participants: Hubert Garavel, Frédéric Lang, Wendelin Serwe.

In addition of correcting 23 bugs in various CADP tools, we also brought the following enhancements:

- The EUCALYPTUS interface was improved regarding ergonomy and customization.
- The CADP tools for 32-bit and 64-bit Intel/Linux architectures were upgraded to use recent compilers and libraries, and CADP was modified to support Mac OS X 10.8 "Mountain Lion".
- The usability of the libraries for writing BCG files was improved to detect and signal an improper ordering of the primitives in application programs.
- The SYNTAX parser generator was improved by correcting two subtle errors, one of them causing an infinite looping on certain erroneous input programs. The CADP compilers developed using SYNTAX were enhanced to perform a better diagnosis of the situations when SYNTAX corrected syntactic errors automatically in erroneous programs.
- We improved an optimization of the CAESAR compiler for LOTOS, leading to a significant reduction of the execution time (from one hour and 51 minutes down to 58 seconds) for some examples of LOTOS programs with many variables. We optimized the CAESAR.OPEN script to invoke the CAESAR compiler directly whenever possible (instead of the GENERATOR tool), which improves the performance of graph generation, in particular for LNT.OPEN.
- Four demonstration examples of CADP were extended with LNT descriptions to illustrate the usage of the LNT language and of its compiler. Two examples have been simplified using the latest features of SVL, which can now handle the "n among m" parallel composition operator of LNT. Also, three examples have been reorganized for a better clarity and two couples of examples, which were closely related, have been merged into single examples.

6.2. Parallel and Distributed Verification

Participants: Hubert Garavel, Radu Mateescu, Wendelin Serwe.

For distributed verification, CADP provides the PBG format, which implements the theoretical concept of *Partitioned LTS* introduced in [46] and provides a unified access to an LTS distributed over a set of remote machines. The PBG format is equipped with the DISTRIBUTOR and PBG_MERGE (previously called BCG_MERGE [45]) tools, which perform the distributed generation of a partitioned LTS and the conversion of a partitioned LTS represented in the PBG format into a monolithic LTS stored in a BCG file.

To facilitate the manipulation of partitioned LTSs, CADP provides the PBG_CP, PBG_MV, and PBG_RM tools for copying, moving, and removing PBG files, maintaining consistency during these operations. The PBG_INFO tool provides several functionalities to inspect PBG files, such as checking consistency (i.e., existence and readability of all fragment files), calculating the size (number of states and transitions) of the corresponding LTS, displaying the list of labels, and concatenating remote log files (this is useful, e.g., to understand the reason why a PBG generation fails, and to compute global statistics about CPU and memory usage by the worker processes).

In 2012, in addition to correcting two bugs in DISTRIBUTOR and several bugs in the CAESAR_NETWORK_1 communication library used by the distributed verification tools, we also improved these tools as follows:

- We enhanced DISTRIBUTOR to support more than 256 distributed processes.
- We enhanced CAESAR_NETWORK_1 with a debugging facility, which enables traces of all distributed processes to be generated.
- We enhanced the graphical monitor of DISTRIBUTOR with the option of sorting the labels alphabetically, which facilitates their visual inspection.
- We extended PBG_INFO to enable the display of all labels in a partitioned LTS.

We also developed a prototype tool, named PBG_OPEN, which is an OPEN/CAESAR-compliant compiler for the PBG format, enabling the use of all CADP on-the-fly verification tools on a partitioned LTS. The main advantage of PBG_OPEN is that it can use the memory of several machines to store the transition relation of a partitioned LTS. Therefore, PBG_OPEN can explore on-the-fly large partitioned LTSs that could not be explored using other tool combinations. To reduce the amount of communications, PBG_OPEN can use a cache to store already encountered states, together with their outgoing transitions.

We experimented all these tools on the Grid'5000 computing infrastructure [35] using up to 512 distributed processes. These experiments confirmed the good scalability of our distributed LTS manipulation approach. A paper describing this work has been published in an international conference [12].

6.3. Timed, Probabilistic, and Stochastic Extensions

Participant: Hubert Garavel.

Process calculi provide a suitable formal framework for describing and analyzing concurrent systems, but need to be extended to model refined aspects of these systems. For instance, it may be necessary to represent probabilistic choices (in addition to deterministic and nondeterministic choices) as well as delays and latencies governed by probability laws. Many such extensions have been proposed in the literature, some of which have been implemented in software tools and applied to nontrivial problems. In particular, two of these extensions (namely, *Interactive Markov Chains* and *Interactive Probabilistic Chains*) are implemented in CADP. Despite these achievements, the state of the art is not satisfactory as the extended languages primarily focus on the probabilistic and stochastic aspects, leaving away the expressive and user-friendly features that process calculi provide for describing conventional concurrent systems.

In 2012, we undertook a study to merge probabilistic and stochastic aspects into modern high-level languages such as LNT. This work is done at Saarland University under the aegis of the Alexander von Humboldt foundation, in collaboration also with RWTH Aachen and Oxford University. We investigated the theoretical concepts, as well as their integration into modeling languages, together with the corresponding behavioural equivalences and temporal logics.

We also started experimenting with state-of-the-art software implementations, such as MODEST and PASS (Saarland University), COMPASS and MRMC (RWTH Aachen), and PRISM (Oxford University). Two of these tools (namely, MODEST and PRISM) have been used for lab exercises in the *Applied Concurrency Theory* block course created by H. Garavel at Saarland University. Following these experiments, evaluation reports have been produced, which provide feedback about issues and suggestions for enhancements. These reports have been addressed to the respective authors of each tool and already led to improvements in certain tools.

6.4. Component-Based Architectures for On-the-Fly Verification

6.4.1. Compositional Model Checking

Participants: Frédéric Lang, Radu Mateescu.

We have continued our work on *partial model checking* following the approach proposed in [29]. Given a temporal logic formula φ to be evaluated on a set S of concurrent processes, partial model checking consists in transforming φ into another equivalent formula φ' to be evaluated on a subset of S. Formula φ' is constructed incrementally by choosing one process P in S and incorporating into φ the behavioral information corresponding to P – an operation called *quotienting*. Simplifications must be applied at each step, so as to maintain formulas at a tractable size.

In 2012, we have continued the development of our prototype tools for partial model checking of the regular alternation-free μ-calculus supporting all features of the input language of EXP.OPEN 2.1. We have also extended our work to handle useful fairness operators of alternation depth 2 in linear time, without developing the complex machinery that would be necessary to evaluate general μ-calculus formulas of alternation depth 2. A paper has been published in an international conference [15] and an extended version has been submitted to an international journal.

6.4.2. On-the-Fly Test Generation

Participants: Radu Mateescu, Wendelin Serwe.

In the context of the collaboration with STMicroelectronics (see § 6.5.1 and § 7.1), we studied techniques for testing if a (hardware) implementation is conform to a formal model written in LNT. Our approach is inspired by the theory of conformance testing [59], as implemented for instance in TGV [51] and JTorX [33].

We developed two prototype tools supporting conformance testing. The first tool implements a dedicated OPEN/CAESAR-compliant compiler for the particular asymmetric synchronous product of the model and the test purpose. This tool is a generic component for on-the-fly graph manipulation, taking as input two graphs and producing as output the graph of the asymmetric synchronous product. The second tool generates the complete test graph, which can be used to extract concrete test cases or to drive the test of the implementation. This tool was built from (slightly extended) existing generic components for on-the-fly graph manipulation (τ-compression and τ-confluence reductions, determinization, resolution of Boolean equation systems). The main advantage of our approach compared to existing tools is the use of LNT for test purposes, which facilitates the manipulation of data values.

6.5. Real-Life Applications and Case Studies

6.5.1. ACE Cache Coherency Protocol

Participants: Hubert Garavel, Abderahman Kriouile, Radu Mateescu, Wendelin Serwe.

In the context of a CIFRE convention with STMicroelectronics (see § 7.1), we studied the system-level cache coherency, a major challenge faced in the current system-on-chip architectures. Because of their increasing complexity (mainly due to the significant number of computing units), the validation effort using current simulation-based validation techniques grows exponentially. As an alternative, we study formal verification.

In 2012, we focused on the ACE (*AXI Coherency Extensions*) cache coherency protocol, a system-level coherency protocol proposed by ARM [25]. In a first step, we developed a formal LNT model (about 2600 lines of LNT) of a system consisting of an ACE-compliant cache coherent interconnect, processors, and a main memory. The model is parametric and can be instantiated with different configurations (number of processors, number of cache lines, number of memory lines) and different sets of supported elementary ACE operations, including an abstract operation that represents any other ACE operation. Using the OCIS simulator, we were able to explore the behavior of the system interactively, which has been found helpful by STMicroelectronics engineers.

Currently, our formal model supports a representative subset of five elementary operations of the ACE protocol (MakeUnique, ReadOnce, ReadShared, ReadUnique, and WriteBack). For each of these operations, we have written a liveness property in MCL expressing that the operation is executed until its termination. Using parametric SVL scripts (about 250 lines) and the EVALUATOR 4.0 model checker, we verified these properties on the fly for up to three memory lines and two processors with two cache lines each. We also generated the corresponding LTS (up to 250 million states and one billion transitions).

We also started considering data integrity properties. This required to translate a state-based property (namely, the consistency between the values stored in memory and in the local caches of the processors) into our action-based setting. This enabled us to automatically exhibit a known error present in a previous version of the ARM specification of the ACE protocol (which was corrected in a subsequent version of the specification). Using the LNT model corresponding to the latest version of the ACE specification, we spotted several potential data integrity issues that we reported to STMicroelectronics, where they are currently under investigation.

6.5.2. Realizability of Choreographies

Participants: Alexandre Dumont, Matthias Güdemann, Gwen Salaün.

Choreographies allow business and service architects to specify, with a global perspective, the requirements of applications built over distributed and interacting software entities. In collaboration with Pascal Poizat (University Paris-Sud), we proposed new techniques for verifying BPMN 2.0 choreographies, and particularly the *realizability* property. Realizability ensures that peers obtained via projection from a choreography interact as prescribed in the choreography requirements. Our approach is formally grounded on a model transformation into the LNT process algebra and the use of equivalence checking. It is also completely tool-supported through interaction with the Eclipse BPMN2 modeler and CADP. These results have been published in an international conference [17].

In collaboration with Meriem Ouederni (University of Nantes), we extended our techniques for analyzing choreographies to restore realizability for non-realizable, but *repairable* choreographies. For this we exploit the counterexamples generated by the equivalence checker BISIMULATOR to identify problematic messages in the choreography. For those messages we add distributed local monitors to the system which delay message sending if necessary, to restore correct message sequences. This iterative approach introduces the minimal number of necessary additional messages to restore realizability, and the monitors are generated in the most permissive way, i.e., by considering all possible interleavings given the behaviour of the peers participating to the choreography. It is fully automated by a prototype tool we implemented. These results have been published in an international conference [14].

We developed a common formal description language, named CIF (*Choreography Intermediate Format*), for the verification of choreographies. CIF is based on an XML representation for easy exchange between programs, an XSD schema for validation, and a translation to LNT for verification. CIF is used as an intermediate language to specify choreographies, but can also serve as target language for translating various choreography specification languages, such as BPMN 2.0. The back-end connection to CADP via LNT enables the automation of some key choreography analysis tasks (repairability, realizability, conformance, etc.). Our framework is extensible with other front-end and back-end connections to, respectively, other choreography languages and formal verification tools.

6.5.3. Self-Configuration Protocol for Distributed Cloud Applications

Participants: Rim Abid, Gwen Salaün.

We collaborate with Noël de Palma and Fabienne Boyer (University Joseph Fourier), Xavier Etchevers and Thierry Coupaye (Orange Labs) in the field of cloud computing applications, which are complex, distributed artifacts involving multiple software components running on separate virtual machines. Setting up, (re)configuring, and monitoring these applications are complicated tasks because a software application may depend on several remote software and virtual machine configurations. These management tasks involve many complex protocols, which fully automate these tasks while preserving application consistency as well as some key properties.

In this work, we focus on a self-configuration protocol, which is able to configure a whole distributed application without requiring any centralized server. The high degree of parallelism involved in this protocol makes its design complicated and error-prone. In order to check that this protocol works as expected, we specify it in LNT and verify it using the CADP toolbox. The use of these formal techniques and tools helped to detect a bug in the protocol, and served as a workbench to experiment with several possible communication models. These results led to a publication in an international conference [18].

We are currently studying two variants of the self-configuration protocol, one handling virtual machine failures, and one allowing dynamicity in the system (addition and removal of virtual machines) using a publish-subscribe communication framework.

6.5.4. Networks of Programmable Logic Controllers

Participants: Hubert Garavel, Fatma Jebali, Jingyan Jourdan-Lu, Frédéric Lang, Eric Léo, Radu Mateescu.

In the context of the Bluesky project (see § 8.1.2.1), we study the software applications embedded on the PLCs (*Programmable Logic Controllers*) manufactured by Crouzet Automatismes. One of the objectives of Bluesky is to enable the rigorous design of complex control applications running on several PLCs connected by a network. Such applications are instances of GALS (*Globally Asynchronous, Locally Synchronous*) systems composed of several synchronous automata embedded on individual PLCs, which interact asynchronously by exchanging messages. A formal analysis of these systems can be naturally achieved by using the formal languages and verification techniques developed in the field of asynchronous concurrency.

For describing the applications embedded on individual PLCs, Crouzet provides a dataflow language with graphical syntax and synchronous semantics, equipped with an ergonomic user interface that facilitates the learning and use of the language by non-experts. To equip the PLC language of Crouzet with functionalities for automated verification, the solution adopted in Bluesky was to translate it into a pivot language (to be defined within the project) that will enable the connection to testing and verification tools covering the synchronous and asynchronous aspects. Our work focuses on the translation from the pivot language to LNT, which will provide a direct connection to all verification functionalities of CADP, namely model checking and equivalence checking.

In 2012, in interaction with Crouzet engineers, we studied the PLC language of Crouzet to understand precisely its static and dynamic semantics. We specified manually in LNT several examples of control applications provided by Crouzet, with the goal of identifying the principles of translating the PLC language of Crouzet to LNT. We formulated in MCL several safety and liveness properties concerning the temporal ordering of input and output events by the control applications, and we successfully verified them on the LNT specifications. We also started to study the network communication mechanisms between PLCs to identify a suitable LNT abstraction of the communication layer.

6.5.5. Other Case Studies

Participants: Frédéric Lang, Radu Mateescu, Wendelin Serwe.

Continuing a study [53] started in the context of the Multival project (see http://vasy.inria.fr/multival), we considered the Platform 2012 architecture proposed by STMicroelectronics, focusing on the Dynamic Task Dispatcher (DTD), a hardware block that assigns a set of application tasks to a set of processors. In 2012, we extended our LNT model and the corresponding MCL properties in order to handle heterogeneous processors equipped with different kinds of processor extensions. We also used constraints on the initialization phase, which reduced the size of the LTS by a factor of up to ten and hence enabled the generation of the LTS for up to eight processors (instead of only six). Both extensions together enabled to discover the possibility of a livelock.

We attempted to investigate this issue further by cosimulation (using the EXEC/CAESAR framework) with the original C++ model of the architect. Unfortunately, the C++ model did not behave correctly for the particular aforementioned application scenario. It was not possible to change this model because the recent evolutions of Platform 2012 excluded the DTD, as its requirements in terms of silicon surface were considered too large. This work, including the LNT model as appendix, has been accepted for publication in an international journal [5].

In collaboration with Nuno Mendes and Claudine Chaouiya (Gulbenkian Institute, Portugal), Yves-Stan Le Cornec (IBISC, University Evry Val d'Essonne) and Grégory Batt (CONTRAINTES project-team, Inria Paris-Rocquencourt), we have studied the use of CADP for checking the reachability of stable states in genetic regulatory networks. A compositional and logical model of genetic regulatory networks called *logical regulatory modules* was defined and translated to LNT processes and EXP.OPEN 2.1 networks of LTSs.

Compositional minimization modulo safety equivalence was applied to the generated network, so as to palliate state explosion while preserving the reachability property. The approach has been illustrated on the segment polarity module involved in the segmentation of the fruit fly embryo and on the delta-notch module involved in cell differentiation in crucial steps of embryonic development of several species. A paper has been submitted to an international journal.

DART Project-Team

6. New Results

6.1. Hardware Distributed Control for Dynamic Reconfigurable Systems

The progress in FPGA technology has allowed FPGA-based reconfigurable embedded systems to target increasingly sophisticated applications, which leads to a high design complexity of such systems especially at the adaptation control level. This complexity results into long design phases and delayed time-to-market. In this context, a centralized control model might be not adapted to the growing size and complexity of embedded systems. The use of a single controller for the whole system might result into a high complexity due to the number of parameters to take into account for runtime adaptation, which makes difficult its modification and test. Besides, the design of such a controller is system-dependent since it treats the system as a whole, which represents an obstacle for design reuse. In order to solve these problems, we propose a control design approach aiming to decrease design complexity and enhance design flexibility, reuse and productivity. This approach is based on a semi-distributed control model [34]. In order to achieve the objectives mentioned above, the proposed approach combines autonomy, modularity, formalism and high-level design. The semi-distributed control model divides the control problem between autonomous controllers handling each the self-adaptation of a reconfigurable component of the system, which allows to decrease their design complexity. Each controller handles three main tasks allocated to three different modules: i)monitoring of events that might trigger the adaptation of the controlled component, ii)decision-making about the required adaptations, and iii)adaptation (reconfiguration) realization. To ensure that reconfiguration decisions made by the controllers respect global system constraints such as security and quality of service constraints, these decisions are coordinated before launching the corresponding partial reconfigurations. The allocation of these tasks to separate modules facilitates their modification and reuse and thus the scalability of the control design. For the decision-making modeling, we use the mode-automata formalism. This formalism is suitable to model the control of the different modes of a reconfigurable system such as energy modes or image display modes. Thanks to its clear semantics, the use of such a formalism facilitates the high-level modeling of the controllers and their automatic generation. In order to facilitate code generation and enhance thus design productivity, our control approach makes use of Model-Driven-Engineering (MDE) [33]. Control systems composed of controllers and coordinators are modeled using the UML (Unified Modeling Language) profile MARTE (Modeling and Analysis of Real-Time and Embedded systems). The automation of MDE, allowed to generate the code of these systems. The generated code was then used to validate the semi-distributed control and to determine its resource overhead compared to centralized control systems.

6.2. Regular interconnection network for HP-SoC architecture

Our Synchronous Communication Asynchronous Computation (SCAC) model is a data-parallel execution model dedicated to the High Performance System-on-Chip. The architecture of this model is composed of huge number of complex routers, called node elements (the NEs), communicating and working in perfect synchronizations. Each NE is potentially connected to its neighbors via a regular connection. Furthermore, each NE is connected to a heterogeneous set of computing groups (clusters) allow asynchronous processing. Each group includes a combination of processors programmable, the PEs (software processing units) and specialized hardware accelerators (hardware processing units) to perform critical tasks demanding the more performance. All the system is controlled by a Network Controller Unit, the NCU. The NCU and The PEs are implemented with the Forth processor.

The synchronous communication in SCAC model is presented by two kinds of communications:

- The NCU/NEs communication. In fact, we defined a hNoC model integrated in the SCAC architecture [31]. This model is based on sub-netting the network of processing nodes which separate the control of communication and processing. From this model, our communication system allows a better management of data congestion in the NEs grid through the broadcast with mask of parallel instructions to activated processing nodes.
- The NE/NE communication which is our last contribution. In fact, we defined the X-net interconnection network which is a regular network dedicated to the massively parallel SCAC architecture. This network interconnects directly each PE with its 8 nearest neighbors in a two-dimensional mesh through a specific router in the NE module.

The aim of these last works is to design a regular NoC for SCAC architecture to allow global synchronization of the system communications and increase high performance in terms of area cost and bandwidth. This network based on IP blocks which offer well flexibility and scalability, was implemented in synthesizable VHDL code that was simulated and targeted Xilinx Virtex6 (XC6VLX240T) board. The difficulty of designing X-net is a compromise between an optimal quality of broadcasting, high bandwidth and important flexibility of use, while reducing power consumption and silicon area.

6.3. ReCoMARTE: A Marte Based Profile for Dynamic Reconfigurable Systems Modeling

During the last decade, DPR has been widely studied as a research topic. Despite its intuitive appeal, the technique had eluded widespread adoption, particularly in industrial applications. This is due to the complexities of the provided design flow and the in-depth knowledge of many low level aspects of FPGA technologies used to implement DPR systems. The aim of our current work is to propose a methodology in order to allow us to introduce PDR in MARTE for modeling all types of FPGAs supporting our chosen PDR flow. Afterwards, using the MDE model transformations, the design flow can be used to bridge the gap between high level specifications and low implementation details to finally generate files used by the Xilinx EDK design flow for implementing the top-level SoC description of the system. Indeed, in its current version, UML MARTE profile lacks dynamic reconfiguration concepts and requirements for the reconfiguration at different abstraction levels. We have concentrated our efforts in the creation of the structural description of the system that is used as an input to the DPR design flow to facilitate the design entry phase of the DPR design flow. Therefore, we defined an extended version of MARTE called RecoMARTE (Reconfigurable MARTE) [16] model these concepts mainly at:

- Application level: For reconfigurable applications combining control and data processing, it is very difficult, even impossible to use the MARTE profile for their specification. Non-functional properties such as control concepts are induced by different configurations or running modes of the system and allow the description of more complex behaviours. We recommend a set of extensions to a MARTE profile. We also focus on modelling heterogeneous reconfigurable components, and address the problem of constraints specification for verification issue.
- Control mechanism: We define necessary requirements for the reconfiguration control mechanism in order to manage reconfiguration at every design level. In addition, our solution allows to describe global contracts and constraints for combining automata. Our modeled reconfiguration controller will be then synthesized using Discrete Controller Synthesis formal technique (collaboration work)I n order to always provide a correct configuration to the system, with respect to constraints specified by the designer
- Deployment level: Our design methodology using RecoMARTE enables the deployment, parameterization and integration of hardware IPs into SoC platform at multiple levels of abstraction. We have introduced IP deployment capabilities in MARTE, which aim at facilitating the import of selected low-level features into the high-level models, their modification, and the creation of an IP-XACT design description that is used to parameterize and integrate the underlying IP descriptions.

- Physical level: introduced extensions in MARTE provide some facilities to allow modeling physical architecture of a chosen FPGA. Our solution allows to carry out the physical placement of static and reconfigurable areas on the platform. This task is done through ranges in terms of physical resources, with respect to placement constraints such as consumed resources.

6.4. Using Marte Profile for NoCs modeling

The modeling of repetitive structures such as network on chip topologies in graphics forms poses a particular challenge. This aspect may be encountered in intensive data/control oriented applications such as H.264 video coder. In this work we have described an adequate methodology for modeling NoCs by using the MARTE standard profile. The proposed study has shown that the Repetitive Structure Modeling (RSM) package of MARTE profile is powerful enough for modeling different topologies. By using this methodology, several aspects such as routing algorithm are modeled based finite state machines. This allows to the MARTE profile to be complete enough for modeling a large number of NoCs architectures. Some work is on-going to synthesize such networks in VHDL from such models [55]. While validating the proposed methodology, a co-design approach has been studied by mapping a H264 video coding system onto a Diagonal Mesh NoC by using the Y Chart of Gaspard2 tool. Before allowing the association of the application/architecture, an architectural optimization targeting power minimization of the most critical module of the application and the router of the architecture has been performed. For instance, a flexible VLSI architecture for full-search VBSME (FSVBSME) has been proposed.

6.5. A Hardware Membranes Based Reconfiguration Services Implementation

Partial and dynamic reconfiguration provides a relevant new dimension to design efficient parallel embedded systems. However, due to the encasing complexity of such systems, ensuring the consistency and parallelism management at runtime is still a key challenge. So architecture models and design methodology are required to allow for efficient component reuse and hardware reconfiguration management. We proposed a novel approach inspired from the well-known component based models used in software applications development. Our model is based on membranes wrapping the systems components. The objective is to improve design productivity and ensure consistency by managing context switching and storage using modular distributed hardware controllers. These membranes are distributed and optimized with the aim to design self-adaptive systems by allowing dynamic changes in parallelism degree and contexts migration [26]. These results are obtained in the Famous project by a collaboration with LABSticc Lorient.

6.6. Formal Techniques for General and Domain-Specific Languages

In 2012 we have finished the previous year's activities on domain-specific languages based on formal model-driven engineering with two papers [18], [24]. Our conclusion is that formal MDE-based language definition is interesting because of its generality but adds extra layers of complexity due to the fact that language concepts and semantics are only formalised indirectly, through the formalisation of MDE concepts used in language definition. We have decided thus to move on towards more direct ways of defining and reasoning about languages. We have been experimenting with the K framework [4] for formally defining both the assembly language and a higher-level language for programming on the upcoming dynamically reconfigurable hardware architecture that our team is developing.

We have also worked on proving the correctness of a compiler between high-level and assembly language, based on new symbolic program-equivalence proof techniques we are developping in collaboration with the K team [29]. We have also been working on generic symbolic execution techniques for programming languages having term-rewriting based semantics [28] (PhD of Andrei Arusoaie, supervised in collaboration with Prof. Dorel Lucanu from the K team of Univ. Iasi (Romania).

[4] http://www.k-framework.org

ESPRESSO Project-Team

6. New Results

6.1. Extensions of the Signal language and the Polychrony formal model

Participants: Thierry Gautier, Paul Le Guernic.

The different works related to the use of the polychronous model as semantic median model (which has also a syntactic instance) for different effective models (AADL [15], Simulink via GeneAuto, UML via CCSL...) lead us to study various possible extensions of the semantic model as well as the syntactic one.

Thus, we are defining a new version, V5, of Signal, that will be a deep evolution from the current V4 version.

In particular, we are investigating the way state diagrams are best represented in the polychronous model of computation, maintaining the multi-clock characteristic property of the representation. We propose a semantic model for these automata, that relies on the Boolean algebra of clocks. A special case of automata is those that may be represented as regular clock expressions, for which we develop a specific formal calculus. These regular expressions may be used as a powerful manner to express regular dynamic properties of polychronous processes. In correspondence with these models, we are defining syntactic structures to represent these Signal State Diagrams.

Moreover, an important challenge we want to address in the next few years is that of providing design automation techniques and tools for engineering heterogeneous cyber-physical systems (CPS). This leads in particular to new requirements related to the language itself in which we want to describe such software architectures. With respect to the current V4 version of Signal, the basic idea is to extend Signal with a syntactic structure that encapsulates a polychronous process P in a *system*, S, that could have a continuous temporal domain providing a real-time clock presented in some time unit (*fs*, ..., *ms*, ..., *sec*, *mn*, ...). Such a real-time clock can be used as a usual signal in the process P encapsulated in S. Systems S_1, ..., S_n may be composed (with the standard composition of Signal) in a same system S, but the *ms* of a given system S_i is a priori not synchronous with the *ms* of another system S_j. Then it is possible to specify constraints in the system S on these different signals, to express for instance some variation limits of different clocks.

For that purpose, we have defined a new taxonomy of polychronous processes to characterize precisely the following classes: system, task, (logical) process, function, reaction, diagram, observer, controller... This characterization is based on properties such as time reference, input-output clock relations, input-output dependences, determinism, exo/endochrony. For example, a system is either a physical system abstraction, or a basic system, or a system of systems. A basic system has a unique continuous time reference; it provides an internal actual discrete time unit subset of its external continuous time, shared by all its components. As another example, for a subclass of logical processes: a function is a deterministic, inout clocked, endochronous and atomic process that denotes a mathematical flow function. All these different semantic classes are provided syntactic counterparts in the new Signal V5.

6.2. Experimental Polarsys platform

Participants: Loïc Besnard, Thierry Gautier, Jean-Pierre Talpin.

In the context of the OPEES project (http://www.opees.org/), we have experimented the IWG Eclipse platform Polarsys (http://www.eclipse.org/org/press-release/20111102_polarsys.php). Polarsys is a new industry collaboration to build open source tools for safety-critical software development. The integration of Polychrony into this platform has been realized in collaboration with the CS company. CS and Inria have produced the Polychrony experimentation report which is included in the global experimentation report. This document gathers the experiments performed by the several partners involved in the OPEES project on the Polarsys platform. An experiment is defined as the way one partner takes his component and uses it to check any of the services within the Polarsys environment. The services are functions the partners want the Polarsys environment to offer.

For the qualification of the Polychrony component on the Polarsys platform, CS and Inria provide the following documents:

- The Tool Quality Assurance Plan Template (TQAP). This document defines the OPEES quality assurance arrangements and gives some guidance to satisfy them. It focuses on qualification aspects and gives in appendices guidance for some criteria tool qualification with an example for Polychrony Tool.
- The Tool Verification Cases and Procedures (TVCP) document. It presents the test cases to be performed for the qualification of Polarsys Polychrony Verifier component as described in the TQAP.
- The Tool Verification Results (TVR). It presents the results of tests performed for the qualification of Polychrony Verifier on several operating systems, as described in the TQAP.

6.3. Translation validation of Polychronous Equations with an iLTS Model-checker

Participants: Van-Chan Ngo, Jean-Pierre Talpin, Thierry Gautier, Paul Le Guernic, Loïc Besnard.

This work [16], [18], which is part of the VeriSync project, focuses on verifying the correctness of transformations on abstract clocks in the Signal compiler [8]. We propose to use model checking technique over Polynomial Dynamical Systems (PDS) with the Sigali model checker [39].

Adopting the *translation validation* approach of [55], [54], we present an automated verification process to prove the correctness of a multi-clocked synchronous language compiler. Due to the very important role of abstract clocks and clock relations, we are interested in proving that abstract clocks and clock relations semantics of source programs are preserved during the compilation phases. Each individual transformation or optimization step of the compiler is followed by our verification process which proves the correctness of this step. The compiler will continue its work if and only if the correctness is proved positively. This approach avoids the disadvantage of proving in advance that the compiler always behaves correctly since every small change to the compiler requires reproving it.

Our verification framework uses polynomial dynamical systems (PDS) over a finite field, as common semantics for both source and transformed programs. We formalize the abstract clocks semantics of *polychronous equations* with the finite field modulo $p = 3$ as a PDS [16]. For a signal x, if x is boolean, we use the values $-1, 0, +1$ to encode (respectively) the fact that it is present and false, absent, or present and true. Then, the abstract clock can be represented by x^2. If x is non-boolean, we only encode the abstract clock by x^2, meaning that $x^2 = 0$ encodes x is absent, $x^2 = 1$ encodes x is present. An appropriate relation called *refinement* for PDSs is proposed to represent the correct transformation relation between the source and transformed programs. Then a dedicated checking procedure is proposed within Sigali to check the correct transformation relation. The checking procedure is based on the simulation techniques [30]. It is implemented as extension function of the Sigali model checker within the Polychrony toolset.

We have proposed an approach to prove the clock semantic preservation of the Signal compiler transformations until the generation code phase as well. The verification method applied to code generation phase addresses the formal verification of the generated C-code from a refined and optimized intermediate specification in which the compiler enforces logical timing constraints and in which the execution order of data-flow equations is completely scheduled. As a result, all individual transformations, optimizations, and code generation phases of the compiler are followed by a verification step which proves the correctness of transformations. The compiler continues if and only if correctness is proved and returns an error and a trace otherwise. The main idea is that the sequential C code is translated into the target synchronous program thanks to the intermediate SSA form, which is based on the work in [3]. In addition, if a refinement relation between two PDSs does not exist, our validator will find the set of states along with their associated events, which can be used to construct counterexamples in the transformed program [18].

6.4. Formal Verification of Transformations on Abstract Clock in Synchronous Compilers

Participants: Van-Chan Ngo, Jean-Pierre Talpin, Paul Le Guernic.

Translation validation was introduced in the 90's by Pnueli et al. as a technique to formally verify the correctness of code generated from the data-flow synchronous language Signal using model checking. Rather than certifying the code generator (by writing it entirely using a theorem prover) or qualifying it (by obeying to the 27 documentations required as per the DO-178C), translation validation provides a scalable approach to assessing the functional correctness of automatically generated code. By revisiting translation validation, which in the 90's suffered from the limitations of theorem proving and model checking techniques available then, we aim at developing a scalable and flexible approach that applies to the existing 500k-lines implementation of Signal, Polychrony, and is capable of handling large-scale, possibly automatically generated, Signal programs, while using of-the-shelf, efficient, model-checkers and SAT/SMT-solving libraries [36], [63].

The abstract clock semantics of polychronous equations is formalized as a first-order logic formula over boolean variables. For a signal x, if x is boolean then we use two boolean variables x, and $\widehat{x}$ to represent the value of signal x and it abstract clock, respectively. If x is non-boolean signal, we only need to capture its abstract clock by a boolean variable $\widehat{x}$. The boolean variable $\widehat{x}$ is true when the signal x is present and otherwise it is absent. The equational structure of a synchronous data-flow language makes that it is natural to represent the relations between abstract clocks described implicitly or explicitly by equations in terms of first-order logic formulas. And then the combination of equations can be represented by the conjunction of the corresponding first-order logic formulas. We assume that all considered programs are supposed to be written with the primitive operators, meaning that derived operators are replaced by their corresponding primitive ones, and there is no nested operators such as $z := x$ default $(y$ when $b)$. The nested operators are broken by using fresh variables. These formulas use the usual logic operators and numerical comparison functions. Consider a general equation $y := R(x_1, x_2, ..., x_n)$, where R is an operator defined in a synchronous language (e.g. `suspend` in Esterel, `when` in Signal...), or it can be a usual boolean or numerical operator, then the abstract clock semantics of this equation can be represented as a first-order logic formula over the clocks and/or the boolean value of the involved signals $\Phi(\widehat{y}, \widehat{x_1}, \widehat{x_2}, ..., \widehat{x_n}, x_1, ...)$. For a boolean expression defined by numerical comparison functions and numerical expressions, to ensure the result formulas are boolean, we only encode the fact that the clocks of boolean and numerical expressions are synchronized, and we avoid encoding the numerical comparison function which defines the value of the boolean expression and the numerical expressions. For each i^{th} equation in program P, we denote by Φ_{eq_i} its abstract clock semantics, then the abstract clock semantics of P can be represented by a first-order logic formula, called its *clock model*, denoted as:

$$\Phi_P = \bigwedge_i^n \Phi_{eq_i} \tag{1}$$

where n denotes the number of equations composed in P. The detailed encoding scheme of the Signal language can be found in [19].

Given two clock models P_1 and P_2, we say that P_2 is a *correct transformation* on abstract clocks of P_1, or $P_2 refines P_1$ w.r.t the clock semantics, if it satisfies:

$$\forall I.(I \models \Phi_{P_2} \rightarrow I \models \Phi_{P_1}) \tag{2}$$

We write $P_2 \sqsubseteq_{clock} P_1$ to denote the fact that P_2 refines P_1. We also provide an approach to check the existence of refinement by using a SMT-solver that is based on the following theorem:

Theorem. Given a source program P_1 and its transformed program P_2, P_2 is a correct transformation of P_1 on abstract clocks if it satisfies that the formula Φ_{P_1} is a logical consequence of the formula Φ_{P_2}, or

$$\models(\Phi_{P_2} \rightarrow \Phi_{P_1}) \text{ if and only if } P_2 \sqsubseteq_{clock} P_1 \tag{3}$$

Here, we delegate the checking of the above formula against the clock models to a SMT-solver that efficiently deals with first-order logic formulas over boolean and numeric expressions. The checking formulas belong to decidable theories, this solver gives two types of answers: *sat* when the formula has a model (there exists an interpretation that satisfies it); or *unsat* otherwise. Our implementation uses the SMTLIB common format [31] to encode the formulas obtained from the previous step as input of SMT solvers. For our implementation, we consider the Yices solver [38], which is one of the best two solvers at the last SMTCOMP competition [59].

6.5. Formal Verification of Transformations on Data Dependency in Synchronous Compilers

Participants: Van-Chan Ngo, Jean-Pierre Talpin, Paul Le Guernic.

We propose an approach to prove the data dependency semantic preservation of transformations in a synchronous compiler (such as that of Signal). In the Signal language, the scheduling or data dependency is expressed implicitly through polychronous equations. We use *Synchronous Data-flow Dependence Graphs* (SDD Graphs) [46], [50] to formalize the data dependency semantics of polychronous equations. A tuple $< G, C, fE >$ is a SDD graph if and only if:

- $G =< N, E, I, O >$ is a dependence graph $< N, E >$ with I/O nodes: the inputs I and the output O such that I, O are subsets of N and I and O are disjoint.
- C is a set of constraints, called clocks.
- $fE : E \longrightarrow C$ is a mapping labeling each edge with a clock; it specifies the existence condition of the edges.

For instance, for the *counter* example:
$zv := v\$1|v := (1 \text{ when } rst) \text{ default } zv + 1$
we get a SDD graph with:

- $N = \{1, v, zv + 1\}$
- $E = \{(1, v), (zv + 1, v)\}$
- $C = \{\widehat{rst}, \widehat{v} \wedge \neg\widehat{rst}\}$
- $fE((1, v)) = \widehat{rst}, fE((zv + 1, v)) = \widehat{v} \wedge \neg\widehat{rst}$

Let $SDD_1 =< G_1, C_1, fE_1 >$ and $SDD_2 =< G_2, C_2, fE_2 >$ to be two SDD graphs which represent the data dependency semantics of source and transformed programs, we say that SDD_2 is a *correct transformation* of SDD_1 on data dependency, or $SDD_2 refines SDD_1$ w.r.t the data dependency semantics, if it satisfies that for any pair of nodes $x, y \in G_1 \cap G_2$ with $(x, y) \in E_1$:

- $fE_1(x, y)) \Rightarrow((x, y) \in E_2 \wedge fE_2(x, y))$ (reinforcement)
- $(fE_2(x, y) \wedge fE_2(y, x) \Leftrightarrow \text{ false})$ (deadlock consistency)

6.6. Experiment with constraint-based testing

Participants: Christophe Junke, Loïc Besnard, Jean-Pierre Talpin.

Based on past experiences with contraint-based testing of Lustre programs, we investigated automatic test *sequences* generation for Signal: from a given test objective expressed as a boolean flow (or an event), we try to generate a sequence of inputs over discrete time which lead to an observation of the test objective. Our approach was based on an existing tool named GATeL, from CEA LIST, with the kind permission of its authors. This tool targets the Lustre language, so we reused Polychrony's Lustre generator to export Signal programs as Lustre nodes and use the result with GATeL to generate test sequences. The resulting test sequences were in turn formatted in a way suitable for simulation according to the original compilation of Signal to C: in other words, the generated sequences were tested on the actual program resulting from compilation of considered Signal specifications. During this experiment, we corrected Signal's Lustre generator tool which suffered from some several bugs that made it emit consistently incorrect Lustre programs. After some work, we could translate faithfully a little more than sixty existing Signal programs of simple to moderate complexity.

Our contribution is an example of how Signal can benefit from the pool of existing tools applicable to Lustre and why having a correct Signal-to-Lustre translator can be useful for Signal programs. This approach has its limits because it is not always possible nor adequate to fully translate a Signal program to Lustre: (1) By requiring the existence of one root clock and changing a program's input/output interface, it may be possible to simulate a Signal program in Lustre, but with loss of information (like user-defined flow dependencies); hence, some results based on the one Lustre implementation of a model may not easily be generalized to every possible execution of the original Signal program; (2) the complexity of Signal's semantics is mainly motivated by the power it gives to handle partial system specifications during the development process, whereas most Lustre tools expect fully defined executable programs; as such, they are of little help when dealing with most Signal programs. For those main reasons, it might be better to study and implement verification techniques around the Signal language and extend the set of formal tools that can reason about Signal programs.

More generally, our experiment can lead us to consider the use of constraint solving techniques with Signal, not only for verification but also compilation and simulation.

6.7. Polychronous modeling, analysis and validation for timed software architectures in AADL

Participants: Yue Ma, Huafeng Yu, Paul Le Guernic, Loïc Besnard, Thierry Gautier, Jean-Pierre Talpin.

High-level architecture modeling languages, such as AADL (Architecture Analysis and Design Language), are gradually adopted in the design of embedded systems so that design choice verification, architecture exploration and system property checking are carried out as early as possible. We are interested in the clock-based timing analysis, modeling and validation of software architectures specified in AADL [15]. In our approach, we first analyze the timing semantics of AADL, from which the formal polychronous/multiclock semantics is derived thanks to the multiclock nature of AADL specifications. Thus users are not suffered to find and/or build the fastest clock in the system. This distinguishes from [45], [37], where synchronous semantics is a prerequisite. This polychronous semantics is then expressed via a polychronous model of computation (MoC) [8] covering both AADL software, execution platform, and their binding. In addition, AADL thread-level scheduling is also explored and integrated according to affine clock relations [58]. In the framework of Polychrony, C or Java code is generated from the polychronous MoC. Simulation can then be carried out for the purpose of performance evaluation and verification.

Polychrony provides the back-end semantic-preserving transformation, scheduling, code generation, formal analysis and verification, architecture exploitation, and distribution [2]. With the scheduler synthesis, the translated AADL model is complete and executable, and can be used for the following analysis and validation [15]: 1) static analysis, including determinism identification and deadlock detection; 2) profiling-based analysis of real-time characteristics of a system [47]; 3) affine clock calculus to analyze the affine relations between clocks [58]; 4) co-simulation of AADL specifications and demonstration using the VCD technique [60]; 5) real-time scheduling and software/hardware allocation through the SynDEx tool [43].

An automatic toolchain dedicated to AADL modeling, scheduling, time analysis, verification, and simulation has been implemented and also integrated as plug-ins in the Eclipse framework. This toolchain (referred to as ASME2SSME) has been migrated from AADL V1.5.8 to AADL V2.0, together with OSATE V2. An experiment of interpretation of AADL Behavior Annex (BA) is initially performed, so that the Behavior Annex plugin is integrated in the modeling and transformation.

The whole model transformation and simulation chain has been migrated to Eclipse Indigo and attached to Polarsys as an Eclipse RCP. A tutorial case study, developed in the framework of the OPEES project [21], is adopted to illustrate the effectiveness of our contribution.

6.8. Static affine clocked-based scheduling and its seamless integration to ASME2SSME

Participants: Huafeng Yu, Yue Ma, Loïc Besnard, Thierry Gautier, Paul Le Guernic, Jean-Pierre Talpin.

An AADL model is not complete and executable if the thread-level scheduling is not resolved. Some scheduling tools, such as Cheddar [57], are well connected to AADL for schedulability analysis, scheduler synthesis and simulation inside these tools. However, they do not completely satisfy our demands for the following reasons: 1) logical and chronometric clocks are easily transformed into each other for formal and real-time analysis; 2) more events, such as input/output frozen events are also involved in the analysis; 3) static and periodic scheduling rather than stochastic/dynamic scheduling is expected due to predictability and formal verification; 4) the scheduling is easily and seamlessly connected to affine clock systems [58] so that formal analysis can be performed in Polychrony. Affine clock relations yield an expressive calculus for the specification and the analysis of time-triggered systems. A particular case of affine relations is the case of affine sampling relation expressed as $y = \{d \cdot t + \phi \mid t \in x\}$ of a reference discrete time x (d, t, ϕ are integers): y is a subsampling of positive phase ϕ and strictly positive period d on x.

We therefore propose a static affine-clocked-based scheduler synthesis process in the transformation from AADL to Signal, which includes the following subprocesses: 1) *calculate hyper-period* from the periods of all the threads according to the least common multiple principle; 2) *perform the scheduling* based on the hyper-period, and valid schedules are calculated according to a static, non-preemptive, and single-processor scheduling policy. More precisely, discrete events of each thread, such as dispatch, input/output frozen time, start and complete, are allocated in the hyper-period on condition that all their timing properties are satisfied. Affine clock relations of these events are ensured during the calculation. In the calculation process, different scheduling policies are considered, such as EDF and RM; 3) *export schedules to Signal affine clocks in a direct way*. This process, implemented as an independent Eclipse plugin, has been seamlessly integrated into the ASME2SSME toolchain.

6.9. Code distribution and architecture exploration via Polychrony and SynDEx

Participants: Huafeng Yu, Yue Ma, Loïc Besnard, Thierry Gautier, Jean-Pierre Talpin, Paul Le Guernic.

We propose an approach to address code distribution and multi-processor architecture exploration via the Polychrony and SynDEx toolchains. We consider high-level AADL models for the specification of multi-processor architecture. This architecture generally has a multiclock nature, thus it is modeled with a polychronous MoC. In this way, users are not suffered to find and/or build the fastest clock for a multi-processor architecture. According to this principle, AADL models are transformed into Signal models. To bridge between the polychronous semantics of Signal and synchronous semantics of SynDEx, clock synthesis in Polychrony [24] is applied. The translation from Signal to SynDEx is integrated in Polychrony. Finally, SynDEx models are used to perform distribution, scheduling, and eventually executive code generation for a specific architecture.

The main advantages of our approach are: 1) a formal model is adopted to connect the three languages, and it helps to preserve the semantic coherence and correct code generation in the transformations; 2) the formal model and methods used in the transformation are transparent to AADL practitioners, and it is fast and efficient to have the illustrative results for architecture exploration; 3) it provides the possibility for one of the three languages to take advantage of the functionalities provided by the other two languages. A toolchain has been developed, which includes model transformations between the three languages, considering both semantic and syntactic aspects. A tutorial case study, developed in the framework of the CESAR project [20], was adopted to demonstrate our contribution.

6.10. Design of safety-critical Java applications using affine abstract clocks

Participants: Adnan Bouakaz, Jean-Pierre Talpin.

Safety-critical Java (SCJ) is a domain specific API of Java that aims at the development of qualified and certified embedded systems. Despite its simplified memory and concurrency models, it is always difficult to ensure functional determinism and schedule feasibility when using shared-memory and traditional lock-based mutual exclusion protocols. Automated code generation techniques from data-flow specifications allow waiving part of the difficult and error-prone tasks of programming real-time schedules for computations and communications from the engineering process. Our ADFG tool aims at automatic SCJ code generation from data-flow specifications for a multitask implementation with an earliest-deadline first scheduler. The tool integrates the necessary formal analyses, model transformations, and the SCJ annotation checker as well.

The underlying data-flow model, called the affine data-flow (ADF) model of computation [14], is similar to cyclo-static data-flow graphs; it has however ultimately periodic production and consumption rates and a time-triggered operational semantics. We have also proposed a scheduling analysis of ADF specifications that consists of two major steps:

- The construction of abstract affine schedules for computations that minimize buffering requirements under the assumption of read-write precedences and exclude overflow and underflow exceptions over communication channels. Affine transformations of clocks were first introduced in the context of Signal programs [58] and used in the ADF model to relate the activation rates of connected actors.
- The concretization of the affine schedules using an earliest-deadline first (EDF) symbolic schedulability analysis in a way that read-write precedences is ensured without the need for lock-based mechanisms and the processor utilization factor is maximized.

6.11. Polychronous controller synthesis from MARTE CCSL timing specifications

Participants: Huafeng Yu, Loïc Besnard, Thierry Gautier, Jean-Pierre Talpin, Paul Le Guernic.

CCSL (Clock Constraint Specification Language) [29] is defined in an annex of the UML MARTE profile [53]. We are interested in the analysis, synthesis, code generation and simulation of polychronous systems specified in CCSL. Timed systems subject to clock expressions or relations can be modeled, specified, analyzed, and simulated within software environments such as SCADE [40], TimeSquare [44] and Polychrony. The motivation of our work, to address the simulation and code generation of polychronous systems, is to take advantage of the formal framework of Polychrony in the context of a high-level specification formalism, MARTE CCSL [62]. Yet, our work considers a novel approach with regard to previous approaches: to generate executable specifications by considering discrete controller synthesis (DCS) [56], [51], [52].

Based on our previous work on clock hierarchization [61] and the general clock synthesis approach [62], our current work concentrates on the study of interface-oriented clock synthesis in the context of distributed components. In this work, CCSL clock constraints are specified on the clocked signals that pass through the interface, and the controller to synthesize is used to ensure the constraints. Interface-level synthesis helps to reduce the synthesis complexity since communication concerns and internal component behavior are isolated from the synthesis. The controllability analysis of signals and clock relations are studied with regard to

endochronous, polychronous, and reactive components. This analysis leads to the separation of controllable and uncontrollable signals in the synthesis. Observers of CCSL clock constraints have been proposed in order to specify control objectives. In addition, properties of local components and the global system, such as determinism and deadlock, are also initially studied.

6.12. An integration language for Averest/Quartz and Polychrony/Signal

Participants: Ke Sun, Jean-Pierre Talpin.

As typical synchronous languages, Quartz and Signal possess their own respective characteristics [11]. In particular, Quartz, an imperative synchronous language, mainly focuses on the description of the control-flow. The Quartz model is always reactive and synchronously deterministic. Different from Quartz models which can only be monochronous, a process in Signal may be polychronous, meaning that its clock hierarchy can form a forest. Therefore, the potential integration between Averest, a framework for Quartz, and Polychrony, a toolset for Signal, may offer a practical mode to develop globally asynchronous locally synchronous (GALS) systems: program imperative and reactive modules in Quartz, then synthesize the scheduler from their Signal network specification.

To maximally benefit from the existing achievements for the two languages [12], the main idea is to communicate the Quartz modules with each other via asynchronous wrappers without changing the original code. Considering that the Quartz modules should be still deterministic in asynchronous environment, the wrapper should be capable of controlling the IO streams. On the other hand, the wrapper, as a module interface, will make sense for automatic scheduler synthesis, the next step.

We will propose a new, easy to use, domain-specific language to help the user specify the input traces as requirements to the environment and define the IO traces as guarantee of the module. From the user-defined specification, a series of clock constraints, assertions, etc. may be synthesized in the form of Signal specification. Thus, this language may bridge the gap between Polychrony/Signal and Averest/Quartz.

MUTANT Project-Team

6. New Results

6.1. Information-Geometric Approach to Real-time Audio Change Detection

Participants: Arnaud Dessein, Arshia Cont.

We developed a generic framework for real-time change detection of audio signals using methods of information geometry. The present method is limited to generative models of audio signals based on generic exponential distribution families. The proposed system detects changes by controlling the information rate of the signal as they arrive in time. The method also addresses shortcomings of traditional approaches based on cumulative sums which assume known parameters before change. This is achieved by calculating exact generalized likelihood ratio test statistics with complete estimation of unknown parameters in respective hypothesis [9]. The interpretation of this framework within a dually flat geometry of exponential families provide tractable algorithms for online use. Results are presented for speech segmentation into different speakers and polyphonic music segmentation.

6.2. Real-time Polyphonic Music Recognition

We investigated real-time recognition of overlapping music events in two context of dictionary-based detection and real-time alignment:

6.2.1. Real-time detection of overlapping sound events using non-negative matrix factorization

Participants: Arnaud Dessein, Arshia Cont.

Non-negative matrix factorization (NMF) methods have naturally found their way since their inception to sound and music processing. This work is an extension to our previous work in [1] on Real-time Music Transcription using sparse NMF methods. We investigate the problem of real-time detection of overlapping sound events by employing NMF techniques. We consider a setup where audio streams arrive in real-time to the system and are decomposed onto a dictionary of event templates learned off-line prior to the decomposition. An important drawback of existing approaches in this context is the lack of controls on the decomposition. We propose and compare two provably convergent algorithms that address this issue, by controlling respectively the sparsity of the decomposition and the trade-off of the decomposition between the different frequency components. Sparsity regularization is considered in the framework of convex quadratic programming, while frequency compromise is introduced by employing the beta-divergence as a cost function. The two algorithms are evaluated on the multi-source detection tasks of polyphonic music transcription, drum transcription and environmental sound recognition. The obtained results in [20] show how the proposed approaches can improve detection in such applications, while maintaining low computational costs that are suitable for real-time.

A specialized version of NMF for Real-time Music Transcription is exposed in Arnaud Dessein's PhD thesis [9].

These methods will be subject to software development in 2013.

6.2.2. Robust Real-time Polyphonic Audio-to-Score Alignment

Participant: Arshia Cont.

The *Antescofo* system is polyphonic since 2009 but its use in highly polyphonic and noisy concert environments have been challenging. To overcome this, we have studied more robust inference mechanisms. As a results, the previous inference mechanism based on maximum a posteriori of Viterbi Forward variables in mixed semi-Markov and Markov chains in [2] were abandoned in favor of a more robust method based on *importance resampling* on state-space models and smoothing of variable-order hybrid chains. This has led to robust real-time alignment and the employment of the system in various Piano performances in 2012. Further extensions are currently under study.

6.3. Real-time Multi-object Detection for Music Signals

Participants: Philippe Cuvillier [Master 2 ATIAM], Arshia Cont.

Multiple-object detection and tracking has been widely used in applications such as missile tracking and radar and has given birth to several formalisms such as Random Finite Sets [33]. Such formalisms can be seen as extensions to existing probabilistic inference mechanisms with explicit birth and death stochastic mechanisms for multiple source tracking.

In this work we aim at studying such formalisms in the case of real-time music signal processing. The idea is to track multiple sources (instruments, audio flows) from one source of observation. This approach can be beneficial to two main applications in real-time music listening:

- Extension of existing audio-to-score [2] or audio-to-audio alignment [7] mechanisms (currently based on one source) to multiple objects can address the following short-comings of existing approaches: explicit consideration for asynchrony of parallel sources; robustness to uncertainties on one or more voices.
- Studying the classical *Partial Tracking* applications in audio processing within the RFS context can lead to better results in low-level sinusoidal partial tracking of sounds.

Early studies of such formalisms are exposed in [25]. Concrete applications will be exposed in 2013.

6.4. Antescofo Language Extensions and Performance Fault-Tolerance

We have improved the *Antescofo* framework widely used for mixed instrumental and live electronic computer music. The new framework paves the way for future language extensions and paves the way for future research regarding performance fault-tolerance, synchronization mechanisms and formal verifications.

6.4.1. Antescofo Language Extensions

Participants: José Echeveste, Jean-Louis Giavitto, Florent Jacquemard, Arshia Cont.

To further extend the *Antescofo* language, the system has been formally modeled as a network of parametric timed automata in [29]. The model obtained provides operational semantics for the input scores, in particular the interaction between the instrumental and electronic parts and the timing and error handling strategies mentioned below. This approach would enable better authoring of time and interaction during programing/composing, permits to use state of the art software verification tools for the static analysis of Antescofo scores and also provides means to address critical aspects of musical performances in real-time.

In parallel, a new grammar for the score language and a new architecture have been designed for *Antescofo*, taking into account new demands from the community such as addition of timed variables in the language, dynamic time processes, time-conditional constructs, and more.

6.4.2. Performance Fault-Tolerance and Synchronization Mechanisms

Participants: José Echeveste, Jean-Louis Giavitto, Arshia Cont.

We formalized the timing strategies for musical events taking into account the variability of environment signals (musicians) and their effect on computer events programmed in *Antescofo*. The result of this work is presented in [15], where new block attributes in the language determine expected behavior in case of environment changes in real-time (errors, timing discrepancies, etc.). These additions have been implemented in the current version of the system and are widely used by the user community.

6.5. Temporal Analysis and Verification of Interactive Music Scores

Participants: Léa Fanchon [Master 2 École Centrale], Florent Jacquemard.

Léa Fanchon's Masters thesis, under the supervision of Florent Jacquemard, [26] presents an analysis module that complements the real-time score authoring and performance in *Antescofo*, with the aim of exploring possible behavior of authored programs with respect to possible deviations in human musician performance. This work employs formal methods for temporal automata networks using linear constraint inference techniques commonly in use for task scheduling and circuit verifications.

Obtained results pave the way for future works in formal verification of interactive multimedia applications, being one of the first of its kind in computer music literature, and provides the following input to programmers and artists using *Antescofo*:

- Evaluation of robustness of the program with respect to the environment's (musician's performance) temporal variations,
- Feedback to programmers/artists on critical synchronization points for better programming.

An article describing this work is currently in preparation for a submission to a computer music conference.

6.6. Formal study of Antescofo as a Reactive System

Participants: Guillaume Baudart [Master 2 ATIAM], Florent Jacquemard, Marc Pouzet [ENS], Jean-Louis Giavitto, Arshia Cont.

An *Antescofo* score/program can be considered as a specification of a reactive system through its coupling of a machine listening with a real-time synchronous language. In his master thesis under the supervision of Florent Jacquemard and Marc Pouzet (team Parkas), Guillaume Baudart has has studied the links between the reactive system of *Antescofo* and existing synchronous languages such as *Lucid Synchrone* [36] and *Reactive ML* [34]. The reactive engine of a preliminary version of *Antescofo* was developed in both languages and their structures were compared.

This study reveals the particularities of musical applications of reactive systems specific to *Antescofo* (see [24]). *Reactive ML* allows dynamic constructions but real-time performance can not be guaranteed especially when the machine listening is combined with the reactive system. On the contrary, *Lucid Synchrone* does not easily allow dynamic process creation. Each language specificity leads to strong considerations in the program/score structure for the artists. This work will be continued in 2013 to further strengthen ties between the reactive aspects of *Antescofo* and that of synchronous languages.

6.7. Tree Structured Presentation of Symbolic Temporal Data

Participants: Florent Jacquemard, Michael Rusinowitch [Project-team Cassis], Luc Segoufin [Project-team Dahu].

In traditional music notation, in particular in the languages used for the notation of mixed music such as Antescofo DSL, the durations are not expressed by numerical quantities but by symbols representing successive subdivisions of a reference time value (the beat). For this reason, trees data structures are commonly used for the symbolic representation of rhythms in computer aided composition softwares such as OpenMusic (developed at Ircam). It is therefore worth studying the applications in rhythm notation of existing formalisms for recognizing, querying, transforming and learning sets of tree structured data.

In 2012 we have studied several classes of tree recognizers which could be of interest in this context. First, with Michael Rusinowitch we have proposed in [16] a novel class of automata computing on unranked trees, which are context free in two dimensions: in the the sequence of successors of a node and also along paths. Second, we studied with Luc Segoufin [21] automata and logics computing on data trees and their relationship. Data trees are unranked ordered trees where each node carries a label from a finite alphabet and a datum from some infinite domain.

PARKAS Project-Team

6. New Results

6.1. Reactive Programming

Participants: Mehdi Dogguy, Louis Mandel, Cédric Pasteur, Marc Pouzet.

ReactiveML is an extension of OCaml with synchronous concurrency, based on synchronous parallel composition and broadcast of signals. The goal is to provide a general model of deterministic concurrency inside a general purpose functional language to program reactive systems. It is particularly suited to program discrete simulations, for instance of sensor networks.

One of the current focus of the research is being able to simulate huge systems, composed of millions of agents, by extending the current purely sequential implementation in order to be able to take advantage of multi-core and distributed architectures. This goal has led to the introduction of a new programming construct, *reactive domain*, which allows to define local time scales. These domains help for the distribution of the code but also increase the expressiveness of the language. In particular, it allows to do time refinement. A paper on this new construct and the related static analysis has been submitted. We have implemented a new runtime for ReactiveML, that uses the MPI (Message Passing Interface) library to run programs on multi-core and distributed architectures.

We have also investigated new static analyses for the language. Following the work of PhD thesis of Mehdi Dogguy, we have studied a new analysis which adds usages on signals to be able to ensure one to one communications. We have also studied a new reactivity analysis which ensures that a process can not prevent the other ones to from executing. This analysis will be published in [10].

6.2. n-Synchronous Languages

Participants: Louis Mandel [contact], Marc Pouzet, Albert Cohen, Adrien Guatto.

The n-synchronous model introduced a way to compose streams which have *almost the same clock* and can be synchronized through the use of a finite buffer.

We have designed the language Lucy-n to program in this model of computation [40]. This language is similar to the first order synchronous data-flow language Lustre in which a buffer operator is added. A dedicated type system allows to check that programs can be executed in bounded memory and to compute sufficient buffer sizes. Technically it is done through the introduction of a subtyping constraint at each bufferization point.

- In collaboration with F. Plateau (Prove&Run), we developed a new resolution constraint algorithm for the clocking of Lucy-n programs [8]. Even if the new algorithm is less efficient that the one using abstraction, it has the advantage to be more precise and thus to accept more programs. It is useful for example for the static scheduling of Latency Insensitive Designs [41].
- We worked on an extension of the synchronous model with integer clocks. This extension allows to produce and consume several values at each activation. It has large implication on the semantics, clock typing, causality and code generation of the language.
- We have continue the work on the code generation. In particular, we have been designing a new intermediate representation that allows to deal with integer clocks.

6.3. Strong normal form for large integers, boolean functions and finite automata

Participant: Jean Vuillemin.

Jean Vuillemin's recent work focusses on finding Strong Normal Form for large Integers, Boolean functions and finite Automata, with applications to circuits and software.

- [16] is the latest version of JV's course notes at ENS "De l'algorithme au circuit".
- [9] shows that the ordered dimension of a Boolean function is a lower bound on the size of most known ordered Decision Diagrams, and that ordered decision diagrams can be efficiently constructed an operated upon.
- [6] shows an approach to circuit protection against side-channel attacks based on a statistical analysis of power traces derived from actual measures of the circuit in operation.

6.4. A theory of safe optimisations in the C11/C++11 memory model and applications to compiler testing

Participants: Francesco Zappa Nardelli [contact], Robin Morisset, Pankaj Pawan.

Compilers sometimes generate correct sequential code but break the concurrency memory model of the programming language: these subtle compiler bugs are observable only when the miscompiled functions interact with concurrent contexts, making them particularly hard to detect. In this work we design a strategy to reduce the hard problem of hunting concurrency compiler bugs to differential testing of sequential code and build a tool that puts this strategy to work. Our first contribution is a theory of sound optimisations in the C11/C++11 memory model, covering most of the optimisations we have observed in real compilers and validating the claim that common compiler optisations are sound in the C11/C++11 memory model. Our second contribution is to show how, building on this theory, concurrency compiler bugs can be identified by comparing the memory trace of compiled code against a reference memory trace for the source code. Our tool identified several mistaken write introductions and other unexpected behaviours in the latest release of the gcc compiler.

A paper on this work has been submitted to an international conference [15].

6.5. A verified compiler for relaxed-memory concurrency

Participant: Francesco Zappa Nardelli [contact].

We studied the semantic design and verified compilation of a C-like programming language for concurrent shared-memory computation above x86 multiprocessors. The design of such a language is made surprisingly subtle by several factors: the relaxed-memory behaviour of the hardware, the effects of compiler optimisation on concurrent code, the need to support high-performance concurrent algorithms, and the desire for a reasonably simple programming model. In turn, this complexity makes verified (or verifying) compilation both essential and challenging. This project started in 2010, and in 2012 we submitted a journal version, describing the correctness proof of all the phases of our CompCertTSO compiler (including experimental fence eliminations). This has been accepted for publication in Journal of the ACM [3].

In collaboration with Jaroslav Sevcik (U. Cambridge), Viktor Vafeiadis (MPI-SWS), Suresh Jagannathan (Purdue U.), Peter Sewell (U. Cambridge).

6.6. Compiling C/C++ concurrency from C++11 to POWER

Participant: Francesco Zappa Nardelli [contact].

The upcoming C and C++ revised standards add concurrency to the languages, for the first time, in the form of a subtle relaxed memory model (the C++11 model). This aims to permit compiler optimisation and to accommodate the differing relaxed-memory behaviours of mainstream multiprocessors, combining simple semantics for most code with high-performance low-level atomics for concurrency libraries.

We studied the the correctness of two proposed compilation schemes for the C++11 load and store concurrency primitives to Power assembly, having noted that an earlier proposal was flawed. (The main ideas apply also to ARM, which has a similar relaxed memory architecture.)

This should inform the ongoing development of production compilers for C++11 and C1x, clarifies what properties of the machine architecture are required, and builds confidence in the C++11 and Power semantics.

A paper describing this work will appear in POPL 2012 [5].

In collaboration with Kayvan Memarian (previously student in the Moscova EPI, currently at U. Cambridge).

6.7. Compilation techniques for synchronous languages

Participants: Marc Pouzet [contact], Adrien Guatto, Léonard Gérard, Cédric Pasteur.

- The generation of efficient sequential code for synchronous data-flow languages raises two intertwined issues: control and memory optimization. While the former has been extensively studied, for instance in the compilation of Lustre and SIGNAL, the latter has been only addressed in a restricted manner. Yet, memory optimization becomes a pressing issue when arrays are added to such languages, for example, SCADE 6 [8]. We have proposed a two-levels solution to the memory optimization problem. It combines a compile-time optimization algorithm, reminiscent of register allocation, paired with language annotations on the source given by the designer. Annotations express in-place modifications and control where allocation is performed. Moreover, they allow external functions performing in-place modifications to be imported safely. Soundness of annotations is guaranteed by a semilinear type system and additional scheduling constraints. A key feature is that annotations for well-typed programs do not change the semantics of the language: removing them may lead to a less efficient code but with the very same semantics.

  The method has been implemented in HEPTAGON, the compiler developed in the team of a Lustre-like synchronous language extended with hierarchical automata and arrays. Experiments show that the proposed approach removes most of the unnecessary array copies, resulting in faster code that uses less memory. This work has been presented at the *ACM Intern. Conf. on Languages, Compilers and Tools for Embedded Systems (LCTES'12)* in June 2012 and it has received the *Best paper award*.

6.8. Generation of Parallel Code from Synchronous Programs

Participants: Albert Cohen [contact], Léonard Gérard, Adrien Guatto, Nhat Minh Le, Marc Pouzet.

- Efficiently distributing synchronous programs is a challenging and long-standing subject. This paper introduces the use of futures in a Lustre-like language, giving the programmer control over the expression of parallelism. In the synchronous model where computations are considered instantaneous, futures increase expressiveness by decoupling the beginning from the end of a computation. Through a number of examples, we show how to desynchronize long computations and implement parallel patterns such as fork-join, pipelining and data parallelism. The proposed extension preserves the main static properties of the base language, including static resource bounds and the absence of deadlock, livelock and races. Moreover, we prove that adding or removing futures preserves the underlying synchronous semantics.

  This work has been presented at the *ACM Intern. Conf. on Embedded Software (EMSOFT 2012)*, in October 2012 and it received the *Best paper award*.

  Further work along these lines is taking place, to generate code for a variety of low-overhead execution models, to cope with real-time constraints, and to formalize and prove the correctness of the underlying concurrent data structures. On the latter point, a paper has been accepted at the ACM Conf. PPoPP 2013.

6.9. Semantics and Implementation of Hybrid System Modelers

Participants: Marc Pouzet [contact], Timothy Bourke.

[8] http://www.esterel-technologies.com/products/scade-suite/

Zélus is a new programming language for modeling systems that mix discrete logical time and continuous time behaviors. From a user's perspective, its main originality is to extend an existing -like synchronous language with Ordinary Differential Equations (ODEs). The extension is conservative: any synchronous program expressed as data-flow equations and hierarchical automata can be composed arbitrarily with ODEs in the same source code. A dedicated type system and causality analysis ensure that all discrete changes are aligned with zero-crossing events so that no side effects or discontinuities occur during integration. Programs are statically scheduled and translated into sequential code which, by construction, runs in bounded time and space. Compilation is effected by source-to-source translation into a small synchronous subset which is processed by a standard synchronous compiler architecture. The resulting code is paired with an off-the-shelf numeric solver.

This experiment show that it is possible to build a modeler for explicit hybrid systems à la Simulink/Stateflow on top of an existing synchronous language, using it both as a semantic basis and as a target for code generation. In parallel with the software development done during the year, we investigate, in collaboration with Albert Benveniste, Benoit Caillaud (Inria Rennes) and Dassault-Systèmes the treatment of Differential Algebraic Equations (DAEs), in explicit or semi-explicit form.

This work will be presented at the *ACM Intern. Conference on Hybrid Systems: Computation and Control (HSCC 2013)* in April 2013.

POP ART Project-Team

6. New Results

6.1. Dependable Distributed Real-time Embedded Systems

Participants: Gwenaël Delaval, Pascal Fradet, Alain Girault [contact person], Emil Dumitrescu.

6.1.1. Tradeoff exploration between reliability, power consumption, and execution time

For autonomous critical real-time embedded systems (*e.g.*, satellite), guaranteeing a very high level of reliability is as important as keeping the power consumption as low as possible. We have designed an off-line ready list scheduling heuristics which, from a given software application graph and a given multiprocessor architecture (homogeneous and fully connected), produces a static multiprocessor schedule that optimizes three criteria: its *length* (crucial for real-time systems), its *reliability* (crucial for dependable systems), and its *power consumption* (crucial for autonomous systems). Our tricriteria scheduling heuristics, **TSH**, uses the *active replication* of the operations and the data-dependencies to increase the reliability, and uses *dynamic voltage and frequency scaling* to lower the power consumption [17], [11]. By running TSH on a single problem instance, we are able to provide the Pareto front for this instance in 3D, therefore exposing the user to several tradeoffs between the power consumption, the reliability and the execution time. The new contribution for 2012 has been the formulation of a new multi-criteria cost function for our ready list scheduling heuristics, such that we are able to prove rigorously that the static schedules we generate meet both the reliability constraint and the power consumption constraint.

Thanks to extensive simulation results, we have shown how TSH behaves in practice. Firstly, we have compared TSH versus an optimal Mixed Linear Integer Program on small instances; the experimental results show that TSH behaves very well compared to the the ILP. Secondly, we have compared TSH versus the ECS heuristic (Energy-Conscious Scheduling [68]); the experimental results show that TSH performs systematically better than ECS.

This is a joint work with Ismail Assayad (U. Casablanca, Morocco) and Hamoudi Kalla (U. Batna, Algeria), who both visit the team regularly.

6.2. Controller Synthesis for the Safe Design of Embedded Systems

Participants: Gwenaël Delaval [contact person], Gregor Goessler, Sebti Mouelhi.

6.2.1. Synthesis of switching controllers using approximately bisimilar multiscale abstractions

The use of discrete abstractions for continuous dynamics has become standard in hybrid systems design (see *e.g.*, [73] and the references therein). The main advantage of this approach is that it offers the possibility to leverage controller synthesis techniques developed in the areas of supervisory control of discrete-event systems [71]. The first attempts to compute discrete abstractions for hybrid systems were based on traditional systems behavioral relationships such as simulation or bisimulation, initially proposed for discrete systems most notably in the area of formal methods. These notions require inclusion or equivalence of observed behaviors which is often too restrictive when dealing with systems observed over metric spaces. For such systems, a more natural abstraction requirement is to ask for closeness of observed behaviors. This leads to the notions of approximate simulation and bisimulation introduced in [54].

These notions enabled the computation of approximately equivalent discrete abstractions for several classes of dynamical systems, including nonlinear control systems with or without disturbances, and switched systems. These approaches are based on sampling of time and space where the sampling parameters must satisfy some relation in order to obtain abstractions of a prescribed precision. In particular, the smaller the time sampling parameter, the finer the lattice used for approximating the state-space; this may result in abstractions with a very large number of states when the sampling period is small. However, there are a number of applications where sampling has to be fast; though this is generally necessary only on a small part of the state-space.

In [45] we have proposed a technique for the synthesis of safety controllers for switched systems using multi-scale abstractions that allow us to deal with fast switching while keeping the number of states in the abstraction at a reasonable level. The finest scales of the abstraction are effectively explored only when fast switching is needed, that is when the system approaches the unsafe set.

We have extended the approach of [45] to the synthesis of controllers for time-bounded reachability. Furthermore we have implemented the algorithms for safety and time-bounded reachability in COSYMA, a tool for automatic controller synthesis for incrementally stable switched systems based on multi-scale discrete abstractions. The tool accepts a description of a switched system represented by a set of differential equations and the sampling parameters used to define an approximation of the state-space on which discrete abstractions are computed. The tool generates a controller — if it exists — for the system that enforces a given safety or time-bounded reachability specification.

We are currently exploring, in the SYMBAD project, controller synthesis for switched systems based on a different approach for the construction of multi-scale abstractions. The goal is to further improve the trade-off between cost and precision.

6.2.2. Modular discrete controller synthesis

Discrete controller synthesis (DCS) [71] allows to design programs in a mixed imperative/declarative way. From a program with some freedom degrees left by the programmer (*e.g.*, free controllable variables), and a temporal property to enforce which is not *a priori* verified by the initial program, DCS tools compute off-line automatically a *controller* which will constrain the program (by *e.g.*, giving values to controllable variables) such that, whatever the values of inputs from the environment, the *controlled program* satisfies the temporal property.

Our motivation *w.r.t.* DCS concerns its modular application, improving the scalability of the technique by using contract enforcement and abstraction of components. Moreover, our aim is to integrate DCS into a compilation chain, and thereby improve its usability by programmers, not experts in discrete control. This work has been implemented into the HEPTAGON/BZR language and compiler [50]. This work is done in collaboration with Hervé Marchand (VERTECS team from Rennes) and Eric Rutten (SARDES team from Grenoble).

The implemented tool allows the generation of the synthesized controller under the form of an HEPTAGON node, which can in turn be analyzed and compiled, together with the HEPTAGON source from which it has been generated. This full integration allows this method to aim different target languages (currently C, JAVA or VHDL), and its integrated use in different contexts.

A formal semantics of BZR has been defined, taking into account its underlying nondeterminism related to the presence of controllable variables. A new implementation has been achieved, including an abstraction method based on [47]. We have used BZR for demonstrating the use of Control Theory and Techniques to the administration of computing systems in a closed-loop management [19].

6.3. Automatic Distribution of Synchronous Programs

Participants: Gwenaël Delaval [contact person], Alain Girault, Gregor Goessler, Xavier Nicollin, Gideon Smeding.

6.3.1. Modular distribution

Synchronous programming languages describe functionally centralized systems, where every value, input, output, or function is always directly available for every operation. However, most embedded systems are nowadays composed of several computing resources. The aim of this work is to provide a language-oriented solution to describe *functionally distributed reactive systems*. This research is conducted within the Inria large scale action SYNCHRONICS and is a joint work with Marc Pouzet (ENS, PARKAS team from Rocquencourt) and Xavier Nicollin (Grenoble INP, VERIMAG lab).

We are working on type systems to formalize, in a uniform way, both the clock calculus and the location calculus of a synchronous data-flow programming language (the HEPTAGON language, inspired from LUCID SYNCHRONE [38]). On one hand, the clock calculus infers the clock of each variable in the program and checks the clock consistency: *e.g.*, a time-homogeneous function, like +, should be applied to variables with identical clocks. On the other hand, the location calculus infers the spatial distribution of computations and checks the spatial consistency: *e.g.*, a centralized operator, like +, should be applied to variables located at the same location. Compared to the PhD of Gwenaël Delaval [48], [49], the goal is to achieve *modular* distribution. By modular, we mean that we want to compile each function of the program into a single function capable of running on any computing location. We make use of our uniform type system to express the computing locations as first-class abstract types, exactly like clocks, which allows us to compile a typed variable (typed by both the clock and the location calculi) into `if ... then ... else ...` structures, whose conditions will be valuations of the clock and location variables.

We currently work on an example of software-defined radio. We have shown on this example how to use a modified clock calculus to describe the localisation of values as clocks, and the architecture as clocks (for the computing resources) and their relations (for communication links).

6.3.2. Distribution of synchronous programs under real-time constraints

With the objective to distribute synchronous data-flow programs (*e.g.*, LUSTRE) over GALS architectures, such that the difference between the original and synchronous systems satisfy given bounds, we have developed a quantitative clock calculus to (1) describe timing properties of the architecture's clock domain, and (2) describe the acceptable difference between the original and distributed programs. The clock calculus is inspired by the network calculus [67], with the difference that clocks are described only with respect to one-another, not with respect to real-time.

As a first result, we have applied our clock calculus to analyze the properties of periodic synchronous data-flow programs executed on a network of processors. Because our clock calculus is relational, it can model and preserve correlated variations of streams. In particular, the common case of a data-flow system that splits a stream for separate treatment, and joins them afterwards, this analysis yields more precise result than comparable methods [24].

We have been able to use the clock calculus as an abstract domain to perform abstract interpretation of synchronous boolean data-flow programs and their distribution on synchronous nodes that communicate asynchronously by sampling shared memory. The analysis discovers the relative clock drift of all clocks of the distributed system as well as bounds on the distance from the original program.

In case the guaranteed maximal distance is too large, we provide methods to synthesize bounds on the relative drift of the architecture's clocks that ensure an acceptable distance. Given the synthesized bounds, we use the known clock drifts and program behavior to synthesize light weight protocols.

6.4. New Programming Languages for Embedded Systems

Participants: Alain Girault [contact person], Pascal Fradet, Vagelis Bebelis, Bertrand Jeannet, Peter Schrammel.

6.4.1. Analysis and scheduling of parametric dataflow models

Recent data-flow programming environments support applications whose behavior is characterized by dynamic variations in resource requirements. The high expressive power of the underlying models (*e.g.*, Kahn Process Networks, the CAL actor language) makes it challenging to ensure predictable behavior. In particular, checking *liveness* (*i.e.*, no part of the system will deadlock) and *boundedness* (*i.e.*, the system can be executed in finite memory) is known to be hard or even undecidable for such models. This situation is troublesome for the design of high-quality embedded systems.

We have introduced the *schedulable parametric data-flow (SPDF)* MoC for dynamic streaming applications [20]. SPDF extends the standard dataflow model by allowing rates to be parametric (*e.g.*, of the form $2xy$, where x and y are parameters, the value of which can change at run-time). SPDF was designed to be statically analyzable while retaining sufficient expressive power. We formulated sufficient and general static criteria for boundedness and liveness. In SPDF, parameters can be changed dynamically even within iterations. The safety of dynamic parameter changes can be checked and their implementation made explicit in the graph. These different analyses are made possible using well-defined static operations on symbolic expressions. The same holds for quasi-static scheduling which is the first step towards code generation for multi-core systems.

We are now focusing on the parallel scheduling of parametric dataflow models. We have proposed a generic and flexible framework to generate parallel ASAP schedules targeted to the new STHORM many-core platform designed by STMicroelectronics. The parametric dataflow graph is associated with generic or user-defined specific constraints aimed at minimizing, timing, buffer sizes, power consumption, or other criteria. The scheduling algorithm executes with minimal overhead and can be adapted to different scheduling policies just by changing some constraints. The safety of both the dataflow graph and constraints can be checked statically and all schedules are guaranteed to be bounded and deadlock free. Our case studies are video decoders for high definition video streaming such as VC-1 or HEVC.

This research is the central topic of Vagelis Bebelis' PhD thesis. It is conducted in collaboration with STMicroelectronics.

6.5. Static Analysis and Abstract Interpretation

Participants: Alain Girault, Bertrand Jeannet [contact person], Peter Schrammel.

6.5.1. Translating data-flow languages for hybrid systems simulation to hybrid automata for hybrid systems verification

Hybrid systems are used to model embedded computing systems interacting with their physical environment. There is a conceptual mismatch between high-level hybrid system languages like SIMULINK[34], which are used for simulation, and hybrid automata, the most suitable representation for safety verification. Indeed, in simulation languages the interaction between discrete and continuous execution steps is specified using the concept of zero-crossings, whereas hybrid automata exploit the notion of staying conditions.

In the context of the INRIA large scale action SYNCHRONICS (see §8.1.1.1), we studied how to translate the ZELUS hydrid data-flow language [34] developped in this project into logico-numerical hybrid automata by carefully pointing out this issue. We investigated various zero-crossing semantics, proposed a sound translation, and discussed to which extent the original semantics is preserved.

This work is part of the PhD thesis of Peter Schrammel and was presented at the conference HSCC'2012 (Hybrid Systems: Computation and Control) [22], [27].

6.5.2. Abstract Acceleration of general linear loops

We investigated abstract acceleration techniques for computing loop invariants for linear loops with linear assignments in their body and guards defined by the conjunction of linear inequalities.

While standard abstract interpretation considers over approximations over the set of reachable states at any loop iteration, and relies on extrapolation (*a.k.a.* widening) for making the analysis converge, abstract acceleration captures the effect of the loop with a single, non-iterative transfer function applied to the reachable states at the loop head. The concept of abstract acceleration has already been applied to restricted form of linear loops, by us [16] and others [58], and extended to logico-numerical loops [16]; the novelty here is to investigate general linear loops.

[34] http://www.mathworks.com

The main idea we developped is to over-approximate the set of transformation matrices associated to any number of iterations of the loop body and to apply this "accelerated" transformation to the initial states. This over-approximation is based on the Jordan normal form decomposition that allows deriving closed form symbolic expressions for the entries of the matrix modeling the effect of exactly n iterations of the loop. We then discover linear relationships between the symbolic expressions that hold for any number of iterations, and we obtain a set of matrices described by a polyhedra on its coefficients, which can be applied to a set of vectors also described by a convex polyhedra.

We also developed a technique to take into account the guard of the loop by bounding the number of loop iterations, which relies again on the Jordan normal form decomposition.

These ideas were implemented and evaluated on a series of simple loops, alone or inside outer loops, exhibiting classical behaviors: polynomial, stable and unstable exponential, inward spirals (damped oscillators), ... Our approach enables proofs that are out of the reach of most other techniques, that are either too unprecise (classical abstract interpretation) or limited to a restricted class of loops, *e.g.*, translation with resets in the case of abstract acceleration, or stable loops (in the sense of control theory) for ellipsoid methods.

This work was initiated during a visit to the University of Colorado-Boulder and is under review.

6.5.3. Logico-Numerical Max-Strategy Iteration

Strategy iteration methods aim at solving fixed point equations and are an alternative to abstract acceleration for fighting against the loss of precision incurred by extrapolation in classical interpretation. It has been shown that they improve precision in static analysis based on abstract interpretation and template abstract domains, *e.g.*, intervals, octagons or template polyhedra. However, they are limited to numerical programs.

We investigated a method for applying max-strategy iteration to logico-numerical programs, that is, programs with numerical and Boolean variables, without explicitly enumerating the Boolean state space. The method is optimal in the sense that it computes the *least fixed pointw.r.t.* the abstract domain.

Our experiments showed that the resulting logico-numerical max-strategy iteration gains one order of magnitude in terms of efficiency in comparison to the purely numerical approach while being almost as precise. Moreover, they are the first experimental results of applying max-strategy iteration to larger programs. This work has been accepted at VMCAI'2013 [23].

6.6. Component-Based Construction

Participants: Alain Girault, Gregor Goessler [contact person], Roopak Sinha, Gideon Smeding.

6.6.1. Incremental converter synthesis

We have proposed and implemented a formal incremental converter-generation algorithm for system-on-chip (SoC) designs. The approach generates a converter, if one exists, to control the interaction between multiple intellectual property (IP) protocols with possible control and data mismatches, and allows pre-converted systems to be re-converted with additional IPs in the future. IP protocols are represented using labeled transition systems (LTS), a simple but elegant abstraction framework which can be extracted from and converted to standard IP description languages such as VHDL. The user can provide control properties, each stated as an LTS with accepting states, to describe desired aspects of the converted system, including fairness and liveness. Furthermore, data specifications can be provided to bound data channels between interacting IPs such that they do not over/under flow. The approach takes into account the uncontrollable environment of a system by allowing users to identify signals exchanged between the SoC and the environment, which the converter can neither suppress nor generate.

Given these inputs, the conversion algorithm first computes the reachable state-space of a maximal non-deterministic converter that ensures (i) the satisfaction of the given data specifications and (ii) the trace equivalence with the given control specifications, using a greatest fix-point computation. It then checks, using the standard algorithm for Büchi games, whether the converter can ensure the satisfaction of the given control specifications (reachability of accepting states) regardless of how the environment behaves. If this is

found to be true, deterministic converters can be automatically generated from the maximal non-deterministic converter generated during the first step. The algorithm is proven to be sound and complete, with a polynomial complexity in the state-space sizes of given IP protocols and specifications. It is also shown that it can be used for incremental design of SoCs, where IPs and specifications are added to an SoC in steps. Incremental design allows to constrain the combinatorial explosion of the explored state-space in each step, and also reduces on-chip wire congestion by decentralizing the conversion process.

A Java implementation has been created, and experimental results show that the algorithm can handle complex IP mismatches and specifications in medium to large AMBA based SoC systems. Future work involves creating a library of commonly-encountered specifications in SoC design such as sharing of control signals between interacting IPs using buffers, signal lifespans, and the generation of optimal converters based on quantitative criteria such as minimal power usage.

This work has been done within the AFMES associated team with the Electric and Computer Engineering Department of the University of Auckland.

6.6.2. Analysis of logical causality

The failure of one component may entail a cascade of failures in other components; several components may also fail independently. In such cases, elucidating the exact scenario that led to the failure is a complex and tedious task that requires significant expertise.

The notion of causality *(did an event e cause an event e'?)* has been studied in many disciplines, including philosophy, logic, statistics, and law. The definitions of causality studied in these disciplines usually amount to variants of the counterfactual test "e is a cause of e' if both e and e' have occurred, and in a world that is as close as possible to the actual world but where e does not occur, e' does not occur either". Surprisingly, the study of logical causality has so far received little attention in computer science, with the notable exception of [62] and its instantiations. However, this approach relies on a causal model that may not be known, for instance in presence of black-box components.

In [6] we have proposed a formal framework for reasoning about causality, based on black-box components interacting according to well identified *interaction models* [5].

We are currently extending to framework to other models of computation and communication, in particular, to timed automata, and developing a refinement of our original approach that reduces the number of false positives.

6.6.3. A Theory of fault recovery for component-based models

In [35][18] we have introduced a theory of fault recovery for component-based models. A model is specified in terms of a set of atomic components that are incrementally composed and synchronized by a set of glue operators. We have defined what it means for such models to provide a recovery mechanism, so that the model converges to its normal behavior in the presence of faults (*e.g.*, in self-stabilizing systems). We have identified corrector components whose presence in a model is essential to guarantee recovery after the occurrence of faults. We have also formalized component based models that effectively separate recovery from functional concerns. We have shown that any model that provides fault recovery can be transformed into an equivalent model, where functional and recovery tasks are modularized in different components.

6.7. Aspect-Oriented Programming

Participants: Dmitry Burlyaev, Pascal Fradet [contact person], Alain Girault.

The goal of Aspect-Oriented Programming (AOP) is to isolate aspects (such as security, synchronization, or error handling) which cross-cut the program basic functionality and whose implementation usually yields tangled code. In AOP, such aspects are specified separately and integrated into the program by an automatic transformation process called *weaving*.

Although this paradigm has great practical potential, it still lacks formalization and undisciplined uses make reasoning on programs very difficult. Our work on AOP addresses these issues by studying foundational issues (semantics, analysis, verification) and by considering domain-specific aspects (availability, fault tolerance or refinement aspects) as formal properties.

6.7.1. Aspects preserving properties

Aspect Oriented Programming can arbitrarily distort the semantics of programs. In particular, weaving can invalidate crucial safety and liveness properties of the base program.

We have identified categories of aspects that preserve some classes of properties [13]. Our categories of aspects comprise, among others, observers, aborters, and confiners. For example, observers do not modify the base program's state and control-flow (*e.g.*, persistence, profiling, and debugging aspects). These categories are defined formally based on a language independent abstract semantic framework. The classes of properties are defined as subsets of LTL for deterministic programs and CTL* for non-deterministic ones. We have formally proved that, for any program, the weaving of any aspect in a category preserves any property in the related class. In a second step, we have designed for each aspect category a specialized aspect language which ensures that any aspect written in that language belongs to the corresponding category. These languages preserve the corresponding classes of properties by construction.

This work was conducted in collaboration with Rémi Douence from the ASCOLAINRIA team at École des Mines de Nantes.

6.7.2. Fault tolerance aspects

In the recent years, we have studied the implementation of specific fault tolerance techniques in real-time embedded systems using program transformation [1]. We are now investigating the use of fault-tolerance aspects in digital circuits. To this aim, we consider program transformations for hardware description languages (HDL). Our goal is to design an aspect language allowing users to specify and tune a wide range of fault tolerance techniques, while ensuring that the woven HDL program remains synthesizable. The advantage would be to produce fault-tolerant circuits by specifying fault-tolerant strategies separately from the functional specifications.

We have reviewed the different fault tolerant techniques used in integrated circuits: concurrent error detection, error detecting and correcting codes (Hamming, Berger codes, ...), spatial and time redundancy. We have designed a simple hardware description language inspired from LUSTRE and Lucid Synchrone. It is a core functional language manipulating synchronous boolean streams. Faults are represented by bit flips and we take into account all fault models of the form "at most 1 faults within n clock signals". The language semantics as well as the fault model have been formalized in Coq. Many basic (library) properties have been shown on that language.

We are currently expressing different variants of triple modular redundancy (TMR) as program transformations. We are also studying optimizations to prevent the insertion of useless voters in TMR. The next step is to prove that these transformations make the programs fault tolerant *w.r.t.* specific fault models. Further work also includes the study of mixed techniques (*e.g.*, spatial and time redundancy), their high level specification using an AOP-like language and their implementation as transformations.

S4 Project-Team

6. New Results

6.1. Petri Nets and their Synthesis

Participants: Eric Badouel, Philippe Darondeau.

6.1.1. Deciding Selective Declassification of Petri Nets

In [20], we consider declassification, as effected by downgrading actions D, in the context of intransitive non-interference encountered in systems that consist of high-level (secret) actions H and low-level (public) actions L. In a previous work, we had shown the decidability of a strong form of declassification, by which D contains only a single action d declassifying all H actions at once. We continue this study by considering selective declassification, where each transition d in D can declassify a subset $H(d)$ of H. The decidability of this more flexible, application-prone declassification framework is proved in the context of (possibly unbounded) Petri nets with possibly infinite state spaces.

6.1.2. Petri Net Distributability

A Petri net is distributed if, given an allocation of transitions to (geographical) locations, no two transitions at different locations share a common input place. A system is distributable if there is some distributed Petri net implementing it. We address in [21] the question of which systems can be distributed, while respecting a given allocation. We state the problem formally and discuss several examples illuminating — to the best of our knowledge — the current status of this work.

6.1.3. Petri Net Reachability Graphs: Decidability Status of First Order Prioperties

We investigated in [13] the decidability and complexity status of model-checking problems on unlabelled reachability graphs of Petri nets by considering first-order, modal and pattern-based languages without labels on transitions or atomic propositions on markings. We have considered several parameters to separate decidable problems from undecidable ones. Not only were we able to provide precise borders and a systematic analysis, but we also demonstrated the robustness of our proof techniques.

6.1.4. α-reconstructibility of Workflow Nets

The α-algorithm is a process mining algorithm, introduced by van der Aalst et al, that constructs a workflow net from an event log. A class of nets, the structured workflow nets, was recognized to be reconstructible by algorithm α from their language (or a representative subset of it). In [14] we assessed more precisely the α-algorithm we provided a characterization of the class of the workflow nets that are discovered by α.

6.2. Hybrid Modeling

Participants: Albert Benveniste, Benoît Caillaud.

Hybrid system modelers have become a corner stone of complex embedded system development. Embedded systems include not only control components and software, but also physical devices. In this area, Simulink is a de facto standard design framework, and Modelica a new player. However, such tools raise several issues related to the lack of reproducibility of simulations (sensitivity to simulation parameters and to the choice of a simulation engine). In [10] we propose using techniques from non-standard analysis to define a semantic domain for hybrid systems. Non-standard analysis is an extension of classical analysis in which infinitesimal (the ϵ and η in the celebrated generic sentence $\forall\epsilon\exists\eta$... of college maths) can be manipulated as first class citizens. This approach allows us to define both a denotational semantics, a constructive semantics, and a Kahn Process Network semantics for hybrid systems, thus establishing simulation engines on a sound but flexible mathematical foundation. These semantics offer a clear distinction between the concerns of the numerical

analyst (solving differential equations) and those of the computer scientist (generating execution schemes). We also discuss a number of practical and fundamental issues in hybrid system modelers that give rise to non-reproducibility of results, non-determinism, and undesirable side effects. Of particular importance are cascaded mode changes (also called "zero-crossings" in the context of hybrid systems modelers). This work has taken place in the framework of the Synchronics large scale initiative (see section 7.1.1).

6.3. Component-Based Design

Participants: Albert Benveniste, Benoît Caillaud, Sophie Pinchinat.

6.3.1. Application of Interface Theories to the Separate Compilation of Synchronous Programs

We study in [15], [26] the problem of separate compilation, i.e., the generation of modular code, for the discrete time part of block-diagrams formalisms such as Simulink, Modelica, or Scade. Code is modular in that it is generated for a given composite block independently from context (i.e., without knowing in which diagrams the block is to be used) and using minimal information about the internals of the block. Just using off-the-shelf C code generation (e.g., as available in Simulink) does not provide modular code. Separate compilation was solved by Lublinerman et al. for the special case of single clocked diagrams, in which all signals are updated at a same unique clock. For the same case, Pouzet and Raymond proposed algorithms that scale-up properly to real-size applications. The technique of Lublinerman et al. was extended to some classes of multi-clocked and timed diagrams. We study this problem in its full generality and we show that it can be cast to a special class of controller synthesis problems by relying on recently proposed modal interface theories.

6.3.2. Contracts for System Design

Systems design has become a key challenge and differentiating factor over the last decades for system companies. Aircrafts, trains, cars, plants, distributed telecommunication military or health care systems, and more, involve systems design as a critical step. Complexity has caused system design times and costs to go severely over budget so as to threaten the health of entire industrial sectors. Heuristic methods and standard practices do not seem to scale with complexity so that novel design methods and tools based on a strong theoretical foundation are sorely needed. Model-based design as well as other methodologies such as layered and compositional design have been used recently but a unified intellectual framework with a complete design flow supported by formal tools is still lacking albeit some attempts at this framework such as Platform-based Design have been successfully deployed. Recently an “orthogonal” approach has been proposed that can be applied to all methodologies proposed thus far to provide a rigorous scaffolding for verification, analysis and abstraction/refinement: contract-based design. Several results have been obtained in this domain but a unified treatment of the topic that can help in putting contract-based design in perspective is still missing. In [25], we intend to provide such treatment where contracts are precisely defined and characterized so that they can be used in design methodologies such as the ones mentioned above with no ambiguity. In addition, the paper provides an important link between interfaces and contracts to show similarities and correspondences. Examples of the use of contracts in system design are provided, including one based on Modal Interfaces, using the Mica tool (see section 5.1).

6.3.3. Ensuring Reachability by Design

In [18], [28], we study the independent implementability of reachability properties, which are in general not compositional. We consider modal specifications, which are widely acknowledged as suitable for abstracting implementation details of components while exposing to the environment relevant information about cross-component interactions. In order to obtain the required expressivity, we extend them with marked states to model states to be reached. We then develop an algebra with both logical and structural composition operators ensuring reachability properties by construction.

6.3.4. Modal event-clock specifications for timed component-based design

Modal specifications are classic, convenient, and expressive mathematical objects to represent interfaces of component-based systems. However, time is a crucial aspect of systems for practical applications, e.g. in the area of embedded systems. And yet, only few results exist on the design of timed component-based systems. In [11], we propose a timed extension of modal specifications, together with fundamental operations (conjunction, product, and quotient) that enable reasoning in a compositional way about timed system. The specifications are given as modal event-clock automata, where clock resets are easy to handle. We develop an entire theory that promotes efficient incremental design techniques.

6.4. Automata, Games and Logics for Supervisory Control and System Synthesis

Participants: Philippe Darondeau, Bastien Maubert, Sophie Pinchinat.

6.4.1. Distributed Control of Discrete Event Systems: A First Step

To initiate a discussion on the modeling requirements for distributed control of discrete-event systems, a partially-automated region-based methodology is presented in [23]. The methodology is illustrated via a well-known example from distributed computing: the dining philosophers.

6.4.2. Enforcing Opacity of Regular Predicates on Modal Transition Systems

In [22] we considered the following problem: Given a labelled transition system LTS partially observed by an attacker, and a regular predicate Sec over the runs of LTS, enforcing opacity of the secret Sec in LTS means computing a supervisory controller K such that an attacker who observes a run of K/LTS cannot ascertain that the trace of this run belongs to Sec based on the knowledge of LTS and K. We then lifted the problem from a single labelled transition system LTS to the class of all labelled transition systems specified by a modal transition system MTS. The lifted problem is to compute the maximally permissive controller K such that Sec is opaque in K/LTS for every labelled transition system LTS which is a model of MTS. The situations of the attacker and of the controller are dissymmetric: at run time, the attacker may fully know LTS and K whereas the controller knows only MTS and the sequence of actions executed so far by the unknown LTS. We addressed the problem in two cases. Let Σ_a denote the set of actions that can be observed by the attacker, and let Σ_c and Σ_o denote the sets of actions that can be controlled and observed by the controller, respectively. We provided optimal and regular controllers that enforce the opacity of regular secrets when $\Sigma_c \subseteq \Sigma_o \subseteq \Sigma_a = \Sigma$. We also provided optimal and regular controllers that enforce the opacity of regular upper-closed secrets ($Sec = Sec.\Sigma^*$) when $\Sigma_a \subseteq \Sigma_c \subseteq \Sigma_o = \Sigma$.

6.4.3. Analysis of partially observed recursive tile systems

The analysis of discrete event systems under partial observation is an important topic, with major applications such as the detection of information flow and the diagnosis of faulty behaviors. In [19], we consider recursive tile systems, which are infinite systems generated by a finite collection of finite *tiles*, a simplified variant of deterministic graph grammars. Recursive tile systems are expressive enough to capture classical models of recursive systems, such as the pushdown systems and the recursive state machines. They are infinite-state in general and therefore standard powerset constructions for monitoring do not always apply. We exhibit computable conditions on recursive tile systems and present non-trivial constructions that yield effective computation of the monitors. We apply these results to the classic problems of opacity and diagnosability.

6.4.4. Uniform Strategies

In [29], we consider turn-based game arenas for which we investigate uniformity properties of strategies. These properties involve bundles of plays, that arise from some semantical motive. Typically, we can represent constraints on allowed strategies, such as being observation-based. We propose a formal language to specify uniformity properties and demonstrate its relevance by rephrasing various known problems from the literature. Note that the ability to correlate different plays cannot be achieved by any branching-time logic if not

equipped with an additional modality, so-called R in this contribution. We also study an automated procedure to synthesize strategies subject to a uniformity property, which strictly extends exitsting results based on, say standard temporal logics. We exhibit a generic solution for the synthesis problem provided the bundles of plays rely on any binary relation definable by a finite state transducer. This solution yields a non-elementary procedure.

6.4.5. Emptiness Of Alternating Parity Tree Automata Using Games With Imperfect Information

In [30], we focus on the emptiness problem for alternating parity tree automata. The usual technique to tackle this problem first removes alternation, going to non-determinism, and then checks emptiness by reduction to a two-player perfect-information parity game. In this note, we give an alternative roadmap to this problem by providing a direct reduction to the emptiness problem to solving an imperfect-information two-player parity game.

6.4.6. On timed alternating simulation for concurrent timed games

We address in [12] the problem of alternating simulation refinement for concurrent timed games (*TG*). We show that checking timed alternating simulation between *TG* is EXPTIME-complete, and provide a logical characterization of this preorder in terms of a meaningful fragment of a new logic, *TAMTL**. *TAMTL** is an action-based timed extension of standard alternating-time temporal logic *ATL**, which allows to quantify over strategies where the designated coalition of players is not responsible for blocking time. While for full *TAMTL**, model-checking *TG* is undecidable, we show that for its fragment *TAMTL*, corresponding to the timed version of *ATL*, the problem is instead decidable and in EXPTIME.

TRIO Project-Team

6. New Results

6.1. Evaluation and optimal dimensioning of real-time systems

- **Code analyses and advanced visualization of software in real-time**

  Participants: Pierre Caserta, Olivier Zendra

  Last years, strong developments for our instrumentation, tracer and analyzer, had been performed, allowing us to really enter the experimental phase and getting first interesting results. A thorough state of the art had also been written.

  This state of the art paper had finally been published in TVCG, a leading journal in computer visualization. Thanks to the experimental setup efforts of previous years, we had been able in 2011 to conduct good experiments. We had designed and implemented a new way to visualize relations between software elements. These relations include static relations (is-a, direct heir, caller, callee, etc.) and dynamic ones (runtime caller, runtime callee). Our new relation visualization comprises a new way of placing way points so as to significantly decrease spatial and visual clutter when visualizing software systems with large numbers (thousands up to millions) of relations. This had lead to a publication in VISSOFT, one of the most recognized conferences in the software visualization domain, as well as a Best Poster in ECOOP, one of the most recognized conferences in the object-oriented domain. The important design and implementation work we had realized on the tracing and analysis software also lead to the publication of our method in ICOOOLPS 2011.

  This year, in 2012, we published our instrumentation and tracing method in Elsevier's Science of Computer Programming journal [9].

  Work has been going onto analyze polymorphism in Java programs, answering an apparently simple yet so far unanswered question: how much polymorphism is there actually in Java programs. This is of paramount importance, since a lot of work occur around polymorphism, which is an important concept, but no one is currently able to tell how much it impact programs in real life. We have begun writing this paper in cooperation with the LIRMM lab in Montpellier. In addition, we are in the process of finishing work pertaining to analyzing program evolutions, looking at differences between versions, and analyzing how dynamic metrics and static metrics correlate to evolution rate.

  Work in this domain has also lead to the writing and successful defense of Pierre Caserta's PhD thesis, titled "Analyse statique et dynamique de code et visualisation des logiciels via la métaphore de la ville : contribution à l'aide à la compréhension des programmes", on 7th December 2012 [7].

  A web site was also designed to publicize our work on the VITRAIL project.

- **Open Power and Energy Optimization PLatform and Estimator**

  Participants: Fabrice Vergnaud, Jérôme Vatrinet, Kévin Roussel, Olivier Zendra.

  Work in this domain was performed in the context of the ANR Open-PEOPLE (Open Power and Energy Optimization PLatform and Estimator) project, financed since the very end of 2008. Inria Nancy Grand Est is responsible for the software part of the platform and is involved in memory management for low-power issues. Work in this project begun in April 2009 (kick-off meeting). We have finished setting up the very important infrastructure for the software part of the Open-PEOPLE platform. We have finished expressing the requirements for the platform, in order to start the actual developments and the actual integration of tools provided by the different partners. In 2011, we have finished expressing the platform architecture and user interface (GUI). We have also finished implementing the part of the software platform that is the remote control to the hardware platform. We finally have finished implementing the core of the software platform and canonical

models handling. This work lead to several technical and the several presentations and posters in conferences.

This year was the result harvesting for our project, in terms of development. We finished the design and implementation of the PCMD (Power Consumption Model Development) and the PCAO (Power Consumption Analysis and Optimization) parts of the software platform, as well as the external tools integration work. We also designed and implemented the Open-PEOPE model sharing website. Again, several demos and publications in conferences resulted [13], [21], [22].

- **Operator calculus and conceptation of algorithms for optimisatio of multi-constraints problems**

  Participants: Jamila Ben Slimane, Hugo Cruz-Sanchez, Bilel Nefzi, René Schott, Ye-Qiong Song

  R. Schott and G. Stacey Staples proposed a solution based on operator calculus for graphs with multi-constraints [26]. These constraints are not necessarily linear or positive. This approach was developed for realistic problems like:

  – configuration of satellites proposing a high-quality coverage [14];
  – optimal utilisation of ressources in hospitals;
  – optimal management in sensor networks [25].

  This work was the result of the collaboration of our team with MADYNES team, LPMA (Laboratoire de Probabilités et Modèles Aléatoires, Paris 6 et 7) and University of Illinois at Edwardsville.

6.2. Real-time analysis

- **Scheduling of tasks in automotive multicore ECUs**

  Participants: Aurélien Monot, Nicolas Navet, Françoise Simonot-Lion.

  As the demand for computing power is quickly increasing in the automotive domain, car manufacturers and tier-one suppliers are gradually introducing multicore ECUs in their electronic architectures. Additionally, these multicore ECUs offer new features such as higher levels of parallelism which ease the respect of safety requirements such as the ISO 26262 and the implementation of other automotive use-cases. These new features involve also more complexity in the design, development and verification of the software applications. Hence, car manufacturers and suppliers will require new tools and methodologies for deployment and validation. We address the problem of sequencing numerous elementary software components, called runnables, on a limited set of identical cores. We show how this problem can be addressed as two sub-problems, partitioning the set of runnables and building the sequencing of the runnables on each core, which problems cannot be solved optimally due to their algorithmic complexity. We then present low complexity heuristics to partition and build sequencer tasks that execute the runnable set on each core, and derive lower bounds on their efficiency (i.e., competitive ratio). Finally, we address the scheduling problem globally, at the ECU level, by discussing how to extend this approach in the case where other OS tasks are scheduled on the same cores as the sequencer tasks. An article providing a summary of this line of work has been published in IEEE TII [12].

- **Probabilistically analysable real-time system**

  Participants: Liliana Cucu-Grosjean, Adriana Gogonel, Codé Lo, Luca Santinelli, Dorin Maxim and Cristian Maxim.
  The adoption of more complex hardware to respond to the increasing demand for computing power in next- generation systems exacerbates some of the limitations of static timing analysis for the estimation of the worst-case execution time (WCET) estimation. In particular, the effort of acquiring (1) detail information on the hardware to develop an accurate model of its execution latency as well as (2) knowledge of the timing behaviour of the program in the presence of varying hardware conditions, such as those dependent on the history of previously executed instructions. These

problems are also known as the timing analysis walls. The probabilistic timing analysis, a novel approach to the analysis of the timing behaviour of next-generation real-time embedded systems, provides answers to timing analysis walls. In [10], [15] we have showed how the probabilistic timing analysis attacks the timing analysis walls. We have also presented experimental evidence that shows how probabilistic timing analysis reduces the extent of knowledge about the execution platform required to produce probabilistically-safe and tight WCET estimations.

Based on existing estimations of WCET or minimal inter-arrival time, we may propose different probabilistic schedulability analyses [19], [11].

- **Statistical analysis of real-time systems**

  Participants: Liliana Cucu-Grosjean, Adriana Gogonel, Lu Yue, Thomas Nolte [Malardelan University], Rob Davis, Ian Bate [University of York], Michael Houston, Guillem Bernat [Rapita].

  The response time analysis of real-time systems usually needs the knowledge of WCET estimation and this knowledge is not always available, e.g., because of intelectual property issues. This problem may be avoided by estimating statistically either the WCET of a task [18], the inter-arrival time [17] or the response time of each task [23].

- **Probabilistic Component-based ApproachesParticipants:** Luca Santinelli, Patrick Meumeu Yomsi, Dorin Maxim, Liliana Cucu-Grosjean.

  We have proposed a probabilistic component-based model which abstracts in the interfaces both the functional and non-functional requirements of such systems. This approach allows designers to unify in the same framework probabilistic scheduling techniques and compositional guarantees that go from soft to hard real- time. We have provided sufficient schedulability tests for task systems using such framework when the scheduler is either preemptive fixed-priority or earliest deadline first. These results were published in [16].

VERTECS Project-Team

6. New Results

6.1. Verification

6.1.1. Probabilistic ω-automata

Participant: Nathalie Bertrand.

Probabilistic ω-automata are a variant version of nondeterministic automata over infinite words where all choices are resolved by probabilistic distributions. Acceptance of a run for an infinite input word can be defined using traditional acceptance criteria for ω-automata, such as Büchi, Rabin or Streett conditions. The accepted language of a probabilistic ω-automata is then defined by imposing a constraint on the probability measure of the accepting runs. Together with Christel Baier and Marcus Grösser from TU Dresden, we studied a series of fundamental properties of probabilistic ω-automata with three different language-semantics: (1) the probable semantics that requires positive acceptance probability, (2) the almost-sure semantics that requires acceptance with probability 1, and (3) the threshold semantics that relies on an additional parameter λ in]0,1[that specifies a lower probability bound for the acceptance probability. We provided a comparison of probabilistic ω-automata under these three semantics and nondeterministic ω-automata concerning expressiveness and efficiency. Furthermore, we addressed closure properties under the Boolean operators union, intersection and complementation and algorithmic aspects, such as checking emptiness or language containment. This work was published in Journal of the ACM [6].

6.1.2. Petri nets reachability graphs

Participant: Christophe Morvan.

In the article [10], we investigate the decidability and complexity status of model-checking problems on unlabelled reachability graphs of Petri nets by considering first-order and modal languages without labels on transitions or atomic propositions on markings. We consider several parameters to separate decidable problems from undecidable ones. Not only are we able to provide precise borders and a systematic analysis, but we also demonstrate the robustness of our proof techniques.

6.1.3. Frequencies in timed automata

Participant: Amélie Stainer.

A quantitative semantics for infinite timed words in timed automata based on the frequency of a run was introduced earlier by Bertrand, Bouyer, Brihaye and Stainer. Unfortunately, most of the results are obtained only for one-clock timed automata because the techniques do not allow to deal with some phenomenon of convergence between clocks. On the other hand, the notion of forgetful cycle was introduced by Basset and Asarin, in the context of entropy of timed languages, and seems to detect exactly these convergences. In [20], we investigate how the notion of forgetfulness can help to extend the computation of the set of frequencies to n-clock timed automata.

6.1.4. Bounded satisfiability for PCTL

Participant: Nathalie Bertrand.

While model checking PCTL for Markov chains is decidable in polynomial-time, the decidability of PCTL satisfiability, as well as its finite model property, are long standing open problems. While general satisfiability is an intriguing challenge from a purely theoretical point of view, we argue that general solutions would not be of interest to practitioners: such solutions could be too big to be implementable or even infinite. Inspired by bounded synthesis techniques, we turn to the more applied problem of seeking models of a bounded size: we restrict our search to implementable – and therefore reasonably simple – models. In [14] and together with John Fearnley and Sven Schewe from University of Liverpool, we propose a procedure to decide whether or not a given PCTL formula has an implementable model by reducing it to an SMT problem. We have implemented our techniques and found that they can be applied to the practical problem of sanity checking – a procedure that allows a system designer to check whether their formula has an unexpectedly small model.

6.1.5. Graph transformation systems

Participant: Nathalie Bertrand.

In [13], we study decidability issues for reachability problems in graph transformation systems, a powerful infinite-state model. For a fixed initial configuration, we consider reachability of an entirely specified configuration and of a configuration that satisfies a given pattern (coverability). The former is a fundamental problem for any computational model, the latter is strictly related to verification of safety properties in which the pattern specifies an infinite set of bad configurations. In this paper we reformulate results obtained, e.g., for context-free graph grammars and concurrency models, such as Petri nets, in the more general setting of graph transformation systems and study new results for classes of models obtained by adding constraints on the form of reduction rules.

6.2. Active and passive testing

6.2.1. More testable properties

Participants: Thierry Jéron, Hervé Marchand.

Testing remains a widely used validation technique for software systems. However, recent needs in software development (e.g., in terms of security concerns) may require to extend this technique to address a larger set of properties. In [11], we explore the set of testable properties within the Safety-Progress classification where testability means to establish by testing that a relation, between the tested system and the property under scrutiny, holds. We characterize testable properties w.r.t. several relations of interest. For each relation, we give a sufficient condition for a property to be testable. Then, we study and delineate a fine-grain characterization of testable properties: for each Safety-Progress class, we identify the subset of testable properties and their corresponding test oracle. Furthermore, we address automatic test generation for the proposed framework by providing a general synthesis technique that allows to obtain canonical testers for the testable properties in the Safety-Progress classification. Moreover, we show how the usual notion of quiescence can be taken into account in our general framework, and, how quiescence improves the testability results. Then, we list some existing testing approaches that could benefit from this work by addressing a wider set of properties. Finally, we propose Java-PT, a prototype Java toolbox that implements the results introduced in this article.

6.2.2. Runtime enforcement of timed properties

Participants: Thierry Jéron, Hervé Marchand, Srinivas Pinisetty.

Runtime enforcement is a powerful technique to ensure that a running system respects some desired properties. Using an enforcement monitor, an (untrusted) input execution (in the form of a sequence of events) is modified into an output sequence that complies to a property. Runtime enforcement has been extensively studied over the last decade in the context of untimed properties. The paper [19], introduces runtime enforcement of timed properties. We revisit the foundations of runtime enforcement when time between events matters. We show how runtime enforcers can be synthesized for any safety or co-safety timed property. Proposed runtime enforcers are time retardant: to produce an output sequence, additional delays are introduced between the events of the input sequence to correct it. Runtime enforcers have been prototyped and our simulation experiments validate their effectiveness.

6.2.3. Test generation for tiles systems

Participants: Sébastien Chédor, Thierry Jéron, Christophe Morvan.

In [17] we explore test generation for Recursive Tile Systems (RTS) in the framework of the classical ioco testing theory. The RTS model allows the description of reactive systems with recursion, and is very similar to other models like Pushdown Automata, Hyperedge Replacement Grammars or Recursive State Machines. We first present an off-line test generation algorithm for Weighted RTS, a determinizable sub-class of RTS, and second, an on-line test generation algorithm for the full RTS model. Both algorithms use test purposes to guide test selection through targeted behaviours.

6.2.4. Partially observed recursive tiles systems

Participants: Sébastien Chédor, Hervé Marchand, Christophe Morvan.

The analysis of discrete event systems under partial observation is an important topic, with major applications such as the detection of information flow and the diagnosis of faulty behaviors. In [18] we consider recursive tile systems, which are infinite systems generated by a finite collection of finite tiles, a simplified variant of deterministic graph grammars. Recursive tile systems are expressive enough to capture classical models of recursive systems, such as the pushdown systems and the recursive state machines. They are infinite-state in general and therefore standard powerset constructions for monitoring do not always apply. We exhibit computable conditions on recursive tile systems and present non-trivial constructions that yield effective computation of the monitors. We apply these results to the classic problems of opacity and diagnosability.

6.2.5. Off-line test selection with test purposes for non-deterministic timed automata

Participants: Nathalie Bertrand, Thierry Jéron, Amélie Stainer.

The LMCS article [7] proposes novel off-line test generation techniques from non-deterministic timed automata with inputs and outputs (TAIOs) in the formal framework of the tioco conformance theory. In this context, a first problem is the determinization of TAIOs, which is necessary to foresee next enabled actions after an observable trace, but is in general impossible because not all timed automata are determinizable. This problem is solved thanks to an approximate determinization using a game approach. The algorithm performs an io-abstraction which preserves the tioco conformance relation and thus guarantees the soundness of generated test cases. A second problem is the selection of test cases from a TAIO specification. The selection here relies on a precise description of timed behaviors to be tested which is carried out by expressive test purposes modeled by a generalization of TAIOs. Finally, an algorithm is described which generates test cases in the form of TAIOs equipped with verdicts, using a symbolic co-reachability analysis guided by the test purpose. Properties of test cases are then analyzed with respect to the precision of the approximate determinization: when determinization is exact, which is the case on known determinizable classes, in addition to soundness, properties characterizing the adequacy of test cases verdicts are also guaranteed.

6.2.6. Monitor-based statistical model checking of timed systems

Participant: Amélie Stainer.

In [16], we present a novel approach and implementation for analysing weighted timed automata (WTA) with respect to the weighted metric temporal logic (WMTL$_{\leq}$). Based on a stochastic semantics of WTAs, we apply statistical model checking (SMC) to estimate and test probabilities of satisfaction with desired levels of confidence. Our approach consists in the generation of deterministic monitors for formulas in WMTL$_{\leq}$, allowing for efficient SMC by run-time evaluation of a given formula. By necessity, the deterministic observers are in general approximate (over- or under-approximations), but are most often exact and experimentally tight. The technique is implemented in the new tool CASAAL. that we seamlessly connect to Uppaal-smc. in a tool chain. We demonstrate the applicability of our technique and the efficiency of our implementation through a number of case-studies.

6.3. Control synthesis

6.3.1. Synthesis of opaque systems

Participant: Hervé Marchand.

Opacity is a security property formalizing the absence of (secret) information leakage. We address the problem of synthesizing opaque systems. A secret predicate S over the runs of a system G is opaque to an external user having partial observability over G, if he can never infer from the observation of a run of G that the run belongs to S. We choose to control the observability of events by adding a device, called a mask, between the system G and the users. We first investigate the case of static partial observability where the set of events the user can observe is fixed once and for all by a static mask. In this context, we show that checking whether a system is opaque is PSPACE-complete, which implies that computing an optimal static mask ensuring opacity is also a PSPACE-complete problem. Next, we introduce dynamic partial observability where the set of events the user can observe changes over time and is determined by a dynamic mask. We show how to check that a system is opaque w.r.t. to a dynamic mask and also address the corresponding synthesis problem: given a system G and secret states S, compute the set of dynamic masks under which S is opaque. Our main result is that the set of such masks can be finitely represented and can be computed in EXPTIME and that this is a lower bound. We also address the problem of computing an optimal mask. This work was published in FMSD [9].

6.3.2. Symbolic Supervisory Control of Infinite Transition Systems under Partial Observation using Abstract Interpretation

Participant: Hervé Marchand.

In the DEDS article [12], we propose algorithms for the synthesis of state-feedback controllers with partial observation of infinite state discrete event systems modelled by Symbolic Transition Systems. We provide models of safe memoryless controllers both for potentially deadlocking and for deadlock free controlled systems. The termination of the algorithms solving these problems is ensured using abstract interpretation techniques which provide an overapproximation of the transitions to disable. We then extend our algorithms to controllers with memory and to online controllers. We also propose improvements in the synthesis of controllers in the finite case which, to our knowledge, provide more permissive solutions than previously proposed in the literature. Our tool SMACS gives an empirical validation of our methods by showing their feasibility, usability and efficiency.

6.3.3. Playing optimally on timed automata with random delays

Participant: Nathalie Bertrand.

In [15], we marry continuous time Markov decision processes (CTMDPs) with stochastic timed automata into a model with joint expressive power. This extension is very natural, as the two original models already share exponentially distributed sojourn times in locations. It enriches CTMDPs with timing constraints, or symmetrically, stochastic timed automata with one conscious player. Our model maintains the existence of optimal control known for CTMDPs. This also holds for a richer model with two players, which extends continuous time Markov games. But we have to sacrifice the existence of simple schedulers: polyhedral regions are insufficient to obtain optimal control even in the single-player case.

ABSTRACTION Project-Team

6. New Results

6.1. Analysis of Biological Pathways

We have improved our framework to design and analyze biological networks. This framework focused on protein-protein interaction networks described as graph rewriting systems. Such networks can be used to model some signaling pathways that control the cell cycle. The task is made difficult due to the combinatorial blow up in the number of reachable species (*i.e.*, non-isomorphic connected components of proteins).

6.1.1. Semantics

Participants: Jonathan Hayman, Tobias Heindel [CEA-List].

Domain-specific rule-based languages can be understood intuitively as transforming graph-like structures, but due to their expressivity these are difficult to model in 'traditional' graph rewriting frameworks.

In [21], we introduce pattern graphs and closed morphisms as a more abstract graph-like model and show how Kappa can be encoded in them by connecting its single-pushout semantics to that for Kappa. This level of abstraction elucidates the earlier single-pushout result for Kappa, teasing apart the proof and guiding the way to richer languages, for example the introduction of compartments within cells.

6.1.2. Semantics and causality

Participants: Vincent Danos [University of Edinburgh], Jérôme Feret, Walter Fontana [Harvard Medical School], Russ Harmer [Paris VII], Jonathan Hayman, Jean Krivine [Paris VII], Chris Thompson-Walsh [University of Cambridge], Glynn Winskel [University of Cambridge].

In [20], we introduce a novel way of constructing concise causal histories (pathways) to represent how specified structures are formed during simulation of systems represented by rulebased models. This is founded on a new, clean, graph-based semantics introduced in the first part of this paper for Kappa, a rule-based modelling language that has emerged as a natural description of protein-protein interactions in molecular biology. The semantics is capable of capturing the whole of Kappa, including subtle side-effects on deletion of structure, and its structured presentation provides the basis for the translation of techniques to other models. In particular, we give a notion of trajectory compression, which restricts a trace culminating in the production of a given structure to the actions necessary for the structure to occur. This is central to the reconstruction of biochemical pathways due to the failure of traditional techniques to provide adequately concise causal histories, and we expect it to be applicable in a range of other modelling situations.

6.1.3. Case study: Combinatorial drift in yeast model

Participants: Vincent Danos [University of Edinburgh], Eric Deeds [University of Kansas], Jérôme Feret, Walter Fontana [Harvard Medical School], Russ Harmer [Paris VII], Jean Krivine [Paris VII].

The assembly of molecular machines and transient signaling complexes does not typically occur under circumstances in which the appropriate proteins are isolated from all others present in the cell. Rather, assembly must proceed in the context of large-scale protein-protein interaction (PPI) networks that are characterized both by conflict and combinatorial complexity. Conflict refers to the fact that protein interfaces can often bind many different partners in a mutually exclusive way, while combinatorial complexity refers to the explosion in the number of distinct complexes that can be formed by a network of binding possibilities.

In [9], we use computational models so as to explore the consequences of these characteristics for the global dynamics of a PPI network based on highly curated yeast two-hybrid data. The limited molecular context represented in this data-type translates formally into an assumption of independent binding sites for each protein. The challenge of avoiding the explicit enumeration of the astronomically many possibilities for complex formation is met by a rule-based approach to kinetic modeling. Despite imposing global biophysical constraints, we find that initially identical simulations rapidly diverge in the space of molecular possibilities, eventually sampling disjoint sets of large complexes. We refer to this phenomenon as “compositional drift". Since interaction data in PPI networks lack detailed information about geometric and biological constraints, our study does not represent a quantitative description of cellular dynamics. Rather, our work brings to light a fundamental problem (the control of compositional drift) that must be solved by mechanisms of assembly in the context of large networks. In cases where drift is not (or cannot be) completely controlled by the cell, this phenomenon could constitute a novel source of phenotypic heterogeneity in cell populations.

6.1.4. Automatic Reduction of Stochastic Semantics

Participants: Ferdinanda Camporesi, Jérôme Feret, Norman Ferns, Thomas Henzinger [Institute of Science and Technology, Austria], Heinz Koeppl [ETH Zürich], Tatjana Petrov [ETH Zürich].

Biology, Protein-protein interaction networks, Stochastic semantics, Verification.

We have proposed an abstract interpretation-based framework for reducing the state-space of stochastic semantics for protein-protein interaction networks. Our framework ensures that the trace distribution of the reduced system is the exact projection of the trace distribution of the concrete system. Moreover, when the abstraction is complete, if each state with the same abstraction is equiprobable at initial state, each state with the same abstraction is equiprobable at any time t.

In [10], we have formalized the model reduction framework for the stochastic semantics and we have established the relationships with the notions of lumpability, and bisimulation.

In [13], we have showed that the reduced models can be expressed in Kappa, and we have provided a procedure to do it.

6.2. Leakage Analysis

Participants: Matteo Zanioli [Correspondent], Pietro Ferrara [ETH, Zurich], Agostino Cortesi [Università Ca’ Foscari].

Abstract interpretation, Information leakage analysis, Object-oriented software, Static analysis.

In [28], we present SAILS, a new tool that combines SAMPLE, a generic static analyzer, and a sophisticated domain for leakage analysis. This tool does not require to modify the original language, since it works with mainstream languages like JAVA™, and it does not require any manual annotation. SAILS can combine the information leakage analysis with different heap abstractions, inferring information leakage over programs with complex data structures. SAILS has been applied to the analysis of the SecuriBench-micro suite. The experimental results underline the effectiveness of the analysis, since SAILS is in position to analyze several benchmarks in about 1 second without producing false alarms in more than 90% of the programs.

6.3. Termination

Participants: Patrick Cousot, Radhia Cousot.

Abstract interpretation, Computational induction, Induction, Proof, Static analysis, Semantic structural induction, Syntactic structural induction, Termination, Variant function, Verification.

In [17], we have introduced an abstract interpretation for termination.

Proof, verification and analysis methods for termination all rely on two induction principles: (1) a variant function or induction on data ensuring progress towards the end and (2) some form of induction on the program structure.

So far, no clear design principle did exist for termination as is the case for safety so that the existing approaches are scattered and largely not comparable with each other.

- For (1), we show that this design principle applies equally well to potential and definite termination. The trace-based termination collecting semantics is given a fixpoint definition. Its abstraction yields a fixpoint definition of the best variant function. By further abstraction of this best variant function, we derive the Floyd/Turing termination proof method as well as new static analysis methods to effectively compute approximations of this best variant function.
- For (2), we introduce a generalization of the syntactic notion of structural induction (as found in Hoare logic) into a semantic structural induction based on the new semantic concept of inductive trace cover covering execution traces by segments, a new basis for formulating program properties. Its abstractions allow for generalized recursive proof, verification and static analysis methods by induction on both program structure, control, and data. Examples of particular instances include Floyd's handling of loop cut-points as well as nested loops, Burstall's intermittent assertion total correctness proof method, and Podelski-Rybalchenko transition invariants.

6.4. Probabilistic Abstract Interpretation

Participants: Patrick Cousot, Michaël Monerau.

Abstract interpretation, Probabilistic systems, Static analysis.

Abstract interpretation has been widely used for verifying properties of computer systems. In [19], we present a way to extend this framework to the case of probabilistic systems.

The probabilistic abstraction framework that we propose allows us to systematically lift any classical analysis or verification method to the probabilistic setting by separating in the program semantics the probabilistic behavior from the (non-)deterministic behavior. This separation provides new insights for designing novel probabilistic static analyses and verification methods.

We define the concrete probabilistic semantics and propose different ways to abstract them. We provide examples illustrating the expressiveness and effectiveness of our approach.

6.5. Formal Verification by Abstract Interpretation

Participant: Patrick Cousot.

Abstract interpretation, Abstraction, Aerospace, Certification, Cyber-physical system, Formal Method, Mission-critical system, Runtime error, Safety-critical system, Scalability, Soundness, Static Analysis, Validation, Verification.

Abstract interpretation is a theory of abstraction and constructive approximation of the mathematical structures used in the formal description of programming languages and the inference or verification of undecidable program properties. Developed in the late seventies with Radhia Cousot, it has since then been considerably applied to many aspects of programming, from syntax, to semantics, and proof methods where abstractions are sound and complete but incomputable to fully automatic, sound but incomplete approximate abstractions to solve undecidable problems such as static analysis of infinite state software systems, contract inference, type inference, termination inference, model-checking, abstraction refinement, program transformation (including watermarking), combination of decision procedures, security, malware detection, etc.

This last decade, abstract interpretation has been very successful in program verification for mission- and safety-critical systems [12]. An example is ASTRÉE which is a static analyzer to verify the absence of runtime errors in structured, very large C programs with complex memory usages, and involving complex boolean as well as floating-point computations (which are handled precisely and safely by taking all possible rounding errors into account), but without recursion or dynamic memory allocation. Astrée targets embedded applications as found in earth transportation, nuclear energy, medical instrumentation, aeronautics and space flight, in particular synchronous control/command such as electric flight control or more recently asynchronous systems as found in the automotive industry. Astrée is industrialized by AbsInt Angewandte Informatik GmbH.

6.6. Static Analysis of Parallel Software

Participant: Antoine Miné.

Abstract interpretation, Embedded software, Parallel software, Rely/guarantee analysis, Run-time errors, Static analysis.

We present in [11] the theoretical foundation and the latest experimental evaluation of ASTRÉEA (5.3), a static analyzer prototype based on abstract interpretation to check for run-time errors in multi-threaded embedded critical C programs. Our method is based on a slightly modified non-parallel analysis that, when analyzing a thread, applies and enriches an abstract set of thread interferences. An iterator then re-analyzes each thread in turn until interferences stabilize. We prove the soundness of our method with respect to the sequential consistency semantics, but also with respect to a reasonable weakly consistent memory semantics. We also show how to take into account mutual exclusion and thread priorities through a partitioning over an abstraction of the scheduler state. This work is an extension of [54], complete with a full formalization and soundness proofs.

In [24], we express rely/guarantee methods in constructive form as an abstract interpretation of the interleaving trace semantics. We also restate the analysis presented in [11] as a further abstraction of rely/guarantee. This theoretical work brings a new understanding of the various causes of incompleteness and imprecision in our previous analysis, including the non-relational, input-insensitive, flow-insensitive, and history-insensitive treatment of interferences, and it opens the way to designing more precise analyses.

6.7. Static Analysis of Bit-Level Machine Integer and Floating-Point Operations

Participant: Antoine Miné.

Abstract interpretation, Embedded software, Numerical abstract domains, Run-time errors, Static analysis.

We present in [22] a few lightweight numeric abstract domains to analyze C programs that exploit the binary representation of numbers in computers, for instance to perform "compute-through-overflow" on machine integers, or to directly manipulate the exponent and mantissa of floating-point numbers. On integers, we propose an extension of intervals with a modular component, as well as a bitfield domain. On floating-point numbers, we propose a predicate domain to match, infer, and propagate selected expression patterns. These domains are simple, efficient, and extensible. We have included them into the ASTRÉE (5.2) and ASTRÉEA (5.3) static analyzers to supplement existing domains. Experimental results show that they can improve the analysis precision at a reasonable cost.

6.8. Inferring Sufficient Conditions with Backward Polyhedral Under-Approximations

Participant: Antoine Miné.

Abstract interpretation, Backward analysis, Numerical abstract domains, Static analysis, Sufficient condition inference, Under-approximations.

In [23], we discuss the automatic inference of sufficient pre-conditions by abstract interpretation and sketch the construction of an under-approximating backward analysis. We focus on numeric domains and propose transfer functions, including a lower widening, for polyhedra, without resorting to disjunctive completion nor complementation, while soundly handling non-determinism. A limited proof-of-concept prototype was designed to validate our approach. Planned applications include the derivation of sufficient conditions for a program to never step outside an envelope of safe states, or dually to force it to eventually fail.

6.9. A Constraint Solver Based on Abstract Domains

Participants: Marie Pelleau [University of Nantes, LINA], Antoine Miné, Charlotte Truchet [University of Nantes, LINA], Frédéric Benhamou [University of Nantes, LINA].

Abstract interpretation, Backward analysis, Numerical abstract domains, Static analysis, Sufficient condition inference, Under-approximations.

In [25], we apply techniques from abstract interpretation to constraint programming (which aims at solving hard combinatorial problems with a generic framework based on first-order logics). We highlight some links and differences between these fields: both compute fixpoints by iterations but employ different extrapolation and refinement strategies; moreover, consistencies in Constraint Programming can be mapped to non-relational abstract domains. We then use these correspondences to build an abstract constraint solver that leverages abstract interpretation techniques (such as relational domains) to go beyond classic solvers. We present encouraging experimental results obtained with our prototype implementation.

6.10. Automatic Inference of Necessary Preconditions

Participants: Patrick Cousot, Radhia Cousot, Manuel Fahndrich [Microsoft Research, Redmond, USA], Francesco Logozzo [Microsoft Research, Redmond, USA].

Abstract interpretation, Backward analysis, Static analysis, Necessary condition inference,

In [18], we consider the problem of automatic precondition inference for: (i) program verification; (ii) helping the annotation process of legacy code; and (iii) helping generating code contracts during code refactoring. We argue that the common notion of sufficient precondition inference (i.e., under which precondition is the program correct?) imposes too large a burden on call-sites, and hence is unfit for automatic program analysis. Therefore, we define the problem of necessary precondition inference (i.e., under which precondition, if violated, will the program always be incorrect?). We designed and implemented several new abstract interpretation-based analyses to infer necessary preconditions. The analyses infer atomic preconditions (including disjunctions), as well as universally and existentially quantified preconditions.

We experimentally validated the analyses on large scale industrial code.

For unannotated code, the inference algorithms find necessary preconditions for almost 64% of methods which contained warnings. In 27% of these cases the inferred preconditions were also sufficient, meaning all warnings within the method body disappeared. For annotated code, the inference algorithms find necessary preconditions for over 68% of methods with warnings. In almost 50% of these cases the preconditions were also sufficient. Overall, the precision improvement obtained by precondition inference (counted as the additional number of methods with no warnings) ranged between 9% and 21%.

6.11. Inference of Necessary Field Conditions with Abstract Interpretation

Participants: Mehdi Bouaziz, Manuel Fahndrich [Microsoft Research, Redmond, USA], Francesco Logozzo [Microsoft Research, Redmond, USA].

In [15], we present a new static analysis to infer necessary field conditions for object-oriented programs. A necessary field condition is a property that should hold on the fields of a given object, for otherwise there exists a calling context leading to a failure due to bad object state. Our analysis also infers the provenance of the necessary condition, so that if a necessary field condition is violated then an explanation containing the sequence of method calls leading to a failing assertion can be produced.

When the analysis is restricted to readonly fields, i.e., fields that can only be set in the initialization phase of an object, it infers object invariants. We provide empirical evidence on the usefulness of necessary field conditions by integrating the analysis into cccheck, our static analyzer for .NET.

Robust inference of readonly object field invariants was the #1 request from cccheck users.

6.12. TreeKs: A Functor to Make Numerical Abstract Domains Scalable

Participant: Mehdi Bouaziz.

Relational numerical abstract domains do not scale up. To ensure a linear cost of abstract domains, abstract interpretation-based tools analyzing large programs generally split the set of variables into independent smaller sets, sometimes sharing some non-relational information. In [14], we present a way to gain precision by keeping fully expressive relations between the subsets of variables, whilst retaining a linear complexity ensuring scalability.

6.13. An Abstract Domain to Infer Types over Zones in Spreadsheets

Participants: Cheng Tie, Xavier Rival.

abstract domains, spreadsheet script languages In [16], we proposed an abstract domain for the abstraction of spreadsheet contents.

Spreadsheet languages are very commonly used, by large user bases, yet they are error prone. However, many semantic issues and errors could be avoided by enforcing a stricter type discipline. As declaring and specifying type information would represent a prohibitive amount of work for users, we propose an abstract interpretation based static analysis for spreadsheet programs that infers type constraints over zones of spreadsheets, viewed as two-dimensional arrays. Our abstract domain consists in a cardinal power from a numerical abstraction describing zones in a spreadsheet to an abstraction of cell values, including type properties. We formalize this abstract domain and its operators (transfer functions, join, widening and reduction) as well as a static analysis for a simplified spreadsheet language. Last, we propose a representation for abstract values and present an implementation of our analysis.

6.14. Hierarchical Abstraction of Dynamic Structures

Participants: Pascal Sotin, Xavier Rival.

abstract domains, shape analysis, domain combination In [26], we designed a hierarchical shape abstract domain for the abstraction of complex data structures found in embedded softwares.

We propose a hierarchical shape abstract domain, so as to infer structural invariants of dynamic structures such as lists living inside static structures, such as arrays. This programming pattern is often used in safety critical embedded software that need to “allocate” dynamic structures inside static regions due to dynamic memory allocation being forbidden in this context. Our abstract domain precisely describes such hierarchies of structures. It combines several instances of simple shape abstract domains, dedicated to the representation of elementary shape properties, and also embeds a numerical abstract domain. This modular construction greatly simplifies the design and the implementation of the abstract domain. We provide an implementation, and show the effectiveness of our approach on a problem taken from a real code.

6.15. Reduced Product Combination of Abstract Domains for Shapes

Participants: Antoine Toubhans, Xavier Rival, Bor-Yuh Evan Chang [University of Colorado at Boulder].

abstract domains, shape analysis, reduced product In [27], we proposed a notion of reduced product for shape abstractions.

Real-world data structures are often enhanced with additional pointers capturing alternative paths through a basic inductive skeleton (e.g., back pointers, head pointers). From the static analysis point of view, we must obtain several interlocking shape invariants. At the same time, it is well understood in abstract interpretation design that supporting a separation of concerns is critically important to designing powerful static analyses. Such a separation of concerns is often obtained via a reduced product on a case-by-case basis. In this paper, we lift this idea to abstract domains for shape analyses, introducing a domain combination operator for memory abstractions. As an example, we present simultaneous separating shape graphs, a product construction that combines instances of separation logic-based shape domains. The key enabler for this construction is a static analysis on inductive data structure definitions to derive relations between the skeleton and the alternative paths. From the engineering standpoint, this construction allows each component to reason independently about different aspects of the data structure invariant and then separately exchange information via a reduction operator. From the usability standpoint, we enable describing a data structure invariant in terms of several inductive definitions that hold simultaneously.

ATEAMS Project-Team

5. New Results

5.1. Programming language support for statically type access to external resources

One of the open issues in programming is how to obtain typed access, including its beneficial IDE support, to data sources that have not been modeling with the programming language's data modeling facilities. Rather most data is modeled externally or not modeled at all. Mark Hills, Jurgen Vinju and Paul Klint proposed, designed and validated a programming language design where meta models for external data are imported and/or inferred at compile-time. These models are then used to generate source code to represent these models natively in the idiom of the programming language.

5.2. Statically analyzing PHP code

Tool support in IDEs for PHP code is limited due to the dynamic nature of the language. Mark Hills, Jurgen Vinju and Paul Klint produced a principled yet pragmatic infrastructure for analyzing PHP code nevertheless. The analyses first use crude but effective over-approximations of the PHP semantics to limit the search spaces and improve accuracy, then exploit information from user-manuals, and then use state-of-the-art static analysis techniques in a fixed point abstract interpretation to arrive at accurate results.

5.3. Modular Language Parametric Refactoring Framework

Anastasia Izmaylova with Jurgen Vinju produced a prototype implementation of a framework for specifying refactoring tools based on type constraints that is open to unpredicted language extensions. The problem with the co-evolution of programming language and their supporting refactoring tools is complexity. Often existing refactorings are not retro-fitted with the new language semantics and new opportunities for refactoring tools are not fulfilled. Anastasia designed a solution based on monad transformers that allows the kind of invasive extensibility needed to adapt complex existing implementation of language semantics with additional features that interact in many ways with the existing features.

5.4. Communication Action Emulation

CAE is a novel epistemic model for describing and evaluating the equivalence of communication models by Floor Sietsma and Jan van Eijck.

5.5. Notation-Parametric Grammar Recovery

Vadim Zaytsev generalized the algorithm for recovering context-free grammars from legacy language documentation. This facilitated the recovery of more grammars to be used in the study of grammarware and software language engineering.

5.6. (In)Validating Domain Knowledge Existence in Legacy Source Code

Davy Landman conducted a large experiment in comparing an extensive domain model to the information present in source code of applications that are used in the domain in question. The project management domain was chosen for this. The experiment is still in progress. Big steps were made in setting up the experiment, which includes reporting comprehensively on a large number of design decisions, in a traceable and reproducible manner.

5.7. Ensō

Ensōis a new programming system based on interpretation of domain-specific modeling languages. The system is co-designed and authors by Tijs van der Storm in collaboration with William Cook and Alex Loh. The two foundations of the Ensōsystem are managed data and object grammars. Managed data provides modular strategies for customizing how programming languages represent and provide access to data.

Object grammars form the second foundation: they facilate declarative, compositional, and bidirectional mappings from textual syntax to object graphs. Domain-specific models in Ensōare parsed and rendered using object grammars, and represented, in memory as managed data. Together they combine into a highly flexible and modular platform for model-driven development.

CARTE Project-Team

6. New Results

6.1. Dynamical systems

Participant: Mathieu Hoyrup.

Birkhoff theorem is a central result in ergodic theory. Consider a dynamical system $(X, T : X \to X)$, start with an initial condition $x \in X$ and construct the trajectory $(x, T(x), T^2(x), ...)$. How is this trajectory distributed in X? What is the limit frequency of visits of a set $A \subseteq X$? Ergodic theorems answer to these questions by showing (i) that the distribution of *almost every* point converges and (ii) by describing the possible distributions associated to trajectories.

For several years we have been working on the project of identifying the exact computational content of several ergodic theorems: can the speed of convergence of limit frequencies be computed? Can one distinguish between points with different limit frequencies? Can we construct (compute) points whose trajectory follow a prescribed distribution? How random (i.e. incompressible) a point has to be for the distribution of its trajectory to converge?

6.1.1. Limit frequencies

We have obtained new insight in the above questions by proving that random elements eventually reach effective closed sets of positive measure (while it was only known for a more restricted class of sets). The paper appeared in *Information and Computation* [11]. This result is a key tool in the proof of the result published in [23].

6.1.2. Information

A chaotic system is unpredictable because it has much more trajectories than observable initial conditions: hence many undistinguishable initial points lead to radically different trajectories. As there are many trajectories, most of them are complex in the sense that they can hardly be compressed, i.e. described in a shorter way than simply listing them. The Shannon-McMilan-Breiman theorem states that the compression-rate of most trajectories coincides with the entropy of the system.

We have been interested in the computational content of this theorem: how random a point has to be to generate a trajectory whose compression rate is the entropy? This question was raised in [71] and has been left open for 14 years. We have solved the problem by showing that Martin-Löf notion of randomness is sufficient. Our recent result presented in [11]is a key ingredient of our proof. We presented the result at *STACS* [23].

6.1.3. Decomposition

The ergodic decomposition theorem states that a dynamical system can always be uniquely decomposed into indecomposable subsystems, technically *ergodic* subsystems. We have been interested in the computability of the decomposition operation. It is known from [71] that this operation is not computable in general. Whether this operation is still not computable when the system can be decomposed into a *finite* number of subsystems was open. We raised the question and answer it negatively in [57]. More precisely, we prove the existence of ergodic measures P and Q such that neither P nor Q is computable relative to $P + Q$. In other words, the operation of splitting a non-ergodic process into ergodic components is not computable, even in the trivial case of a combination of 2 ergodic processes. The paper is currently in press and will appear in *Annals of Pure and Applied Logic* [14].

6.2. Computations

Participant: Mathieu Hoyrup.

6.2.1. Inversion of computable functions

We strengthen the preceding result by making P and Q computable. This result is a particular case of a more general problem. In many situations an operator $F \to Y$ can be computed but can hardly be reversed: given $F(x)$, x cannot always we recovered (computed) even when F is one-to-one. We introduce a strong notion of discontinuity for the inverse of F and prove that it entails the existence of a non-computable $x \in X$ such that $F(x)$ is computable. Our result on the ergodic decomposition can be derived by applying our general result to the operator $F(P, Q) = P + Q$ which is computable but difficult to reverse. At the same time we prove a significant improvement of a classical result of Pour-El and Richards [67] about the computability of linear operators. The paper [26] is currently submitted.

6.2.2. Computability and measure theory.

We study the constructive content of the Radon-Nikodym theorem, show that it is not computable in general and precisely locate its non-computability in the Weihrauch lattice. The paper [15] appeared in the first issue of the new journal *Computability*.

6.3. Computer virology

6.3.1. Behavioral analysis

Participants: Isabelle Gnaedig, Jean-Yves Marion.

Our study on behavioural malware detection has been continued. We have been developing an approach detecting suspicious schemes on an abstract representation of the behavior of a program, by abstracting program traces, rewriting given subtraces into abstract symbols representing their functionality. Considering abstract behaviors allows us to be implementation-independent and robust to variants and mutations of malware. Suspicious behaviors are then detected by comparing trace abstractions to reference malicious behaviors.

We had previously proposed to abstract trace automata by rewriting them with respect to a set of predefined behavior patterns defined as a regular language described by a string rewriting system [32]. We then have increased the power of our approach on two aspects. We fist have modified the abstraction mechanism, keeping the abstracted patterns in the rewritten traces, which allows us to handle interleaved patterns. Second, we have extended the rewriting framework to express data constraints on action parameters by using term rewriting systems. An important consequence is that, unlike in [32], using the data-flow, we can now detect information leaks in order to prevent unauthorized disclosure or modifications of information.

We also have introduced model checking in our approach: the predefined behavior patterns, used to abstract program traces, have been defined by first order temporal logic formulas, as well as the reference suspicious behaviors, given in a signature. The infection problem can then be seen as the satisfaction problem of the formula of the signature by an abstracted trace of the program, which can be checked using existing model checking techniques. This work has been published at the ESORICS conference [20].

6.3.2. Analyzing cryptographic implementations

Participants: Joan Calvet, Jean-Yves Marion.

Analyzing cryptographic implementations has important applications, especially for malware analysis where they are an integral part both of the malware payload and the unpacking code that decrypts this payload. These implementations are often based on well-known cryptographic functions, whose description is publicly available. While potentially very useful for malware analysis, the identification of such cryptographic primitives is made difficult by the fact that they are usually obfuscated. Current state-of-the-art identification tools are ineffective due to the absence of easily identifiable static features in obfuscated code. However, these implementations still maintain the input-output (I/O) relationship of the original function. In a joint work with José M. Fernandez published in [22], we present a tool that leverages this fact to identify cryptographic functions in obfuscated programs, by retrieving their I/O parameters in an implementation-independent fashion, and comparing them with those of known cryptographic functions. In experimental evaluation, we successfully

identified the cryptographic functions TEA, RC4, AES and MD5 in obfuscated programs. In addition, our tool was able to recognize basic operations done in asymmetric ciphers such as RSA.

6.3.3. Self-replication

Participant: Jean-Yves Marion.

Self-replication is one of the fundamental aspects of computing where a program or a system may duplicate, evolve and mutate. Our point of view is that Kleene's (second) recursion theorem is essential to understand self-replication mechanisms. An interesting example of self-replication codes is given by computer viruses. This was initially explained in the seminal works of Cohen and of Adleman in the eighties. In fact, the different variants of recursion theorems provide and explain constructions of self-replicating codes and, as a result, of various classes of malware. None of the results are new from the point of view of computability theory. We just propose a self-modifying register machine as a model of computation in which we can effectively deal with self-reproduction and in which new offsprings can be activated as independent organisms. This work was published by Jean-Yves Marion in a special issue on the honor of Alan Turing [16].

6.3.4. Reverse engineering by morphological analysis

Participants: Guillaume Bonfante, Jean-Yves Marion, Fabrice Sabatier, Aurélien Thierry.

Let us suppose we are given some malware and we want to know what it is doing. One may run it, or one may analyze it more or less statically. Typically, an expert tries to guess the behavior of a malware through the analysis of its binary code (in tools such as Ida). The task is much simpler if the expert already knows some part of the code. We have shown that morphological analysis could be used in such a context. We have rediscovered the parts of the malware `Duqu` within `Stuxnet`. We have rediscovered the compilation options used to include OpenSSL's functions within `Waledac` [21].

CASSIS Project-Team

6. New Results

6.1. Automated Deduction

We develop general techniques which allow us to re-use available tools in order to build a new generation of 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 decision procedures for a wide range of verification problems.

6.1.1. Building and verifying decision procedures

Participants: Alain Giorgetti, Olga Kouchnarenko, Christophe Ringeissen, Elena Tushkanova.

We have developed a methodology to build decision procedures by using superposition calculi which are at the core of equational theorem provers. We are interested in developing automated deduction techniques to prove properties about these superposition-based decision procedures. To this aim, we plan to further investigate the use of schematic superposition, which has been already applied to check the termination and the combinability of superposition-based procedures. We have been working on the development of a framework for specifying and verifying superposition-based procedures. In [52], we present an implementation in Maude of the two inference systems corresponding to superposition and schematic superposition. Thanks to this implementation we automatically derive termination of superposition for a couple of theories of interest in verification.

Until now, schematic superposition was only studied for standard superposition. In [62], we introduce a schematic superposition calculus modulo a fragment of arithmetics, namely the theory of Integer Offsets. This new schematic calculus is used to prove the decidability of the satisfiability problem for some theories extending Integer Offsets. We illustrate our theoretical contribution on theories representing extensions of classical data structures, e.g., lists and records. Our Maude-based implementation has been extended to incorporate this new schematic superposition calculus modulo Integer Offsets. It enables automatic decidability proofs for theories of practical use.

6.2. Security Protocol Verification

The design of cryptographic protocols is error-prone. Without a careful analysis, subtle flaws may be discovered several years after the publication of a protocol, yielding potential harmful attacks. In this context, formal methods have proved their interest for obtaining good security guarantees. Many analysis techniques have been proposed in the literature [72]. We have edited a book [65] where each chapter presents an important and now standard analysis technique. We develop new techniques for richer primitives, wider classes of protocols and higher security guarantees.

6.2.1. Equational theories of cryptographic primitives

Participant: Michaël Rusinowitch.

Some attacks exploit in a clever way the interaction between protocol rules and algebraic properties of cryptographic operators. In [76], we provide a list of such properties and attacks as well as existing formal approaches for analyzing cryptographic protocols under algebraic properties.

Encryption "distributing over pairs" is employed in several cryptographic protocols. We have shown that unification is decidable for an equational theory HE specifying such an encryption [15]. We model block chaining in terms of a simple, convergent, rewrite system over a signature with two disjoint sorts: list and element and present in [27] an algorithm for deciding the unification problem modulo this rewrite system. Potential applications of this unification procedure include flaw detection for protocols employing the CBC encryption mode. We have also studied a very simple property satisfied by the RSA-based implementation of the *blind signature scheme* and we have shown its unification problem is undecidable [28]. It is the simplest theory, to our knowledge, for which unification is undecidable.

In their seminal work Dolev and Yao used string rewriting to check protocol security against an active intruder. The main technical result and algorithm were improved by Book and Otto who formulated the security check in terms of an extended word problem for cancellation rules. We extend in [16] their main decidability result to a larger class of string rewrite systems called opt-monadic systems.

6.2.2. Voting protocols

Participants: Mathilde Arnaud, Véronique Cortier, David Galindo-Chacon, Stéphane Glondu, Malika Izabachene, Steve Kremer, Cyrille Wiedling.

Voting is a cornerstone of democracy and many voting systems have been proposed so far, from old paper ballot systems to purely electronic voting schemes. Although many works have been dedicated to standard protocols, very few address the challenging class of voting protocols. We have studied several protocols that are currently in use:

- Helios is an open-source web-based end-to-end verifiable electronic voting system, used e.g. by UCL and the IACR association in real elections. We have discovered a vulnerability which allows an adversary to compromise the privacy of voters and we have presented a fixed version, showed to satisfy a formal definition of ballot secrecy using the applied pi calculus [21]. One main advantage of Helios is its verifiability, up-to the ballot box (a dishonest ballot box may add ballots). We are now working on defining a variant of Helios that prevents from ballot stuffing, even against a dishonest ballot box. Our approach consists in introducing an additional authorities that provides credentials that the ballot box can verify but not forge. This new version is under implementation and we are proving computational security for both ballot secrecy (inherited from Helios) and full verifiability (due to our credentials).
- Norway has used e-voting in its last political election in September 2011, with more than 25 000 voters using the e-voting option. Using formal models, we have analyzed the underlying protocol w.r.t. privacy, considering several corruption scenarios [41].
- The Section 07 of CNRS (now split into Section 06 and Section 07) has proposed a voting protocol for Face-to-Face meetings to enhanced the verifiability of an election run through electronic devices. We have formally modeled this protocol and proved both ballot secrecy and verifiability.

Even a basic property like ballot secrecy is difficult to define formally and several definitions co-exist. The loss of privacy may not only come from the protocol but also from the tally function itself and depends on what needs to be kept private. We have proposed a general and quantitative definition of privacy, that captures two previously proposed definitions [35]. Security based on cryptography relies on the fact that certain operations (such as decrypting) are computationally infeasible. However, e-voting protocols should also guarantee privacy in the future, when computers will have an increased computational power and will be able e.g. to break nowadays keys. Such privacy in the future is called *everlasting privacy* and we have proposed a definition of *practical everlasting privacy*.

6.2.3. Other families of protocols

Participants: Véronique Cortier, Steve Kremer, Robert Künnemann, Cyrille Wiedling.

Securing routing Protocols. The goal of routing protocols is to construct valid routes between distant nodes in the network. If no security is used, it is possible for an attacker to disorganize the network by maliciously interacting with the routing protocols, yielding invalid routes to be built. That is why secure versions of routing protocols are now developed. The security model differs from standard protocols since the adversary can only control some nodes of the network. The security of a routing protocols therefore depends on the network topology. In [39], we show a simple reduction result: if there is an attack then there is an attack in a four nodes topology. It is therefore sufficient to study security for a finite number of distinct topologies, allowing to reuse existing tools such as ProVerif.

Security APIs. In some systems, it is not possible to trust the host machine on which sensitive codes are executed. In that case, security-critical fragments of a program should be executed on some tamper resistant device (TRD), such as a smartcard, USB security token or hardware security module (HSM). The exchanges between the trusted and the untrusted infrastructures are ensured by special kind of API (Application Programming Interface), that are called *security APIs*. We have previously designed a generic API for key-management based on key hierarchy [77]. In [40], [60], we have extended our API to handle key-revocation such that the security tokens can still be used (it is not necessary to revoke the full token) and such that any key can be revoked (even upper keys in the hierarchy). In [64], we propose a universally composable key management functionality and show how to achieve a secure, distributed implementation on TRDs.

6.2.4. Automated verification of indistinguishability properties.

Participants: Rémy Chrétien, Véronique Cortier, Steve Kremer.

New emerging classes of protocols such as voting protocols often require to model less classical security properties, such as anonymity properties, strong versions of confidentiality and resistance to offline guessing attacks. Many of these properties can be modelled using the notion of indistinguishability by an adversary, which can be conveniently modeled using process equivalences.

Static case. The YAPA tool [17] can check static equivalence for convergent equational theories. It is proved to terminate for a wide class of equational theories that includes subterm convergent theories (e.g. encryption, signatures, pairing and hash) and layered convergent theories (e.g. blind signatures). The procedure is generic in the sense that it remains sound and complete (but may not terminate) for any convergent theory. It has been implemented in the YAPA tool[3]. The KISS tool [19] is also able to verify static equivalence for convergent equational theories. Termination has been shown for subterm convergent equational theories (a subset of layered convergent theories) as well as several equational theories motivated by electronic voting protocols such as blind signatures and trap-door commitment schemes (which are out of the scope of YAPA).

In [20], we show how to *combine* decision procedures: if static equivalence and deduction are decidable for two disjoint equational theories then they are decidable for the union of the theories. In [25] we develop a method that allows us in some cases to simplify the task of deciding static equivalence in a multi-sorted setting, by removing a symbol from the term signature and reducing the problem to several simpler equational theories. We illustrate our technique at hand of bilinear pairings.

Active case. In [36] we present a novel procedure to verify equivalence properties for a bounded number of sessions which is able to handle a large class of equational theories. Although, we were unable to prove termination of the resolution procedure, the procedure has been implemented in a prototype tool and has been effectively tested on examples. We were able to verify properties such as guessing attacks in password protocols, strong flavors of confidentiality and anonymity properties, including fully automated checking of anonymity of an electronic voting protocol by Fujioka et al. which was outside the scope of existing tools.

In [42] we study this equivalence problem when cryptographic primitives are modeled using a group equational theory, a special case of monoidal equational theories. We reduce the problem to solving systems of equations over rings and provide several new decidability and complexity results, notably for equational theories which have applications in security protocols, such as exclusive or and Abelian groups which may additionally admit a unary, homomorphic symbol.

Rémy Chrétien has recently started a PhD on deciding trace equivalence for an unbounded number of sessions. His first findings show that for some classes of protocols, decidability of trace equivalence can be reduced to equivalence of deterministic pushdown automata (which is decidable [81]).

Note that for simple processes without branch nor replication observational equivalence can be reduced to checking whether two symbolic constraints (representing honest agents) are equivalent [75]. We have published a new proof that symbolic constraints equivalence is decidable for the large class of subterm convergent theories [18].

[3] http://www.lsv.ens-cachan.fr/~baudet/yapa/

6.2.5. Soundness of the Dolev-Yao Model

Participants: Véronique Cortier, Guillaume Scerri.

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. A recent line of research consists in identifying cases where it is possible to obtain the best of both cryptographic and formal worlds: fully automated proofs and strong, clear security guarantees.

Existing soundness results for symmetric encryption are not satisfactory. This is due to the fact that dishonest keys may introduce many behaviors that cannot be easily captured in symbolic models. Guillaume Scerri has started a PhD thesis on designing more flexible symbolic models for cryptographic proofs. His first result is a computationally sound symbolic model in the presence of dishonestly generated keys, allowing a symbolic adversary to generate new equalities between terms, on-the-fly [38].

6.3. Model-based Verification

We have investigated extensions of regular model-checking to new classes of rewrite relations on trees. We have studied specification and proof of modular imperative programs.

6.3.1. Algorithms for Tree Walking Automata

Participants: Pierre-Cyrille Héam, Vincent Hugot, Olga Kouchnarenko.

Tree walking automata are widely used to tackle data base algorithmic problems, particularly to analyse queries over XML documents. The emptiness problem for tree walking automata is known to be EXPTIME-complete. The general algorithm to solve this problem consists in transforming the tree walking automaton into a classical top-down tree automaton. The best known in the literature algorithm works in time $O(s2^{n^2})$ where n is the number of states of the tree walking automaton, and s is the size of the alphabet. In [24] we have proposed a new algorithm based on an *overloop* concept and working in time $O(2^{n^2})$. Then our approach has been improved for deterministic tree walking automata to have in this case a $O(2^{n \log n})$ time complexity. Finally, we have also proposed a polynomial-time approximation based semi-algorithm for the emptiness problem. The algorithms have been implemented and experimental results confirm the relevance of the approach.

6.3.2. Algorithms for Tree Automata with Global Constraints

Participants: Pierre-Cyrille Héam, Vincent Hugot, Olga Kouchnarenko.

Extending tree automata models to be able to compare different tree branches is an important and challening issue for systems' modeling and for verifying their properties. Several exetensions have been proposed in the litterature. Among them we are interested in the model of Tree Automata with Global Constraints (TAGED) introduced in 2009. The membership problem for this new model is known to be NP-complete, and the emptyness problem is known to be EXPTIME-complete. In [47] we have investigated some complexity results for tree automata with a bounded number of equality constraints. We have proved that with a unique constraint the emptyness problem is in PTIME and that it is EXPTIME-complete with only two constraints. For a bounded number of constraints, the membership problem is in PTIME.

6.3.3. Verification of Linear Temporal Patterns over Finite and Infinite Traces

Participants: Pierre-Cyrille Héam, Vincent Hugot, Olga Kouchnarenko.

In the regular model-checking framework, reachability analysis can be guided by temporal logic properties, for instance to achieve the counter example guided abstraction refinement (CEGAR) objectives. A way to perform this analysis is to translate a temporal logic formula expressed on maximal rewriting words into a "rewrite proposition" – a propositional formula whose atoms are language comparisons, and then to generate semi-decision procedures based on (approximations of) the rewrite proposition. In [46] we have investigated suitable semantics for LTL on maximal rewriting words and their influence on the feasibility of a translation, and we have proposed a general scheme providing exact results for a fragment of LTL corresponding mainly to safety formulæ, and approximations for a larger fragment.

6.3.4. Rewriting-based Mathematical Model Transformations

Participants: Walid Belkhir, Alain Giorgetti.

We have pursued our collaboration with the Department "Temps-Fréquence" of the FEMTO-ST institute (Franche-Comté Electronique Mécanique Thermique et Optique - Sciences et Technologies, CNRS UMR 6174) on the formalization of asymptotic methods (based on two-scale convergence) to automatically generate asymptotic models of large arrays of micro- and nanosystems. The goal is to provide engineers with an implementation of this mathematical tool inside a modeling software. We follow therefore a multidisciplinary approach which combines a generalization and formalization effort of mathematical asymptotic methods, together with rewriting-based formal transformation techniques. This approach is described in [53], together with an example and a presentation of the architecture of the software under design. A second contribution [34] is a detailed formal specification and analysis of lazy pattern-matching mechanism modulo associativity and commutativity, and its integration into a strategy language. The pattern-matching solutions are stored in a lazy list composed of a first substitution at the head and a non-evaluated object that encodes the remaining computations. Rule and strategy applications also produce a lazy list of terms. This contribution has been published in EPTCS as the proceedings of the 10th International Workshop on Reduction Strategies in Rewriting and Programming, where a lighter version was presented in 2011 [69].

6.4. Model-based Testing

Our research in Model-Based Testing (MBT) aims to extend the coverage of tests. The coverage refers to several artefacts: model, test scenario/property, and code of the program under test. The test generation uses various underlying techniques such as symbolic animation of models [22] or symbolic execution of programs by means of dedicated constraints, SMT solvers, or model-checkers.

6.4.1. Automated Test Generation from Behavioral Models

Participants: Fabrice Bouquet, Kalou Cabrera, Jérome Cantenot, Frédéric Dadeau, Elizabeta Fourneret, Jean-Marie Gauthier, Jonathan Lasalle.

We have introduced an original model-based testing approach that takes a behavioural view (modelled in UML) of the system under testing and automatically generates test cases and executable test scripts according to model coverage criteria. We continue to extended this result to SysML specifications for validating embedded systems [26]. To allow the test generation from SysML model, we study the transformation into a low level language more close of hardware in [44].

In the context of software evolution, we have worked on exploiting the evolution of requirements in order to classify test sequences, and precisely target the parts of the system impacted by this evolution. We have proposed to define the life cycle of a test via three test classes: (i) Regression, used to validate that unimpacted parts of the system did not change, (ii) Evolution, used to validate that impacted parts of the system correctly evolved, and (iii) Stagnation, used to validate that impacted parts of the system did actually evolve. The associated algorithms are under implementation in a dedicated prototype to be used in the SecureChange european project. A link with the security model proof has been started with partners of the project in [54] that allows to generate test needs associated to security properties verified on model.

6.4.2. Scenario-Based Verification and Validation

Participants: Fabrice Bouquet, Kalou Cabrera, Frédéric Dadeau, Elizabeta Fourneret.

Test scenarios represent an abstract test case specification that aims at guiding the model animation in order to produce relevant test cases. Contrary to the previous section, this technique is not fully automated since it requires the user to design the scenario, in addition to the model.

We have designed a scenario based testing language for UML/OCL that can be either connected to a model animation engine or to a symbolic animation engine, based on a set-theoretical constraint solver [22]. In the context of the ANR TASCCC project, we are investigating the automation of test generation from Security Functional Requirements (SFR), as defined in the Common Criteria terminology. SFRs represent security functions that have to be assessed during the validation phase of security products (in the project, the Global Platform, an operating system for latest-generation smart cards). To achieve that, we are working on the definition of description patterns for security properties, to which a given set of SFRs can be related. These properties are used to automatically generate test scenarios that produce model based test cases. The traceability, ensured all along the testing process, makes it possible to provide evidences of the coverage of the SFR by the tests, required by the Common Criteria to reach the highest Evaluation Assurance Levels.

We have proposed a dedicated formalism to express test properties. A test property is first translated into a finite state automaton which describes a monitor of its behaviors. We have proposed dedicated property coverage criteria that can be used either to measure the property coverage of a given test suite, or to generate test cases, exercising nominal or robustness aspects of the property.

In the context of the SecureChange project, we also investigate the evolution of test scenarios. As the system evolves, the model evolves, and the associated test scenarios may also evolve. We are currently extending the test generation and management of system evolutions to ensure the preservation of the security.

6.4.3. *Mutation-based Testing of Security Protocols*

Participants: Frédéric Dadeau, Pierre-Cyrille Héam.

Verification of security protocols models is an important issue. Nevertheless, the verification reasons on a model of the protocol, and does not consider its concrete implementation. While representing a safe model, the protocol may be incorrectly implemented, leading to security flaws when it is deployed. We have proposed a model-based penetration testing approach for security protocols [9]. This technique relies on the use of mutations of an original protocol, proved to be correct, for injecting realistic errors that may occur during the protocol implementation (e.g. re-use of existing keys, partial checking of received messages, incorrect formatting of sent messages, use of exponential/xor encryption, etc.). Mutations that lead to security flaws are used to build test cases, which are defined as a sequence of messages representing the behavior of the intruder. We have applied our technique on protocols designed in HLPSL, and implemented a protocol mutation tool that performs the mutations. The mutants are then analyzed by the CL-Atse [82] front-end of the AVISPA toolset [66]. Experiments show the relevance of the proposed mutation operators and the efficiency of the CL-Atse tool to conclude on the vulnerability of a protocol and produce an attack trace that can be used as a test case for implementations.

6.4.4. *Code-related Test Generation and Static Analysis*

Participants: Fabrice Bouquet, Frédéric Dadeau, Ivan Enderlin, Alain Giorgetti.

In collaboration with the CEA we enhance the innovative verification technique SANTE (Static ANalysis and TEsting), combining value analysis, program slicing and test generation, with two novel, optimized and adaptive strategies of program slicing based on threat dependencies [37]. We study the properties of threat dependencies, introduce the notion of slicing-induced cover, and prove the underlying theoretical results. Compared to a basic usage of program slicing, our advanced strategies need only quadratic additional work in order to optimize the calls of costly dynamic analysis. We give a detailed evaluation of all slicing strategies and compare them with one another.

We have designed a new annotation language for PHP, named PRASPEL for PHP Realistic Annotation SPEcification Language. This language relies on *realistic domains* which serve two purposes. First, they assign to a data a domain that is supposed to be specific w.r.t. a context in which it is employed. Second, they provide two features that are used for test generation: (i) *samplability* makes it possible to automatically generate a value that belongs to the realistic domain so as to generate test data, (ii) *predicability* makes it possible to check if the value belongs to a realistic domain. This approach is tool-supported in a dedicated framework for PHP which makes it possible to produce unit test cases using random data generators, execute the test cases on an instrumented implementation, and decide the conformance of the code w.r.t. the annotations by runtime assertion checking. This principle has been extended to generate grammar-based textual data [43] based on various strategies, namely uniform random generation, bounded exhaustive generation and rule-coverage-based test generation.

6.4.5. Specification, implementation and validation of generation algorithms

Participant: Alain Giorgetti.

We have shown how to use logic programming and bounded-exhaustive testing to design and validate algorithms generating a family of combinatorial objects [45]. The focus is on computer assistance for the task of validation of an implementation with respect to a different implementation or a formal specification. Among the numerous perspectives, these generation algorithms can to their turn be embedded in bounded exhaustive testing tools, such as the one proposed in [43].

6.5. Verification of Collaborative Systems

We investigate security problems occurring in decentralized systems. We develop general techniques to enforce read and update policies for controlling access to XML documents based on recursive DTDs (Document Type Definition). Moreover, we provide a necessary and sufficient condition for undoing safely replicated objects in order to enforce access control policies in an optimistic way.

6.5.1. Automatic Analysis of Web Services Security

Participants: 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 secured services (using e.g. digital signing or timestamping) we propose a novel approach to automated orchestration of services under security constraints. Given a community of services and a goal service, we reduce the problem of generating a mediator between a client and a service community to a security problem where an intruder should intercept and redirect messages from the service community and a client service till reaching a satisfying state. In [30] we present a tool that compiles the attack trace desribing the execution of a the mediator into its corresponding runnable code. For that the tool computes an executable specification of the mediator as prudent as possible of her role in the orchestration. This specification is expressed in ASLan language, a formal language designed for modeling Web Services tied with security policies that was developed in AVANTSSAR project. Then we can check with automatic tools that this ASLan specification verifies required security properties such as secrecy and authentication. If no flaw is found, we compile the specification into a Java servlet that can be used by the mediator to execute the orchestration. This process has been implemented in AVANTSSAR Platform [29].

In [31] we give a decision procedure for the satisfiability problem of general deducibility constraints. Two cases are considered: the standard Dolev-Yao theory and its extension with an associative, commutative idempotent operator. The result is applied to solve the automated distributed orchestration problem for secured Web services.

Finall we show in [32] how to check satisfiability of negative deducibility constraints and we apply the result to the orchestration of secured services under non-disclosure policies. We show in particular how to handle separation-of-duty constraints in orchestration.

6.5.2. Secure Querying and Updating of XML Data

Participants: Abdessamad Imine, Houari Mahfoud, Michaël Rusinowitch.

Over the past years several works have proposed access control models for XML data where only read-access rights over nonrecursive DTDs are considered. A small number of works have studied the access rights for updates. In this work, we propose a general model for specifying access control on XML data in the presence of the update operations of W3C XQuery Update Facility [56], [48]. Our approach for enforcing such update specification is based on the notion of query rewriting. A major issue is that query rewriting for recursive DTDs is still an open problem [49], [55]. We show that this limitation can be avoided using only the expressive power of the standard XPath, and we propose a linear algorithm to rewrite each update operation defined over an arbitrary DTD (recursive or not) into a safe one in order to be evaluated only over the XML data which can be updated by the user. This work represents the first effort for securely XML updating in the presence of arbitrary DTDs (recursive or not) and a rich fragment of XPath. Finally, we study the interaction between read and update access rights to preserve the confidentiality and integrity of XML data.

We introduce an extension of hedge automata called bidimensional context-free hedge automata, proposing a new uniform representation of vertical and horizontal computation steps in unranked ordered trees. We also extend the parameterized rewriting rules used for modeling the W3C XQuery Update Facility in previous works, by the possibility to insert a new parent node above a given node. Since the rewrite closure of hedge automata languages with these extended rewriting systems is a computable context-free hedge language we can perform some static typechecking on these XML transformations [63].

6.5.3. On the Polling Problem in Social Networks

Participants: Bao Thien Hoang, Abdessamad Imine.

We tackle the polling problem in social networks where the privacy of exchanged information and user reputation are very critical. Indeed, users want to preserve the confidentiality of their votes and to hide, if any, their misbehaviors. Recent works proposed polling protocols based on simple secret sharing scheme and without requiring any central authority or cryptography system. But these protocols can be deployed safely provided that the social graph structure should be transformed into a ring-based structure and the number of participating users is perfect square. Accordingly, devising polling protocols regardless these constraints remains a challenging issue. In this work, we propose a simple decentralized polling protocol that relies on the current state of social graphs [58], [33]. More explicitly, we define one family of social graphs and show their structures constitute necessary and sufficient condition to ensure vote privacy and limit the impact of dishonest users on the accuracy of the output of the poll.

6.5.4. Access Control Models for Collaborative Applications

Participants: Fabrice Bouquet, Asma Cherif, Abdessamad Imine.

The importance of collaborative systems in real-world applications has grown significantly over the recent years. The most of new applications are designed in a distributed fashion to meet collaborative work requirements. Among these applications, we focus on Real-Time Collaborative Editors (RCE) that provide computer support for modifying simultaneously shared documents, such as articles, wiki pages and programming source code by dispersed users. Although such applications are more and more used into many fields, the lack of an adequate access control concept is still limiting their full potential. In fact, controlling access in a decentralized fashion in such systems is a challenging problem, as they need dynamic access changes and low latency access to shared documents. In [12], we propose a generic access control model based on replicating the shared document and its authorization policy at the local memory of each user. We consider the propagation of authorizations and their interactions. We propose an optimistic approach to enforce access control in existing collaborative editing solutions in the sense that the access control policy can be temporarily violated. To enforce the policy, we resort to the selective undo approach in order to eliminate the effect of illegal document updates. Since, the safe undo is an open issue in collaborative applications. We investigate a theoretical study of the undo problem and propose a generic solution for selectively undoing operations. Finally, we apply our framework on a collaboration prototype and measure its performance in the distributed grid GRID'5000 to highlight the scalability of our solution.

We realize the verification of Ramos protocol for concurrent writing and reconfiguration for collaborative systems in [23]. The Ramos protocol implements a fault-tolerant, and a context consistency (ensuring a total order of write operations) based on an asynchronous message-passing model. Communication takes place via gossip messages, which are sent at any frequency between a dynamic set of nodes into the collaborative system.

CELTIQUE Project-Team

5. New Results

5.1. Control-Flow Analysis by Abstract Interpretation

Control-flow analysis (CFA) of functional programs is concerned with determining how the program's functions call each other. In the case of the lambda calculus, this amounts to computing the flow of lambda expressions in order to determine what functions are effectively called in an application $(e_1\ e_2)$. This work shows that it is possible to use abstract interpretation techniques to derive systematically a control-flow analysis for a simple higher-order functional language. The analysis approximates the interprocedural control-flow of both function calls and returns in the presence of first-class functions and tail-call optimization. A number of advantages follow from taking this approach:

- The systematic derivation of a CFA for a higher-order functional language from a well-known operational semantics provides the resulting analysis with strong mathematical foundations. Its correctness follows directly from the general theorems of abstract interpretation.
- The approach is easily adapted to different variants of the source language. We demonstrate this by deriving a CFA for functional programs written in continuation-passing style.
- The common framework of these analyses enables their comparison. We take advantage of this to settle a question about the equivalence between the analysis of programs in direct and continuation-passing style.
- The resulting equations can be given an equivalent constraint-based presentation, providing *ipso facto* a rational reconstruction and a correctness proof of constraint-based CFA.

This work was published in the journal Information and Computation [14]

5.2. Secure the Clones: Static Enforcement of Policies for Secure Object Copying

Participants: Thomas Jensen, David Pichardie.

Exchanging mutable data objects with untrusted code is a delicate matter because of the risk of creating a data space that is accessible by an attacker. Consequently, secure programming guidelines for Java stress the importance of using defensive copying before accepting or handing out references to an internal mutable object.

However, implementation of a copy method (like clone()) is entirely left to the programmer. It may not provide a sufficiently deep copy of an object and is subject to overriding by a malicious sub-class. Currently no language-based mechanism supports secure object cloning.

We propose a type-based annotation system for defining modular copy policies for class-based object-oriented programs. A copy policy specifies the maximally allowed sharing between an object and its clone. We provide a static enforcement mechanism that will guarantee that all classes fulfill their copy policy, even in the presence of overriding of copy methods, and establish the semantic correctness of the overall approach in Coq.

The mechanism has been implemented and experimentally evaluated on clone methods from several Java libraries. The work as been presented at ESOP 2011. In 2012 a journal special issue has been published in Logical Methods in Computer Science [13].

5.3. A formally verified SSA-based middle-end

Participants: Delphine Demange, David Pichardie.

CompCert is a formally verified compiler that generates compact and efficient PowerPC, ARM and x86 code for a large and realistic subset of the C language. However, CompCert foregoes using Static Single Assignment (SSA), an intermediate representation that allows for writing simpler and faster optimizers, and is used by many compilers. In fact, it has remained an open problem to verify formally a SSA-based compiler middle-end.

We report on a formally verified, SSA-based, middle-end for CompCert. Our middle-end performs conversion from CompCert intermediate form to SSA form, optimization of SSA programs, including Global Value Numbering, and transforming out of SSA to intermediate form.

In addition to provide the first formally verified SSA-based middle-end, we address two problems raised by Leroy: giving a simple and intuitive formal semantics to SSA, and leveraging the global properties of SSA to reason locally about program optimizations. The work as been presented at ESOP 2012 [16].

5.4. Non linear analysis: fast inference of polynomial invariants

Participants: Thomas Jensen, David Cachera, Arnaud Jobin.

The problem of automatically inferring non-linear (polynomial) invariants of programs is still a challenge in program verification. A central observation in existing work on generating polynomial invariants is that n-ary relations between variables that can be described as the zeroes of a set of polynomials, correspond to a lattice of polynomials ideals. Such ideals are finitely generated , and all the approaches proposed so far in the literature rely on Gröbner base computations for computing ideal intersection or inclusion, or analysing the effects of polynomial assignments to variables. Computing Gröbner bases however slows down considerably the overall analysis.

We have proposed an abstract interpretation based method for inferring polynomial invariants that entirely avoids computing Gröbner bases. The method is precise and efficient, and is obtained without restricting the expressiveness of the polynomial programming language. Our analysis handles a general polynomial structured programming language that includes if and while constructs where branching conditions are both polynomial equalities and disequalities. Our analysis uses a form of weakest precondition calculus for showing that a polynomial relation $g = 0$ holds at the end of a program. We show that this backward approach, which was already observed to be well adapted to polynomial disequality guards can be extended to equality guards by using parameterized polynomial division.

Based on this anlysis, we have designed a constraint-based algorithm for inferring polynomial invariants. Such constraint-based techniques (rather than iteration) when dealing with loops means that it becomes feasible to analyse conditionals precisely, using parameterized polynomial division. A salient feature of this analysis, which distinguishes it from previous analyses, is that it does not require the use of Gröbner base computations. We have implemented this algorithm in Maple and our benchmarks show that our analyzer can successfully infer invariants on a sizeable set of examples, while performing two orders of magnitude faster than other existing implementations [19].

5.5. Result Certification of Static Analysis Results

Participants: Thomas Jensen, Frédéric Besson, Pierre-Emmanuel Cornilleau, Ronan Saillard.

Result Certification, Static program analysis, Decision procedures

We develop a lightweight approach for verifying *a posteriori* that the result of a static analysis is correct. The approach consists in encoding the program semantics directly inside an Intermediate Verification Language e.g., Why3 as an executable program interpreter. Running the standard VcGen of the IVL for the interpreter specialised for a program annotated with analysis results therefore amounts to generating program specific verification conditions [20]. This approach has the advantage of reducing the size of the Trusted Computing Base (TCB) because the VcGen is generic and language agnostic. Moreover, unlike traditional approaches, our TCB does not embed a compiler from the source code to the language of the IVL.

Verification conditions are usually discharged by Satisfiability Modulo Theory (SMT) provers that are therefore part of the TCB. To reduce further the TCB, we advocate for proof-generating SMT provers which results can be independently verified by reflexive Coq proof-checkers. For the EUF logic, we have proposed a novel compact format and proved correct an efficient Coq checker [17].

5.6. Towards efficient abstract domains for regular language based static analysis

Participants: Thomas Genet, Valérie Murat, Yann Salmon.

We develop a specific theory and the related tools for analyzing programs whose semantics is defined using term rewriting systems. The analysis principle is based on regular approximations of infinite sets of terms reachable by rewriting. The tools we develop use, so-called, Tree Automata Completion to compute a tree automaton recognizing a superset of all reachable terms. This over-approximation is then used to prove properties on the program by showing that some "bad" terms, encoding dangerous or problematic configurations, are not in the superset and thus not reachable. With such technique, like with any approximated technique, is when the "bad" terms are in the superset. We proposed a new CounterExample Guided Abstraction Refinement (CEGAR) algorithm for tree automata completion. Our approach relies on a new equational-abstraction based completion algorithm to compute a regular overapproximation of the set of reachable states in finite time. This set is represented by, so-called, R/E-automata, a new extended tree automaton formalism whose structure can be exploited to detect and remove false positives in an efficient manner. Our approach has been implemented in Timbuk and used to analyze Java programs by exploiting a translation from the Java byte code to term rewriting systems. These results have been published in [18]. Now, we aim at applying this technique to the static analysis of programming languages whose semantics is based on terms, like functional programming languages. The first step in this direction is to take into account the evaluation strategy of the language when approximating the set of reachable terms [30].

5.7. Cryptography

Participants: Pierre-Alain Fouque, Jean-Christophe Zapalowicz.

Pierre-Alain Fouque joined the team Celtique from September 2011 to August 2012. As a cryptographer, he still worked on symmetric cryptography with his PhD and postdoc students and proposed new security analysis of the block-ciphers AES and Camellia using meet-in-the-middle techniques in [27], [22] at IWSEC'12 and Indocrypt'12 and new security proofs for signature schemes AbdallaFLT12 at Eurocrypt'12 and elliptic-curve hash function [25] at LatinCrypt'12 with nice properties.

With Pierre-Alain, we also worked on more practical security aspects since his delegation in the Celtique team was to study side-channel attacks and formal methods. In side-channel attacks, we work with people from DGA and NTT in Japan to present new efficient attacks on one well-known implementation of RSA in many smartcards. Our attack targets any implementation of RSA using the Chinese Remainder Theorem in order to speed-up the computation, any exponentiation algorithm and the Montgomery multiplication. Usually, public-key cryptography requires large integer arithmetic and in order to accelerate the computation of the modulo, Montgomery proposed a new algorithm that avoids the need of arbitrary euclidean division which is the most consuming part of the exponentiation algorithm. This algorithm uses a small register (8, 16 or 32 bits depending on the architecture) during the computation and if a fault makes the value of this register much shorter, we show that we can recover the factorization of the RSA modulus in polynomial time. Furthermore, we describe on many proposed hardware architectures that our attack can indeed be used in practice if a laser is used to provoke the fault. This article has been published at CHES'12.

With people from DGA, we also studied how fault attack can be used to have buffer overflow effects. Indeed, by accelerating the clock, it is possible to avoid some instruction in the assembler code of a function. Consequently, if a fault avoids the function epilogue that restores the stack and registers to the state they were in before the function was called, then the stack pointer is changed and we can execute another function. Such attacks show that code executed in embedded processor have to be protected using buffer overflow techniques.

Finally, we also worked with people from DGA and Grenoble University to study security proofs in a computational logic. We show that the mode of operations of some hash functions is secure in [21] and published at CSF'12. In particular, we show a small bug in the security proof of the sponge construction used in the new SHA-3 candidate and winner of the competition Keccak.

COMETE Project-Team

6. New Results

6.1. Foundations of information hiding

Information hiding refers to the problem of protecting private information while performing certain tasks or interactions, and trying to avoid that an adversary can infer such information. This is one of the main areas of research in Comète; we are exploring several topics, described below.

6.1.1. Measuring information leakage

A fundamental concern in computer security is to control information flow, whether to protect confidential information from being leaked, or to protect trusted information from being tainted. In view of the pragmatic difficulty of preventing undesirable flows completely, there is now much interest in theories that allow information flow to be quantified, so that "small" leaks can be tolerated. In [19] we introduced g-leakage, a rich generalization of the min-entropy model of quantitative information flow. In g-leakage, the benefit that an adversary derives from a certain guess about a secret is specified using a gain function g. Gain functions allow a wide variety of operational scenarios to be modeled, including those where the adversary benefits from guessing a value close to the secret, guessing a part of the secret, guessing a property of the secret, or guessing the secret within some number of tries. We proved important properties of g-leakage, including bounds between min-capacity, g-capacity, and Shannon capacity. We also showed a deep connection between a strong leakage ordering on two channels, C1 and C2, and the possibility of factoring C1 into C2 C3 , for some C3 . Based on this connection, we proposed a generalization of the Lattice of Information from deterministic to probabilistic channels.

6.1.2. Interactive systems

In [12] we have considered systems where secrets and observables can alternate during the computation. We have shown that the information-theoretic approach which interprets such systems as (simple) noisy channels is not valid anymore. However, the principle can be recovered if we consider more complicated types of channels, that in Information Theory are known as channels with memory and feedback. We have shown that there is a complete correspondence between interactive systems and such kind of channels. Furthermore, we have shown that the capacity of the channels associated to such systems is a continuous function of the Kantorovich metric.

6.1.3. Unlinkability

Unlinkability is a privacy property of crucial importance for several systems (such as RFID or voting systems). Informally, unlinkability states that, given two events/items in a system, an attacker is not able to infer whether they are related to each other. However, in the literature we find several definitions for this notion, which are apparently unrelated and shows a potentially problematic lack of agreement. In [22] we shed new light on unlinkability by comparing different ways of defining it and showing that in many practical situations the various definitions coincide. It does so by (a) expressing in a unifying framework four definitions of unlinkability from the literature (b) demonstrating how these definitions are different yet related to each other and to their dual notion of "inseparability" and (c) by identifying conditions under which all these definitions become equivalent. We argued that the conditions are reasonable to expect in identification systems, and we prove that they hold for a generic class of protocols.

6.1.4. A compositional method to compute the sensitivity of differentially private queries

Differential privacy is a modern approach in privacy-preserving data analysis to control the amount of information that can be inferred about an individual by querying a database. The most common techniques are based on the introduction of probabilistic noise, often defined as a Laplacian parametric on the sensitivity of the query. In order to maximize the utility of the query, it is crucial to estimate the sensitivity as precisely as possible.

In [28] we considered relational algebra, the classical language for expressing queries in relational databases, and we proposed a method for computing a bound on the sensitivity of queries in an intuitive and compositional way. We used constraint-based techniques to accumulate the information on the possible values for attributes provided by the various components of the query, thus making it possible to compute tight bounds on the sensitivity.

6.1.5. A differentially private mechanism of optimal utility for a region of priors

Differential privacy (already introduced in the previous section) is usually achieved by using mechanisms that add random noise to the query answer. Thus, privacy is obtained at the cost of reducing the accuracy, and therefore the utility, of the answer. Since the utility depends on the user's side information, commonly modeled as a prior distribution, a natural goal is to design mechanisms that are optimal for every prior. However, it has been shown in the literature that such mechanisms do not exist for any query other than counting queries.

Given the above negative result, in [38] we considered the problem of identifying a restricted class of priors for which an optimal mechanism does exist. Given an arbitrary query and a privacy parameter, we geometrically characterized a special region of priors as a convex polytope in the priors space. We then derived upper bounds for utility as well as for min-entropy leakage for the priors in this region. Finally we defined what we call the tight-constraints mechanism and we discussed the conditions for its existence. This mechanism has the property of reaching the bounds for all the priors of the region, and thus it is optimal on the whole region.

6.1.6. Differential privacy with general metrics

Differential privacy, already described above, is a formal privacy guarantee that ensures that sensitive information relative to individuals cannot be easily inferred by disclosing answers to aggregate queries. If two databases are adjacent, i.e. differ only for an individual, then querying them should not allow to tell them apart by more than a certain factor. The transitive application of this property induces a bound also on the distinguishability of two generic databases, which is determined by their distance on the Hamming graph of the adjacency relation.

In [37] we lifted the restriction relative to the Hamming graphs and we explored the implications of differential privacy when the indistinguishability requirement depends on an arbitrary notion of distance. We showed that we can express, in this way, (protection against) kinds of privacy threats that cannot be naturally represented with the standard notion. We gave an intuitive characterization of these threats in terms of Bayesian adversaries, which generalizes the characterization of (standard) differential privacy from the literature. Next, we revisited the well-known result on the non-existence of universally optimal mechanisms for any query other than counting queries. We showed that in our setting, for certain kinds of distances, there are many more queries for which universally optimal mechanisms exist: Notably sum, average, and percentile queries. Finally, we showed some applications in various domains: statistical databases where the units of protection are groups (rather than individuals), geolocation, and smart metering.

6.1.7. Privacy for location-based systems

The growing popularity of location-based systems, allowing unknown/untrusted servers to easily collect and process huge amounts of users' information regarding their location, has recently started raising serious concerns about the privacy of this kind of sensitive information. In [36] we studied geo-indistinguishability, a formal notion of privacy for location-based systems that protects the exact location of a user, while still allowing approximate information - typically needed to obtain a certain desired service - to be released.

Our privacy definition formalizes the intuitive notion of protecting the user's location within a radius r with a level of privacy that depends on r. We presented three equivalent characterizations of this notion, one of which corresponds to a generalized version [37] of the well-known concept of differential privacy. Furthermore, we presented a perturbation technique for achieving geo-indistinguishability by adding controlled random noise to the user's location, drawn from a planar Laplace distribution. We demonstrated the applicability of our technique through two case studies: First, we showed how to enhance applications for location-based services with privacy guarantees by implementing our technique on the client side of the application. Second,

we showed how to apply our technique to sanitize location-based sensible information collected by the US Census Bureau.

6.1.8. Compositional analysis of information hiding

Systems concerned with information hiding often use randomization to obfuscate the link between the observables and the information to be protected. The degree of protection provided by a system can be expressed in terms of the probability of error associated to the inference of the secret information. In [15] we considered a probabilistic process calculus to specify such systems, and we studied how the operators affect the probability of error. In particular, we characterized constructs that have the property of not decreasing the degree of protection, and that can therefore be considered safe in the modular construction of these systems. As a case study, we applied these techniques to the Dining Cryptographers, and we derived a generalization of Chaum's strong anonymity result.

In [29], a similar framework was proposed for reasoning about the degree of differential privacy provided by such systems. In particular, we investigated the preservation of the degree of privacy under composition via the various operators. We illustrated our idea by proving an anonymity-preservation property for a variant of the Crowds protocol for which the standard analyses from the literature are inapplicable. Finally, we made some preliminary steps towards automatically computing the degree of privacy of a system in a compositional way.

6.1.9. Anonymous and route-secure communication systems

Incentives to Cooperation. Anonymity systems have a broad range of users, ranging from ordinary citizens who want to avoid being profiled for targeted advertisements, to companies trying to hide information from their competitors, to entities requiring untraceable communication over the Internet. With these many potential users, it would seem that anonymity services based on a consumer/provider users will naturally be well-resourced and able to operate efficiently. However, cooperation cannot be taken for granted. Current deployed systems show that some users will indeed act selfishly, and only use the system to send their messages whilst ignoring the requests to forward others' messages. Obviously, with not enough cooperative users, the systems will hardly operate at all, and will certainly not be able to afford adequate anonymity guarantees. It is therefore vital that these systems are able to deploy incentives to encourage users' cooperation and so make the anonymity provision effective. Some interesting approaches to achieve that have been proposed, such as make running relays easier and provide better forwarding performance.

To evaluate whether these approaches are effective, we need a framework which empowers us to analyze them, as well as provide guidelines and some mechanism design principles for incentive schemes. This much we have provided in [30], exploiting notions and techniques from Game Theory. We proposed a game theoretic framework and used it to analyze users' behaviours and also predict what strategies users will choose under different circumstances and according to their exact balance of preferences among factors such as anonymity, performance (message delivery time) and cost. Significantly, we also used the model to assess the effectiveness of the gold-star incentive mechanism, which was introduced in Tor network to encourage users to act as cooperative relays, and thus enhance the service performance for well-behaved forwarders.

Trust in anonymity networks. Trust metrics are used in anonymity networks to support and enhance reliability in the absence of verifiable identities, and a variety of security attacks currently focus on degrading a user's trustworthiness in the eyes of the other users. In [16] we have presented an enhancement of the Crowds anonymity protocol via a notion of trust which allows crowd members to route their traffic according to their perceived degree of trustworthiness of each other member of the crowd. Such trust relations express a measure of an individual's belief that another user may become compromised by an attacker, either by a direct attempt to corrupt or by a denial-of-service attack. Our protocol variation has the potential of improving the overall trustworthiness of data exchanges in anonymity networks, which cannot normally be taken for granted in a context where users are actively trying to conceal their identities. Using such formalization, in the paper we have then analyzed quantitatively the privacy properties of the protocol under standard and adaptive attacks.

6.2. Foundations of Concurrency

Distributed systems have changed substantially in the recent past with the advent of phenomena like social networks and cloud computing. In the previous incarnation of distributed computing the emphasis was on consistency, fault tolerance, resource management and related topics; these were all characterized by *interaction between processes*. Research proceeded along two lines: the algorithmic side which dominated the Principles Of Distributed Computing conferences and the more process algebraic approach epitomized by CONCUR where the emphasis was on developing compositional reasoning principles. What marks the new era of distributed systems is an emphasis on managing access to information to a much greater degree than before.

The *Concurrent constraint programming (ccp)* paradigm focuses on information access and therefore it is suited for this new era of concurrent systems. Ccp singles out the fundamental aspects of asynchronous systems whose agents (or processes) evolve by accessing information in a global medium. In the works [20], [21], [31], [26] described below we developed algorithms and extended the foundations of ccp.

6.2.1. Spatial and Epistemic Modalities for Constraint-based Calculi

Epistemic concepts were crucial in distributed computing as was realized in the mid 1980s with Halpern and Moses' groundbreaking paper on common knowledge. This led to a flurry of activity in the next few years with many distributed protocols being understood from an epistemic point of view. The impact of epistemic ideas in the concurrency theory community was slower in coming. We believe that epistemic ideas need to be exploited more by concurrency theorists and we did so in the following works.

In [26] we introduced spatial and epistemic process calculi for reasoning about spatial information and knowledge distributed among the agents of a system. We also introduced domain-theoretical structures to represent spatial and epistemic information. Finally we provided operational and denotational techniques for reasoning about the potentially infinite behaviour of spatial and epistemic processes. We also gave compact representations of infinite objects that can be used by processes to simulate announcements of common knowledge and global information. We also developed an interpreter of these calculi in [31].

6.2.2. Bisimilarity for Constraint-based Calculi

Bisimilarity is a standard behavioural equivalence in concurrency theory, but a well-behaved notion of bisimilarity for ccp has been proposed only recently. When the state space of a system is finite, the ordinary notion of bisimilarity can be computed via the well-known partition refinement algorithm, but unfortunately, this algorithm does not work for ccp bisimilarity. In [20] we proposed a variation of the partition refinement algorithm for verifying ccp bisimilarity. To the best of our knowledge this is the first work providing for the automatic verification of program equivalence for ccp.

In [20] we only studied the strong version of bisimilarity. Weak bisimiliarity is obtained from the strong case by taking into account only the actions that are observable in the system. Typically, the standard partition refinement can also be used for deciding weak bisimilarity simply by using Milner's reduction from weak to strong bisimilarity; a technique referred to as saturation. In [21] we showed that, because of its involved labeled transitions, the above-mentioned saturation technique does not work for ccp. We also gave an alternative reduction from weak ccp bisimilarity to the strong one that allows us to use the ccp partition refinement algorithm for deciding this equivalence.

In the more traditional setting of the pi-calculus we have also proposed an approach to restrict access to information.

6.2.3. Locality in the Pi-Calculus

In [25] we enriched the pi-calculus with an operator for confidentiality (hide), whose main effect is to restrict the access to the object of the communication, thus representing confidentiality in a natural way. The hide operator is meant for local communication, and it differs from new in that it forbids the extrusion of the name and hence has a static scope. Consequently, a communication channel in the scope of a hide can be implemented as a dedicated channel, and it is more secure than one in the scope of a new. To emphasize the

difference, we introduced a spy context that represents a side-channel attack and breaks some of the standard security equations for new. To formally reason on the security guarantees provided by the hide construct, we also introduced an observational theory and establish stronger equivalences by relying on a proof technique based on bisimulation semantics.

6.2.4. Foundations of Probabilistic Concurrent Systems

In [17] we have solved an open problem in the literature by proving that two known semantics for the probabilistic mu-calculus, a denotational semantics and a two-player stochastic game semantics, coincide on all models.

In [18] we have improved the result of [17] by introducing a new logic called probabilistic mu-calculus with independent product. We have proved that two semantics coincide in all models: a denotational semantics and a two-player game semantics based on a novel class of concurrent games. Furthermore, we have shown how the new logic is strictly more expressive than the other. This allows the encoding of other important temporal logics for probabilistic concurrent systems such as PCTL.

In [27] we have introduced a proof system designed for supporting human-aided verification of properties (expressed as probabilistic mu-calculus formulas ([17]) of concurrent probabilistic processes described by SOS-style operational semantics.

6.2.5. Interference metrics for Mobile ad-hoc networks (MANETs)

Mobile ad-hoc networks consist of a collection of nodes that communicate with each other through wireless links without a pre-established networking infrastructure. A common feature of most of these networks is free node mobility. Each device will therefore change its links to other devices frequently. These frequent changes in the network topology can cause the nodes to continuously enter and exit each other transmission area. Hence, highly dynamic routing algorithms are needed to ensure the connectivity. Moreover, mobile devices may have strict requirements on the energy consumption because their expected life-time often depends on the energy stored in a battery or other exhaustible power sources. For these reasons, finding a good trade-off between network connectivity, power saving and interference reduction is one of the most critical challenges in managing mobile ad hoc networks. In [23], we have proposed an effective framework for analysing protocol connectivity and measuring the level of interference and, based on that for developing novel interference-aware communication strategies. Though other models exist in the literature, to our best knowledge, our framework is the most comprehensive and effective for the behavioral analysis and a quantitative assessment of interference for wireless networks in the presence of node mobility.

CONTRAINTES Project-Team

6. New Results

6.1. Inferring Reaction Rule Models from Ordinary Differential Equations

Participants: François Fages, Steven Gay, Sylvain Soliman.

Many models in Systems Biology are described as Ordinary Differential Equations (ODEs), which allow for numerical integration, bifurcation analyses, parameter sensitivity analyses, etc. However, before fixing the kinetics and parameter values and going to simulations, various analyses can be performed based only on the structure of the model. This approach has rapidly developed in Systems Biology in the last decade, with for instance, the analyses of structural invariants in Petri net representation, model reductions by subgraph epimorphims, qualitative attractors in logical dynamics or temporal logic properties by analogy to circuit and program verification. These complementary analysis tools do not rely on kinetic information, but on the structure of the model with reactions.

In [8], [19], we present a symbolic computation algorithm for inferring a reaction model from an ODE system, based a general compatibility condition between the kinetic expression and the structure of a reaction, and report on its use for automatically curating the writing in SBML of the models in the respository biomodels.net. SBML is now a standard for sharing and publishing reaction models. However, since SBML does not enforce any coherence between the structure and the kinetics of a reaction, an ODE model can be transcribed in SBML without reflecting the real structure of the reactions, hereby invalidating many structural analyses. We show that the automatic writing in SBML of the models of biomodels.net allows us to reduce the percentage of models with a non well-formed reaction from 66% to 28%.

6.2. Petri Net Analyses of Biochemical Networks using Constraint Logic Programming

Participants: François Fages, Thierry Martinez, Faten Nabli, Sylvain Soliman.

Petri nets are a simple formalism for modeling concurrent computation. Recently, they have emerged as a promising tool for modeling and analyzing biochemical interaction networks, bridging the gap between purely qualitative and quantitative models. Biological networks can indeed be large and complex, which makes their study difficult and computationally challenging.

In [10], we focus on two structural properties of Petri nets, siphons and traps, that bring us information about the persistence of some molecular species. We present a Boolean model and two constraint-based methods for enumerating all minimal siphons and traps of a Petri net, by iterating the resolution of Boolean satisfiability problems executed with either a SAT solver or a CLP(B) program. We compare the performances of these methods with respect to a state-of-the-art algorithm from the Petri net community. On a benchmark with 80 Petri nets from the Petriweb database and 403 Petri nets from curated biological models of the Biomodels database, we show that miniSAT and CLP(B) solvers are overall both faster by two orders of magnitude with respect to the dedicated algorithm. Furthermore, we analyse why these programs perform so well on even very large biological models and show a polynomial time complexity result for Petri nets of fixed treewidth, using a similar theorem for constraint satisfaction problems with bounded treewidth constraint graphs.

In [5] we present a method to compute the minimal semi-positive invariants of a Petri net representing a biological reaction system, as resolution of a Constraint Satisfaction Problem. This analysis brings both qualitative and quantitative information on the models, in the form of conservation laws, consistency checking, etc. thanks to finite domain constraint programming. It is noticeable that some of the most recent optimizations of standard invariant computation techniques in Petri nets correspond to well-known techniques in constraint solving, like symmetry-breaking. A simple implementation based on GNU-Prolog's finite domain solver, and including symmetry detection and breaking, was incorporated into the BIOCHAM modelling environment and in the independent tool Nicotine. Some illustrative examples and benchmarks are provided.

6.3. Subgraph Epimorphisms

Participants: François Fages, Steven Gay, Thierry Martinez, Francesco Santini, Sylvain Soliman.

The operations of deleting and merging vertices are natural operations for reducing a graph. While graph reductions through a sequence of vertex deletions (resp. mergings) characterize subgraph isomorphisms (resp. graph epimorphisms), sequences of both vertex deletion and merging operations characterize subgraph epimorphisms. Our proposal is thus to use subgraph epimorphism for comparing graphs in applications in systems biology and image analysis, when a more flexible notion than the classical notion of subgraph isomorphism is required.

In collaboration with Christine Solnon (INSA Lyon), we have developed the theory of subgraph epimorphisms. We have defined the SEPI, EPI and SISO distances between two graphs as the size of the largest SEPI (resp. EPI, SISO) lower bound graphs. These distances are equal to the minimum number of respectively vertex deletion and/or merging operations that are necessary to obtain isomorphic graphs. They are also metrics on graphs and we have $d_d \geq d_{md}$ and $d_m \geq d_{md}$. From a computational point of view, we have shown that the existence of a SEPI between two graphs is an NP-complete problem and have presented a constraint satisfaction algorithm for solving it.

Our algorithm is implemented in BIOCHAM and is currently improved for better performance on large graphs and generalized as a SEPI graph constraint propagation algorithm for computing SEPI lower and upper bounds.

6.4. Parameter Search with Temporal Logic Constraints

Participants: Grégory Batt, François Fages, Anthony Lins, Sylvain Soliman, Pauline Traynard, Jannis Uhlendorf, Luma Vittorino.

Our method for solving temporal logic constraints in first-order linear time logic LTL(R_{lin}), opens up the field of model-checking to optimization through the definition of a continuous degree of satisfaction for temporal logic formulae. This satisfaction degree can be used in a number of ways, e.g. as a fitness function with continuous optimization methods to find unknown parameter values in a model, to perform sensitivity analyses and compute the robustness of a system w.r.t. a temporal property and a perturbation of the parameters. or to find control parameters.

This approach is implemented in BIOCHAM and is one unique feature of this modeling environment. In this implementation, the continuous optimization procedure we use is the Covariance Matrix Adaptation Evolutionary Strategy CMAES of Nikolaus Hansen from the EPI TAO. A parallel version of Biocham implements this method on the Jade cluster of 10000 cores at GENCI for running our most challenging parameter search problems.

This year, in collaboration with Fernando Buarque, we have explored another continuous optimization method of the family of Particle Swarm Optimization (PSO), called Fish School Optimization (FSS). In [13], we report on our first results which are encouraging for using FSS for decreasing the sensitivity of the method to initial conditions and being able to maintain several swarms of solutions.

6.5. Coupled Model of the Cell Cycle and Circadian Clock

Participants: François Fages, Sylvain Soliman, Denis Thieffry, Pauline Traynard.

Recent advances in cancer chronotherapy techniques support the evidence that there exist imortant links between the cell cycle and the circadian clock genes. One purpose for modeling these links is to better understand how to efficiently target malignant cells depending on the phase of the day and patient characterictics. This is at the heart of our participation in collaboration with the EPI BANG in the EraNet SysBio project C5Sys, follow up of the former EU STREP project TEMPO.

This year we have investigated the effect of transcription inhibition during mitosis, as a reverse coupling from the cell cycle to the circadian clock. We use temporal logic constraints and the parallel version of BIOCHAM for parameter search, running on the Jade cluster of 10000 processors at the GENCI CINES, to couple dynamical models in high dimension and fit models to experimental data time series obtained in Franck Delaunay's lab in Nice, CNRS.

6.6. STL-based Analysis of TRAIL-induced Apoptosis

Participants: Grégory Batt, François Bertaux, Szymon Stoma.

Extrinsic apoptosis is a programmed cell death triggered by external ligands, such as the TNF-related apoptosis inducing ligand (TRAIL). Depending on the cell line, the specific molecular mechanisms leading to cell death may significantly differ. Precise characterization of these differences is crucial for understanding and exploiting extrinsic apoptosis. Cells show distinct behaviors on several aspects of apoptosis, including (i) the relative order of caspases activation, (ii) the necessity of Mitochondria Outer Membrane Permeabilization (MOMP) for effector caspase activation, and (iii) the survival of cell lines overexpressing Bcl2, leading to classification of cell lines into two groups (type I and type II). In [21], we challenge this type I/II cell line classification. We encode the three aforementioned distinguishing behaviors in a formal language, called signal temporal logic (STL), and use it to extensively test the validity of a previously-proposed model of TRAIL-induced apoptosis with respect to experimental observations made on different cell lines. Then, STL-guided parameter search is used to solve the few inconsistencies found between model and data. We show that these three criteria do not define consistent cell line classifications in type I or type II, and suggest mutants that are predicted to exhibit ambivalent behaviors. In particular, this finding sheds light on the role of a feedback loop between caspases, and reconciliates two apparently-conflicting views regarding the importance of either upstream or downstream processes for cell type determination. More generally, our work suggests that rather than being considered as defining criteria for cell type classification, these three distinguishing behaviors should be merely considered as type I or II features. On the methodological point of view, this work illustrates the biological relevance of STL-diagrams, STL population data, and STL-guided parameter search. Such tools are well adapted to the ever-increasing availability of heterogeneous knowledge on complex signal transduction pathways.

6.7. Real-time Control of Gene Expression in Yeast

Participants: Grégory Batt, François Fages, Jannis Uhlendorf, Jean-Baptiste Lugagne, Artémis Llamosi, Pascal Hersen.

Gene expression plays a central role in the orchestration of cellular processes. The use of inducible promoters to change the expression level of a gene from its physiological level has significantly contributed to the understanding of the functioning of regulatory networks. However, from a quantitative point of view, their use is limited to short-term, population-scale studies to average out cell-to-cell variability and gene expression noise and limit the nonpredictable effects of internal feedback loops that may antagonize the inducer action. In this project, in collaboration with the Hersen Lab at MSC (Paris Diderot University), we show that, by implementing an external feedback loop, one can tightly control the expression of a gene over many cell generations with quantitative accuracy. To reach this goal, we developed a platform for real-time, closed-loop control of gene expression in yeast that integrates microscopy for monitoring gene expression at the cell level, microfluidics to manipulate the cells environment, and original software for automated imaging, quantification, and model predictive control. By using an endogenous osmostress responsive promoter and playing with the osmolarity of the cells environment, we show that long-term control can, indeed, be achieved for both time-constant and time-varying target profiles at the population and even the single-cell levels [6]. Importantly, we provide evidence that real-time control can dynamically limit the effects of gene expression stochasticity. We anticipate that our method will be useful to quantitatively probe the dynamic properties of cellular processes and drive complex, synthetically engineered networks.

6.8. Genome Engineering of Mammalian Cells: Targeted and Efficient Integration of Multi-unit Genetic Payloads

Participants: Grégory Batt, Xavier Duportet.

Targeted integration of multi-unit genetic payloads would greatly benefit elucidating complex cellular mechanisms and implementing new functions in mammalian cells. Current technologies are however time-consuming and require tedious post-integration controls. To address this problem, we propose a modular framework to assemble large multi-unit genetic payloads and target their integration into either one or both alleles of a chromosomal locus of choice. To achieve this, we combine in a two-step process the customizable targeting properties of homing endonucleases with the efficiency and specificity of a large serine recombinase. We have demonstrated that an optimized version of BxB1 recombinase allows the targeted integration of large genetic circuits (up to 7 transcription units, 60kb) into a preintegrated landing pad in the AAVS1 locus, with a significant increase in efficiency compared to other site-specific recombination systems (integration in 10% of transfected cells without selection) . By reducing the time and efforts to generate large populations of isogenic stable cell lines adapted to study multi-component genetic systems, our framework is a valuable tool for mammalian synthetic biology and offers great potential for a broad range of biotechnology and therapeutic applications.

6.9. Reifying Global Constraints

Participants: François Fages, Raphaël Martin, Thierry Martinez, Sylvain Soliman.

Global constraints were introduced two decades ago as a means to model some core aspects of combinatorial problems with one single constraint for which an efficient domain filtering algorithm can be provided, possibly using a complete change of representation. However, global constraints are just constraint schemas on which one would like to apply usual constraint operations such as reification, i.e. checking entailment, disentailment and negating the constraint. This is currently not the case in state-of-the-art tools and was not considered in the global constraint catalog until recently. In [20], we propose a general framework for reifying global constraints and apply it to some important constraints of the catalog, such as the cumulative constraint for instance. We show that several global constraints that were believed to be hard to negate can in fact be efficiently negated, and that entailment and disentailment can be efficiently tested. We also point out some new global constraints that are worth studying from this point of view and provide some performance figures obtained with an implementation in Choco.

This scheme is currently used for compiling the Rules2CP constraint modeling language to Choco, and to internalize search in CSPs through constraint reification.

6.10. Railway Time Tabling Optimization

Participants: François Fages, David Fournier, Thierry Martinez, Sylvain Soliman.

Metros are able to generate electricity on a metro line by braking. This energy is immediately available in the third rail and is lost if no metro in the neighbourhood can consume it. It is thus possible to decrease the total energy consumption of a metro line by synchronizing the accelerations and braking of the metros. In [2], [9], we propose a classification of energy optimization timetable problems and we present a model for optimizing energy consumption which does not significantly alter the quality of service, by subtly modifying dwell times. We show however that this optimization problem is NP-hard. We present a hybrid genetic/linear programming algorithm for computing the distribution of braking metros. In this hybridization, the objective function is computed by a linear program and by a heuristic, and the dwell times are modified by a genetic algorithm. On a typical example with real data, the savings exceed 7%. Furthermore, on a benchmark of the literature for a simpler problem, we discuss the results obtained with our genetic algorithm, a tabu search algorithm and the mixed integer linear program used by the authors.

DEDUCTEAM Team

5. New Results

5.1. Dedukti

Together with Mathieu Boespflug (McGill University), Quentin Carbonneaux and Ronan Saillard have developed a new version of the front-end of Dedukti, written in OCaml, replacing an inefficient previous version, as well as a new version of the back-end using the Lua Just-In-Time compiler.

Ronan Saillard has internalized the Lua back-end of Dedukti, so that it is no longer necessary to explicitly call it when using Dedukti.

Ronan Saillard has extended the input language of Dedukti to allow the user to declare dependencies between modules, to write definition or to explicitly require to type-check a term.

Ronan Saillard has added a new feature to Dedukti to make opaque definitions. As with usual definitions, the proof term of an opaque definition is type-checked, but it is then immediately forgotten in order to decrease memory consumption.

5.2. Embeddings in the $\lambda\Pi$-calculus modulo

Ali Assaf has designed an embedding of the HOL logic in the $\lambda\Pi$-calculus modulo and implemented it in the HOLiDe system [40].

Together with Mathieu Boespflug, Ali Assaf and Guillaume Burel have developed an embedding of the Calculus of Inductive Constructions with universes in the $\lambda\Pi$-calculus modulo and Ali Assaf is currently implementing it in a new version of the CoqInE system.

Catherine Dubois and Raphaël Cauderlier have studied a translation in the $\lambda\Pi$-calculus modulo of features coming from object oriented programming languages, such as inheritance and late binding. This compilation scheme has been applied to produce a new back-end for FoCaLize [8], through a compilation to Dedukti. This new back-end is expected to be lighter than the present one producing Coq code and also to be able to combine local and external proofs coming from different proof environments [44]. They are currently working on a translation of the full FoCaLize language—not restricted to its object oriented features—and on a proof of its correctness with respect to the existing FoCaLize semantics.

5.3. Automated Theorem Proving

Guillaume Burel has shown that presenting theories by means of rewriting rules in Deduction modulo leads to more efficient proof search methods than using axioms, provided the rewriting system enjoys a proof theoretical property, namely cut admissibility.

He has been investigating which theories can be encoded as rewriting systems admitting cuts. Surprisingly, it turned out that any consistent theory in predicate logic can. This has been shown by studying the links between the set-of-support strategy of the Resolution method and the extension of the method based on Deduction modulo. He has also shown how to reduce the size of the corresponding rewriting systems [42].

Guillaume Burel has also studied how to improve the confidence in iProver Modulo. When it finds a resolution proof, it is now able to produce a proof that can be checked by Dedukti. The encoding of Resolution proofs in the $\lambda\Pi$-calculus modulo that is used is shallow, making more plausible the long-term goal of interoperability of provers, both interactive and automated, through Dedukti.

Simon Cruanes has explored several ideas for combining the Superposition calculus—one of the most powerful calculi for automated reasoning within first-order logic with equality—with Deduction modulo. Combining the term rewriting system for a theory in Deduction modulo with the ordered rewriting on which Superposition is based on proved to be difficult, yielding incomplete calculi; in most cases it boils down to the fact that the combination of confluent terminating term rewriting systems is in general neither terminating nor confluent. In order to experiment quickly ideas by implementing them, he has written a Superposition-based prover in OCaml, with some special features—automatic ordering of rewrite rules in the input, non-clausal calculus to be able to use equivalence relations as rewrite rules. The prover is 8,000 lines of code and is designed to be flexible and modular, but still has decent performance and can prove some non-trivial theorems.

Together with Mélanie Jacquel (Cedric), David Delahaye and Catherine Dubois have investigated Zenon for verifying proof rules added to help the automation in the provers of Atelier B. They have augmented Zenon with specific rules for dealing with set operations and predicates, obtained by applying super deduction—a variant of Deduction modulo [33].

5.4. Proof theory

We believe that our work on proof-checking and automated theorem proving cannot be separated from a more theoretical research on proof theory.

Together with Denis Cousineau, Gilles Dowek and Olivier Hermant have related semantic criteria for proof normalization and admissibility of the cut rule in Deduction modulo [17], [26].

Gilles Dowek has proposed a new way to define classical connectives in a constructive framework [46].

Together with Murdoch J. Gabbay (Heriot Watt), Gilles Dowek has proposed a new nominal logic that handles binders in terms [16] and a new semantics for predicate logic [29].

During her visit in the team, Cecilia Englander has studied the correspondence between natural deduction and sequent calculus.

Together with Ying Jiang (Beijing), Gilles Dowek has defined a logic for finite structures. Kailiang Ji is currently investigating the use of proof search algorithms in Deduction modulo to automatically prove theorems in this theory.

5.5. Safety of aerospace systems

Together with Anthony Narkawicz (Nasa-Langley) and César Muñoz (Nasa-Langley), Gilles Dowek has designed a prevention bands algorithm, that is an algorithm that computes and displays to the pilot of an aircraft, a sequence of safe and unsafe intervals on ground speed, heading or vertical speed and they have proved this algorithm correct in the PVS system [18].

This algorithm computes with real numbers, but its implementation computes with floating point numbers. Moreover this algorithm is numerically unstable as it uses comparisons of numbers, computed with square root and division operations. This has led Pierre Néron to design a program transformation algorithm to eliminate square roots and divisions in straight-line programs. This way computation can be made exact.

Together with César Muñoz, Pierre Néron has completed this year the design of this program transformation algorithm and he has proved, in the PVS system, its termination and correctness: preservation of semantics and absence of square roots and divisions in the produced program [35].

Together with César Muñoz, Pierre Néron has also implemented this transformation algorithm as a PVS automatic proof strategy, that allows a wider range of expressions, using a deep embedding of PVS in PVS itself.

Pierre Néron and Raphaël Bost have proposed an optimization of one aspect of that algorithm: the definition of a common template for arithmetic expression.

5.6. Constraint Solving

Catherine Dubois has developed in collaboration with Matthieu Carlier and Arnaud Gotlieb (Oslo) a formally verified constraint finite domain solver. It focuses on arc-consistency and has been developed with Coq [24].

5.7. Models of Computation

Together with Pablo Arrighi (Grenoble), Gilles Dowek has reformulated Gandy's proof of the physical Church-Thesis in the quantum case [11]. Gilles Dowek has proposed the idea that the Galileo thesis could be seen as a consequence of the physical Church-Turing thesis and therefore as a consequence of Gandy's principles [15]. Gilles Dowek has proposed a definition of a notion of non deterministic computation over the real numbers [14] that could be used as a language to describe continuous non deterministic physical phenomena. All this work has then been presented in a tutorial at the conference *Language and Automata Theory and Applications* [28].

Together with Pablo Arrighi, Gilles Dowek has investigated further the principle of a finite density of information [38] and in particular the impact of this definition on the notion of a chaotic dynamical system [37].

Together with Pablo Arrighi, Gilles Dowek has investigated a generalization of the notion of cellular automaton where the principle of a bounded density of information is formulated independently of the geometry of space. This led to the notion of a Causal graph dynamic [12].

Nachum Dershowitz and Gilles Dowek have shown that extending Turing machines with a two-dimensional tape, made this formalism usable in practice to implement classical algorithms [45].

Alejandro Díaz-Caro and Gilles Dowek have proposed to take a fresh look at non deterministic λ-calculi—such as quantum λ-calculi—and derive non determinism from type isomorphism [30].

Together with Giulio Manzonetto (Paris 13) and Michele Pagani (Paris 13), Alejandro Díaz-Caro has considered an extension of the call-by-value λ-calculus with a may-convergent non-deterministic choice and a must-convergent parallel composition, endowed with a type system. They have proved that a term is typable if and only if it is converging, and that its typing tree carries enough information to give a bound on the length of its lazy call-by-value reduction. Moreover, when the typing tree is minimal, such a bound becomes the exact length of the reduction [31].

Together with Barbara Petit (Sardes), Alejandro Díaz-Caro has considered the non-deterministic extension of the call-by-value lambda calculus, which corresponds to the additive fragment of the linear-algebraic lambda-calculus. They have defined a fine-grained type system, capturing the right linearity present in such formalisms. After proving the subject reduction and the strong normalisation properties, they have proposed a translation of this calculus into the System F with pairs, which corresponds to a non linear fragment of linear logic. The translation provides a deeper understanding of the linearity in this setting [32].

Together with Pablo Arrighi, Barbara Petit, Pablo Burias (Rosario), Mauro Jaskelioff (Rosario), and Benoît Valiron (Penn), Alejandro Díaz-Caro has studied possible typing systems for the full linear-algebraic λ-calculus in which the non-deterministic calculus can be seen as a particular case. They have proposed a type system that keeps track of "the amount of a type" that is present in each term [13]. As an example of its use, they have shown that it can serve as a guarantee that the normal form of a term is barycentric, that is that its scalars are summing to one. They also proposed a type system similar to the one presented in [32], but for the full calculus, ensuring confluence and convergence [23]. Finally, they provided a full type system that is able to statically describe the linear combinations of terms resulting from the reduction of programs, also ensuring convergence [19].

FORMES Team

6. New Results

6.1. Higher-Order Abstract Syntax

This recently started project funded by the National Science Foundation of China aims at setting up a generic infrastructure for representating logical systems and automate their meta-theoretical study. We view a logical system as a type theory made of three components: a language of terms, types being particular terms; a set of typing rules; and a set of computational rules described by typed higher-order rewrite rules.

There are several challenges in this project. The first is to define logical frameworks which are expressive enough -at least as expressive as Girard's System F or Edingburgh's LF- to define the syntax and semantics of rich type theories, such as CoqMTU as an extreme example. A second challenge is to develop new techniques for checking the three main properties of higher-order rewrite rules: type preservation -which is usually easy-, confluence and termination. Our work here has progressed steadily, in paticular with new advanced techniques for checking termination and confluence described next. A third challenge is to formalize these results in Coq, in order to provide proof certificates for particular cases. The fourth challenge is to build a a general infrastructure in Coq in which all these techniques become available in order to study particular logical systems.

As initial steps, we undertook the following formalizations :

- Hua Mei implemented an intensional framework for simply typed lambda-calculus in Coq, where α- and β-conversions have been axiomized.
- Frédéric Blanqui has formalized in Coq the pure lambda-calculus following the definition of Curry and Feys in [43] (named variables and explicit alpha-equivalence), and the proof of termination of β-reduction for simply-typed λ-terms based on computability predicates [51]. To the best of his knowledge, this is the first formalization of the termination of β-reduction using named variables and explicit alpha-equivalence, all the other formalizations using De Bruijn indices [73] or nominal logic [48].
- Qian Wang formalized completely the theory of CoqMTU in Coq augmented with strong set-theoretic axioms in order to get around Gödel's incompleteness theorem. This is described in more details next.

6.2. CoqMTU

The proof-assistant Coq is based on a complex type theory, which resulted from various extensions of the Calculus of Constructions studied independently fromf each other. With Bruno Barras, we decided to address the challenge of proving the real type theory underlying Coq, and even, indeed, its recent extension CoqMT. To this end, we have studied formally the theory CoqMTU, which extends the calculus of Constructions with inductive types, a predicative hierarchy of universes and a decidable theory T for some first-order inductive types for which large elimination is no more available. This work has been published at LICS [1]. It leaves open the question whether large elimination can be accomadated for those inductive types which carry along a decidable theory T. This problem has been solved recently by Wang, who constructed a set-theoretic model of CoqMTU with strong elimination.

6.3. Normal Rewriting

There are many forms of rewriting used in the litterature: plain rewriting (rules are fired via plain pattern matching), rewriting modulo T (rules are fired via pattern matching modulo T), higher-order rewriting (rules are fired via higher-order pattern matching, but apply to simply typed lambda-terms terms provided the redex is of base type and in beta-normal eta-long form). For each of these rewriting mechanisms, there are results describing how to check confluence and termination.

Regarding confluence, these results describe which *critical pairs* must be computed in order to check the confluence property of the rewriting relation, assuming some termination property. In [17], we describe a general abstract result which can then be instantiated to all of the previous cases, and removes the assumptions above for higher-order rewriting. This is done via two novel notions: abstract positional rewriting allows us to capture the notion of critical peak without having to talk about a specific term structure; abstract normal rewriting with a triple (R, S, E) allows us to capture all different forms of rewriting: $S = E = \varnothing$ for plain rewriting; $S = \varnothing$ for rewriting modulo; E is alpha-conversion for higher-order rewriting, while the set of simplifiers S is made of beta-reduction and eta-expansion, R being the set of user-defined rules. Of course, there are other applications of normal rewriting described in the paper: for first-order computations, but also for higher-order computations at higher types, or using eta-reduction instead of eta-expansion, therefore solving a long-standing open problem.

Regarding termination, these results are very preliminary. In a recent paper submitted to ACM Transactions on Computational Logics, we extend the termination proof methods for higher-order computations based on plain pattern matching to higher-order rewriting systems based on higher-order pattern matching. We accomodate, for the one hand, with a weakly polymorphic, algebraic extension of Church's simply typed λ-calculus, and on the other hand, with any use of eta, as a reduction, as an expansion or as an equation. User's rules may be of any type in this type system, either a base, functional, or polymorphic type. Our techniques fit well with higher-order reduction orderings, such as the computability path ordering, but can also be used by other techniques, such as higher-order dependency pairs. All examples of normal higher-order rewrite rules that can be found in the litterature can be treated by our techniques, even those for which termination is by no means obvious to the expert.

6.4. Decreasing Diagrams

Based on the so-called Newman's lemma, the method for checking confluence introduced in the former paragraph applies to terminating computations. A completely different technique based on the so-called Hindley-Rosen's lemma applies when computation do not terminate, and is at the basis of Tait's confluence proof for the pure lambda-calculus. In recent papers, van Oostrom succeeded to capture both within a single framework thanks to the notion of decreasing diagram of a labelled abstract relation [76], see also [11] for an improved proof. Decreasing diagrams are specific convertibility proofs for local peaks, which labels are smaller in some sense than those of the local peak they aim at replacing. Any convertibility proof can then be converted into a confluence proof by recursively replacing its local peaks by their associated decreasing diagrams. Using a subtle characterization of confluence for arbitrary (possibly non-terminating) relations by cofinal derivations due to Klop [11], van Oostrom showed that any confluent relation which convertibility classes are countable, can be labelled in a way that makes it a labelled relation satisfying the decreasing diagram condition.

In [15], we first give a new, simple proof of van Oostrom's initial result based on a subtle well-founded order on conversions, and generalize it to rewriting modulo by using *strongly coherent cliffs* as an analog of decreasing diagrams for peaks. We then extend Klop's cofinal derivations to *cofinal streams*, and prove again a completeness result under the strong coherence assumption. Finally, we derive from these results a new, compact proof of Toyama's theorem that confluence is a modular property of rewriting systems built on disjoint vocabularies, and extend it to rewriting modulo when strong coherence is satisfied.

We are now trying to get rid of the strong coherence assumption by introducing a weaker analog of decreasing diagrams, *decreasing cliffs*. A preliminary result was presented early november at the Japanese Term Rewriting Workshop in Sendai.

This line of work is very promising. We expect it will eventually lead to the solution of an old open problem, the characterization of a class of non-left linear, non-terminating rewrite systems for which confluence is decidable by means of (parallel) critical pairs. We believe that the implementation of such a result would be impact the way confluence proofs are carried out, including in type theory.

6.5. Higher-order Reduction Orderings

Since HORPO , several higher-order reduction orderings have been described, based on either Dershowitz's RPO , Blanqui-Jouannaud-Okada's Computational Closure , and Arts and Giesel' dependency pairs . Our work continues in three different directions:

- CPO is an order for simply typed lambda-terms that allows to show strong normalization of beta-reduction even in presence of higher-order rewrite rules provided these rules decrease in the ordering [32]. It is currently the only automated mechanism that achieves non-trivial computations by turning Girard's computability predicates method into a usable tool. It has been shown that CPO can handle weakly polymorphic type disciplines, as well as inductive types. Recently, we have shown that CPO scales up to dependently typed calculi as LF. We are currently writing a paper describing CPO and its extensions to calculi with inductive and dependent types which should be submitted to a journal by the end of the year.
- Frédéric Blanqui defended his "Habilitation à diriger des recherches" at the University Denis Diderot (Paris 7) on July 13. In [13], he gives a synthetic view on how the notion of computability closure can be used to prove the termination of various kinds of rewrite relations (class rewriting or rewriting with matching modulo), and how it relates with other notions (dependency pairs, semantic labeling, and HORPO, the predecessor of CPO.
- Frédéric Blanqui has developed an automated termination prover called HOT based on the above work on the computability closure and his former work on size annotations [31]. For its first participation, HOT won the international competition on termination in the category "higher-order rewriting union beta".

6.6. Certification of Termination Proofs

Frédéric Blanqui and Kim Quyen Ly continued to work on the development of a new version of Rainbow based on Coq extraction mechanism [59]. We developed a tool generating from an XSD file, Coq and OCaml data structures representing the XML types defined the XSD file, and OCaml parsing functions for generating such data structures from an XML file. The main difficulty was to topologically reorder the XSD type definitions in order to get simple and well defined Coq data structures. We also defined and proved in Coq a function for checking the correctness of termination certificates based on the DP transformation [26]. The main difficulty was to manage the evolution of the arity function along the transformation. Indeed, to simplify the translation of CPF elements into the data structures used in CoLoR [30], we decided to use a fixed but infinite set of symbols [69]. However the arity function need to be updated along the transformations applied to the system. These results are presented in [20].

6.7. Certification of Moca

Frédéric Blanqui has formalized in Coq and proved the correctness and completeness of the construction functions generated by Moca for the theory of groups [29]. The first difficulty is to represent the Moca functions themselves in a faithful way because, in Coq, there is no "when" clauses and "match" constructions are expanded into elementary "case" constructions with no tuple patterns and patterns of depth one only. In addition, Coq termination checker only accepts functions with exactly one structurally decreasing argument, which is generally not the case of Moca functions. The second difficulty is the completeness proof: it requires the use of intermediate data structures for reasoning on normal forms. During his internship, Rémi Nollet (L3, ENS Lyon) improved the representation of OCaml functions by using inductive predicates, and extended the correctness proof to commutative groups.

6.8. First steps towards the certification of an ARM simulator

The simulation of Systems-on-Chip (SoC) is nowadays a hot topic because, beyond providing many debugging facilities, it allows the development of dedicated software before the hardware is available. Low-consumption

CPUs such as ARM play a central role in SoC. However, the effectiveness of simulation depends on the faithfulness of the simulator. To this effect, we started to prove significant parts of such a simulator, SimSoC. Basically, on one hand, we develop a Coq formal model of the ARM architecture while on the other hand, we consider a version of the simulator including components written in Compcert-C [58]. Then we prove that the simulation of ARM operations, according to Compcert-C formal semantics, conforms to the expected formal model of ARM. Size issues are partly dealt with using automatic generation of significant parts of the Coq model and of SimSoC from the official textual definition of ARM [3]. A second step was achieved in [12], with the proof a significant instruction (ADC, Add with Carry). A crucial technical issue was then raised: facilitating reasoning by inversion on the rules defined in Compcert-C. Hundreds such steps are required for a single instruction, and each of them generates a dozen of new names. Relying on Coq tactic inversion results in unmanageable scripts, very fragile and difficult to maintain. In 2012 we dealt with this issue by designing our own inversion mechanism, allowing us to improve automation of the proof, while keeping enough command so that interactive steps refer to controlled names. It was then possible to get a much shorter proof on ADC and to prove at least one instruction in each category of the ARM instruction set.

6.9. Certified implementation of BIP

BIP (*Behavior, Interaction, Priority*) is a component-based language designed at VERIMAG for modeling and programming complex embedded systems [27]. A BIP model is essentially a set of atomic components described with explicit states and transitions, composed together in a hierarchical way. The main original feature of BIP lies in a very rich notion of *connector* for defining interactions between components [33]. An efficient implementation of BIP in C++ is already available at VERIMAG.

Building on our previous experience on SimSoC, we started to work on a certified implementation of BIP. Our long term objective is to propose a certified compilation chain from BIP models to embedded code, through a first translation from BIP to Compcert-C.

In 2012 we focused on a simple subset of BIP Currently, we have a first definition of a formal semantics of this subset in Coq, in two versions: an relational version, inspired by a rule-based operational semantics, and a functional version, which specifies a possible implementation of the relational version (in particular, it includes a scheduler). We also produce a Compcert-C code which is expected to behave exactly like the functional semantics, and we started to state and prove corresponding statements on very simple BIP models.

6.10. Formal model and proofs for Netlog protocols

Netlog is a language designed and implemented in the Netquest project for describing protocols. Netlog has a precise semantics, provides a high level of abstraction thanks to its Datalog flavor and benefits from an efficient implementation. This makes it a very interesting target language for proofs of protocols.

Jean-François Monin, Stéphane Grumbach (formerly LIAMA/Netquest) and Yuxin Deng (Jiaotong University, Shanghai) designed a formal model of Netlog in Coq, where the two possible semantics are derived from common basic blocks. In a fully certified framework, a formal proof of the Netlog engine (running on each node) would be required. We don't attack this part at the moment: we assume that the implementation respects the general properties stated in our model and focus on the issues raised by the distributed model of computation provided by Netlog. This framework could be applied to an algorithm constructing a Breadth-First Search Spanning Tree (BFS) in a distributed system [45].

In 2011, Jean-François Monin and Meixian Chen (Jiaotong Shanghai) generalized the model in order to take the removal of datalog facts into account, and used the improved framework to Prim's algorithm. In 2012, this work was slightly improved and published in [16].

6.11. Formalisation of security APIs for mobile phones

This work is in cooperation with Nokia Beijing, who was interested by the application of verification technologies to mobile phones. We decided to focus on security APIs, considering that mobile devices are commonly used by end-users to store their personnal data (e.g., passwords), while running all sort of downloaded applications at the same time.

For 2012, we (including Nokia) agreed to consider devices under Android, though Nokia switched to windows, in order to circumvent copyright issues.

Three models and corresponding sets of APIs for password storage applications on Android were developed. Each model fixes some bugs of the previous one and introduces a new feature. We consider the third model is enough for the basic function and well built to be safe. Then, a full Coq proof of the third model was developed as well as its corresponding API's security property. A suitable abstraction of the application on the phone within its environment is described as a state transition system. Then we proved by induction that the expected secrets actually remain secret at any reachable state.

6.12. Trace Analysis

Simulation sessions produce huge trace files, sometimes now in hundreds of gigabytes, that are hard to analyze with a quick response time. This comes down to two sub-problems:

- The trace file size. Trace files are huge because they include lots of information. But when looking for a specific problem, one does not need all of this information. To search one given defect, one may ignore a large amount of the data in the trace file. One would like the trace file to contain only relevant information to the concerned problem.
- The expressive power of the language to analyze the trace, and its usability. If the language is limited to expression search, it is easy to use but hard to construct sophisticated formulas. If the language used is Linear Temporal Logic (LTL), there is a very high expressive power but many engineers are unable to write a LTL formula and to maintain it over time.

We have started to build a trace analysis tool. It includes a language which allows expression of time-related formulas as a subset of LTL, but is simple to formulate expressions. When this language is compiled, the compiler generates two outputs:

- a filter script that will help reduce the size of the trace file.
- a program that analyzes such trace files to find whether the formula is satisfied.

When compiling one trace language input file, it generates a filter script. The filter script is a set of data descriptors. It describes which events from the simulator must be traced and which should be ignored. Then during the simulation, the filter is loaded and only the required output is generated.

We have started to design a trace language and a compiler, and extended the SimSoC simulator to support generation of trace files with a filter. A first version of the trace language compiler has been implemented in OCAML, which generates OCAML programs for trace analysis. In the current version under development, the filters are not yet parallelized with simulation.

GALLIUM Project-Team

6. New Results

6.1. Language design and type systems

6.1.1. The Mezzo programming language

Participants: Jonathan Protzenko, François Pottier.

In the past ten years, the type systems community and the separation logic community, among others, have developed highly expressive formalisms for describing ownership policies and controlling side effects in imperative programming languages. In spite of this extensive knowledge, it remains very difficult to come up with a programming language design that is simple, effective (it actually controls side effects!) and expressive (it does not force programmers to alter the design of their data structures and algorithms).

The Mezzo programming language, formerly known as HaMLet, aims to bring new answers to these questions.

We have come up with a solid design for the programming language: many features of the language have been reworked or consolidated this year, and we believe we strike a good balance between expressiveness and complexity. We wrote several flagship examples that illustrate the gains offered by Mezzo, as well as two (yet unpublished) papers discussing the design of the language. Jonathan Protzenko implemented a prototype type-checker; although it is not perfect yet, several non-trivial examples are successfully type-checked.

The current state of the Mezzo programming language is best described in [40]; a former version of this document can be found as [39].

François Pottier wrote a formal definition of (a slightly lower-level variant of) Mezzo, and proved that Mezzo is type-safe: that is, well-typed programs cannot crash (but they can stop abruptly if a run-time check fails). The proof, which is about 15,000 lines, has been machine-checked using Coq. A paper that describes this work is in preparation.

This work was facilitated by Pottier's experience with a similar previous proof. In particular, out of the above 15,000 lines, about 2,000 lines correspond to a re-usable library for working with de Bruijn indices, and about 3,000 lines correspond to a re-usable formalisation of "monotonic separation algebras", which help reason about resources (memory, time, knowledge, ...) and how they evolve over time. These libraries have not yet been fully documented and released; this might be done in the future.

6.1.2. Coercion abstraction

Participants: Julien Cretin, Didier Rémy.

Expressive type systems often allow non trivial conversions between types, which may lead to complex, challenging, and sometimes ad hoc type systems. Such examples are the extension of System F with type equalities to model GADTs and type families of Haskell, or the extension of System F with explicit contracts. A useful technique to simplify the meta-theoretical studies of such systems is to make type conversions explicit as "coercions" inside terms.

Following a general approach to coercions based on System F, we introduced a language F-iota with abstraction over coercions and where all type transformations are represented as coercions. The main difficulty is dealing with coercion abstraction, as abstract coercions whose types are uninhabited cannot be erased at run-time. We proposed a restriction, called parametric F-iota, that ensures erasability of all coercions by construction. This work was presented at the POPL conference in January [22].

We extended parametric F-iota with non-interleaved positive recursive types and with erasable isomorphisms. We generalized the presentation of the language viewing coercions as conversions between typings (pairs of a typing environment and a type) rather than between types. An extended version with full proofs will be submitted for journal publication.

We also studied a more liberal version of F-iota where coercion inhabitation is no more ensured by construction (which limits expressiveness), but instead by providing coercion witnesses in source terms. This extension requires pushing abstract coercions under redexes so that they do not block the reduction. As a consequence, coercions cannot be reified in System F, and we need a direct proof of termination of iota-reduction. We completed one such proof based on reducibility candidates.

6.1.3. Ambivalent types for principal type inference with GADTs

Participants: Jacques Garrigue [Nagoya University], Didier Rémy.

Type inference for Generalized Abstract Data Types (GADTs) is always a matter of compromise because it is inherently non monotone: assuming more specific types for GADTs may ensure more invariants, which in turn may result in more general types. Moreover, even when types of GADTs parameters are explicitly given, they introduce equalities between types, which makes them inter-convertible but with a limited scope. This may then creates an ambiguity when leaving the scope of the equation: which representative should be used for the equivalent forms? Ideally, one should use a type disjunction, but this is not allowed—for good reasons. Hence, to avoid arbitrary choices, these situations must be rejected, forcing the user to add more annotations to resolve ambiguities.

We proposed a new approach to type inference with GADTs. While some uses of equations are unavoidable and create real ambiguities, others are gratuitous and create artificial ambiguities, To distinguish between the two, we introduced *ambivalent types*: a way to trace types that have been obtained by an unavoidable use of an equation. We then redefined ambiguities so that only ambivalent types become ambiguous and should be rejected or resolved by a programmer annotation.

Interestingly, the solution is fully compatible with unification-based type inference algorithms used in ML dialects. The work was presented at the ML workshop [31] and implemented in the latest version 4.00 of OCaml.

6.1.4. GADTs and Subtyping

Participants: Gabriel Scherer, Didier Rémy.

Following the addition of GADTs to the OCaml language in version 4.00 released this year, we studied the theoretical underpinnings of variance subtyping for GADTs. The question is to decide which variances should be accepted for a GADT-style type declaration that includes type equality constraints in constructor types. This question exposes a new notion of decomposability and unexpected tensions in the design of a subtyping relation. Our formalization partially reuses earlier work by François Pottier and Vincent Simonet [54]. It was presented at the ML Workshop [33]. An extended version including full proofs is available as a technical report [38] and was submitted for presentation at a conference.

6.1.5. Singleton types for code inference

Participants: Gabriel Scherer, Didier Rémy.

Inspired by tangent aspects of the PhD work of Julien Cretin, we investigated the use of singleton types for code inference. If we can prove that a type contains, in a suitably restricted pure lambda-calculus, a unique inhabitant modulo program equivalence, the compiler can infer the code of this inhabitant. This opens the way to type-directed description of boilerplate code, through type inference of finer-grained type annotations. The preliminary results seem encouraging, both on the theoretical side (identifying general situations for type-directed programming) and the practical side (mining existing OCaml code for usage situations).

6.1.6. Programming with names and binders

Participants: Nicolas Pouillard, François Pottier.

Following Nicolas Pouillard's Ph.D. defense in January 2012 [11], Nicolas Pouillard and François Pottier produced a unified presentation of Pouillard's approach to programming with abstract syntax, in the form of a paper that was published in the Journal of Functional Programming [16].

6.1.7. A type-and-capability calculus with hidden state

Participant: François Pottier.

During the year 2010, François Pottier developed a machine-checked proof of an expressive type-and-capability system, which can be used to type-check and prove properties of imperative ML programs. The proof is carried out in Coq and takes up roughly 20,000 lines of code. In the first half of 2011, François Pottier wrote a paper that describes the system and its proof in detail. This paper was published, after a revision, in 2012 [15].

6.2. Formal verification of compilers and static analyses

6.2.1. The CompCert verified C compiler

Participants: Xavier Leroy, Sandrine Blazy [project-team Celtique], Jacques-Henri Jourdan, Valentin Robert.

In the context of our work on compiler verification (see section 3.3.1), since 2005 we have been developing and formally verifying a moderately-optimizing compiler for a large subset of the C programming language, generating assembly code for the PowerPC, ARM, and x86 architectures [5]. This compiler comprises a back-end part, translating the Cminor intermediate language to PowerPC assembly and reusable for source languages other than C [4], and a front-end translating the CompCert C subset of C to Cminor. The compiler is mostly written within the specification language of the Coq proof assistant, from which Coq's extraction facility generates executable Caml code. The compiler comes with a 50000-line, machine-checked Coq proof of semantic preservation establishing that the generated assembly code executes exactly as prescribed by the semantics of the source C program.

The two major novelties of CompCert this year are described separately: verification of floating-point arithmetic (section 6.2.2) and a posteriori validation of assembly and linking (section 6.2.3). Other improvements to CompCert include:

- The meaning of "volatile" memory accesses is now fully specified in the semantics of the CompCert C source language. Their translation to built-in function invocations, previously part of the unverified pre-front-end part of CompCert, is now proved correct.
- CompCert C now natively supports assignment between composite types (structs or unions), passing composite types by value as function parameters, and other instances of using composites as r-values, with the exception of returning composites by value from a function.
- A new pass was added to the compiler to perform inlining of functions. Its correctness proof raised interesting challenges to properly relate the (widely different) call stacks of the program before and after inlining.
- The constant propagation optimization is now able to propagate the initial values of global variables declared `const`.
- The common subexpression elimination (CSE) optimization was improved so as to eliminate more redundant memory loads.

Two versions of the CompCert development were publicly released, integrating these improvements: versions 1.10 in March and 1.11 in July. We also wrote a 50-page user's manual [37] and a technical report on the CompCert memory model [35].

In parallel, we continued our collaboration with Jean Souyris, Ricardo Bedin França and Denis Favre-Felix at Airbus. They are conducting an experimental evaluation of CompCert's usability for avionics software, and studying the regulatory issues (DO-178 certification) surrounding the potential use of CompCert in this context. Preliminary results were presented at the 2012 Embedded Real-Time Software and Systems conference (ERTS'12) [29].

6.2.2. Formalization of floating-point arithmetic in Compcert

Participants: Sylvie Boldo [project-team Toccata], Jacques-Henri Jourdan, Xavier Leroy, Guillaume Melquiond [project-team Toccata].

The aim of this research theme was to formalize the semantics and compilation of floating-point arithmetic in the CompCert compiler. Prior to this work, floating-point arithmetic was axiomatized in the Coq proof of CompCert, then mapped to OCaml's floating-point operations during extraction. This approach was prone to errors and fails to formally guarantee conformance to the IEEE-754 standard for floating-point arithmetic.

To remedy this situation, Jacques-Henri Jourdan replaced this axiomatization by a fully-formal Coq development, building on the Coq formalization of IEEE-754 arithmetic provided by the Flocq library. Sylvie Boldo and Guillaume Melquiond, authors of Flocq, adapted their library to the needs of this development. The new formalization of floating-point arithmetic is used throughout CompCert: to give semantics to FP computations in the source, intermediate and target (assembly) languages; to perform correct compile-time FP evaluations during constant propagation; to prove the correctness of code generation scheme for conversions between integers and FP numbers; and to parse FP literals with correct rounding.

A paper describing this work is accepted for presentation at the forthcoming ARITH 2013 conference [20].

6.2.3. Validation of assembly and linking

Participants: Valentin Robert, Xavier Leroy.

Valentin Robert designed and implemented a validation tool for the assembly and linking phases of the CompCert C compiler. These passes are not formally verified and call into off-the-shelf assemblers and linkers. The `cchecklink` tool of Valentin Robert improves the confidence that end-users can have in these passes by validating *a posteriori* their operation. The tool takes as inputs the PowerPC/ELF executable produced by the linker, as well as the abstract syntax trees for assembly files produced by the formally-verified part of CompCert. It then proceeds to establish a correspondence between the two sets of inputs, via a thorough structural analysis on the ELF executable, light disassembling of the machine code, expansion of CompCert's macro-asm instructions, and propagation of constraints over symbolic names. The tool produces detailed diagnostics if any discrepancies are found.

6.2.4. Improving CompCert's reusability for verification tools

Participants: Xavier Leroy, Jacques-Henri Jourdan, Andrew Appel [Princeton University], Sandrine Blazy [project-team Celtique], David Pichardie [project-team Celtique].

Several ongoing projects focus on proving the soundness of verification tools that reuse parts of the CompCert development, namely some of the intermediate languages, their formal semantics, and the CompCert passes that produce these intermediate forms. This is the case for the Verasco ANR project, which focuses on the proof of a static analyzer based on abstract interpretation, and for the Verified Software Toolchain (VST) project, led by Andrew Appel at Princeton University, which develops a concurrent separation logic embedded in Coq. However, the CompCert intermediate languages, currently designed to fit the needs of a compiler, are not perfectly suited to static analysis and deductive verification.

To improve the reusability of CompCert's Clight language in the Verasco and VST projects, Xavier Leroy is currently revising the CompCert C front-end passes so that function-local C variables whose address is never taken are pulled out of memory and replaced by nonadressable temporary variables. The resulting Clight intermediate form is much easier to analyze or prove correct, as temporary variables cannot suffer from aliasing problems.

Likewise, Sandrine Blazy, Jacques-Henri Jourdan, Xavier Leroy and David Pichardie designed a variant of CompCert's RTL intermediate language, called CFG. Like RTL, CFG represents the flow of control by a graph; unlike RTL, CFG is independent of the target processor, and supports complex expressions instead of 3-address code. These features of CFG make it a better target for static analysis, both non-relational (e.g. David Pichardie's certified interval analysis) and relational. Jacques-Henri Jourdan implemented and proved correct a compilation pass that produces CFG code from the Cminor intermediate language of CompCert.

6.2.5. Formal verification of hardware synthesis

Participants: Thomas Braibant, Adam Chlipala [MIT].

Verification of hardware designs has been thoroughly investigated, and yet, obtaining provably correct hardware of significant complexity is usually considered challenging and time-consuming. Hardware synthesis aims to raise the level of description of circuits, reducing the effort necessary to produce them.

This yields two opportunities for formal verification: a first option is to verify (part of) the hardware compiler; a second option is to study to what extent these higher-level design are amenable to formal proof.

During a visit at MIT, Thomas Braibant worked on the implementation and proof of correctness of a prototype hardware compiler in Coq, under Adam Chlipala's supervision. This compiler produces descriptions of circuits in RTL style from a high-level description language inspired by BlueSpec. After joining Gallium, Thomas Braibant continued working part time on this subject, finishing the proof of this compiler, and implementing a few hardware designs of mild complexity. This work was presented at the 2012 Coq Workshop [30] and will be submitted to a conference in 2013.

6.2.6. A formally-verified alias analysis

Participants: Valentin Robert, Xavier Leroy.

Valentin Robert improved the verified static analysis for pointers and non-aliasing that he initiated in 2011 during his Master's internship supervised by Xavier Leroy. This alias analysis is intraprocedural and flow-sensitive, and follows the "points-to" approach of Andersen [41]. An originality of this alias analysis is that it is conducted over the RTL intermediate language of the CompCert compiler: since RTL is essentially untyped, the traditional approaches to field sensitivity do not apply, and are replaced by a simple but effective tracking of the numerical offsets of pointers with respect to their base memory blocks. The soundness of this alias analysis is proved against the operational semantics of RTL using the Coq proof assistant and techniques inspired from abstract interpretation. A paper describing the analysis and its soundness proof was presented at the CPP 2012 conference [26].

6.3. The OCaml language and system

6.3.1. The OCaml system

Participants: Xavier Clerc [team SED], Damien Doligez, Alain Frisch [Lexifi SAS], Jacques Garrigue [University of Nagoya], Fabrice Le Fessant [Inria Saclay and OCamlPro start-up company], Jacques Le Normand [Lexifi SAS], Xavier Leroy.

This year, we released versions 4.00.0 and 4.00.1 of the OCaml system. Version 4.00.0 (released in July) is a major release that fixes about 150 reported bugs and 4 unreported bugs, and introduces 57 new features suggested by users. Version 4.00.1 (released in October) is a bug-fix release that fixes 3 major and 20 minor bugs. Damien Doligez acted as release manager for both versions.

The major innovation in OCaml 4.00 is support for generalized algebraic datatypes (GADTs). These non-uniform datatype definitions enable programmers to express some invariants over data structures, and the OCaml type-checker to enforce these invariants. They also support interesting ways of reflecting types into run-time values. GADTs are found in proof assistants such as Coq and in functional languages such as Agda and Haskell. Their integration in OCaml raised delicate issues of partial type inference and principality of inferred types, to which Jacques Garrigue and Jacques Le Normand provided original solutions [45].

Other features of this release include:

- Lightweight notations to facilitate the use of first-class modules.
- Better reporting of type errors.
- Improvements in native-code generation.
- Performance and security improvements in the hashing primitive and hash tables.
- New warnings for unused code (variables, record fields, etc.)
- A new back-end for the ARM architecture.

6.3.2. Namespaces for OCaml

Participants: Gabriel Scherer, Didier Rémy, Fabrice Le Fessant [Inria Saclay].

As part of an ongoing discussion among members of the OCaml Consortium, we investigated the formal aspects of "namespaces" and their putative status in the OCaml language. Namespaces aim at providing OCaml programmers with efficient ways to manage and structure the names of compilation units, in contrast with the flat, global space of compilation units provided today in OCaml. This formalization provides scientific support to ongoing design and engineering discussions. It was presented at the December 2011 IFIP 2.8 working group on functional programming, and at the December 2012 meeting of the OCaml Consortium.

6.4. Software specification and verification

6.4.1. Tools for TLA+

Participants: Damien Doligez, Leslie Lamport [Microsoft Research], Stephan Merz [EPI VeriDis], Tomer Libal [Microsoft Research-Inria Joint Centre], Hernán Vanzetto [Microsoft Research-Inria Joint Centre].

Damien Doligez is head of the "Tools for Proofs" team in the Microsoft-Inria Joint Centre. The aim of this team is to extend the TLA+ language with a formal language for hierarchical proofs, formalizing the ideas in [47], and to build tools for writing TLA+ specifications and mechanically checking the corresponding formal proofs.

This year, the TLA+ project released two new versions (in January and in November) of the TLA+ tools: the GUI-based TLA Toolbox and the TLA+ Proof System, an environment for writing and checking TLA+ proofs. This environment is described in a paper presented at the 2012 symposium on Formal Methods [21]. The January release (version 1.0 of TLAPS and 1.4.1 of Toolbox) added support for back-ends based on SMT provers (CVC3, Z3, Yices, VeriT), which dramatically extends the range of proof obligations that the system can discharge automatically. The November release includes many bug-fixes and performance improvements.

We have also improved the theoretical design of the proof language with respect to temporal properties. This design will be implemented in TLAPS in the near future.

Web site: http://tlaplus.net/

6.4.2. The Zenon automatic theorem prover

Participants: Damien Doligez, David Delahaye [CNAM], Mélanie Jacquel [CNAM].

Damien Doligez continued the development of Zenon, a tableau-based prover for first-order logic with equality and theory-specific extensions. Version 0.7.1 of Zenon was released in May.

David Delahaye and Mélanie Jacquel designed and implemented (with some help from Damien Doligez) an extension of Zenon called SuperZenon, based on the Superdeduction framework of Brauner, Houtmann, and Kirchner [43].

Both Zenon and SuperZenon entered the CASC theorem-proving contest, where, as expected, SuperZenon did much better than Zenon.

6.4.3. Hybrid contract checking via symbolic simplification

Participant: Na Xu.

Program errors are hard to detect or prove absent. Allowing programmers to write formal and precise specifications, especially in the form of contracts, is one popular approach to program verification and error discovery. Na Xu formalizes and implements a hybrid contract checker for a pure subset of OCaml. The key technique we use is symbolic simplification, which makes integrating static and dynamic contract checking easy and effective. This technique statically verifies that a function satisfies its contract or blames the function violating the contract. When a contract satisfaction is undecidable, it leaves residual code for dynamic contract checking.

A paper describing this result is published in the proceeding of the PEPM'2012 conference [27]. An extended version of this paper will appear in the journal Higher-Order and Symbolic Computation. Na Xu implemented this approach in a prototype based on the OCaml 3.12.1 compiler and experimented with nontrivial examples such as sorting algorithms and balancing AVL trees (see http://gallium.inria.fr/~naxu/research/hcc.html).

6.4.4. Probabilistic contracts for component-based design

Participants: Na Xu, Gregor Goessler [project-team POPART], Alain Girault [project-team POPART].

We define a framework of probabilistic contracts for constructing component-based embedded systems, based on the formalism of discrete-time Interactive Markov Chains. A contract specifies the assumptions a component makes on its context and the guarantees it provides. Probabilistic transitions represent allowed uncertainty in the component behavior, for instance, to model internal choice or reliability. Action transitions are used to model non-deterministic behavior and communication between components. An interaction model specifies how components interact with each other.

We provide the ingredients for a component-based design flow, including (1) contract satisfaction and refinement, (2) parallel composition of contracts over disjoint, interacting components, and (3) conjunction of contracts describing different requirements over the same component. Compositional design is enabled by congruence of refinement. A paper describing the details of this result is published in the journal Formal Methods in System Design [14].

MARELLE Project-Team

5. New Results

5.1. Coq and SMT provers

Participants: Michaël Armand, Benjamin Grégoire, Laurent Théry.

Continuing the work of previous years, we added an extra theory to the interface between Coq and Satisfiability Modulo Theory (SMT) provers: instantiation. It is the last really needed piece to make our tactic based on SMT provers really useful to Coq users. Part of the work was to make the proof work on statements existing in the Propositional type instead of the boolean type. This requires a change in the correctness proof.

5.2. Formal proofs on Pi

Participant: Yves Bertot.

We studied the chain of definitions and proofs necessary to show that

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} \cdots$$

and removed the axiom that was left on this topic in Coq's standard library. This part re-used a past contribution of Guillaume Allais during an internship from Ecole Normale Supérieure de Lyon. We then added a study of Machin's formula to compute decimals of π.

5.3. Formal proofs on linear algebra

Participants: Guillaume Cano, Maxime Dénès, Anders Mörtberg [University of Chalmers, Sweden], Vincent Siles [University of Chalmers, Sweden], Yves Bertot.

This year we completed a work on matrix canonical forms, providing formal proofs for the following results:

- Smith normal forms of matrices on principal ideal domains are unique,
- Every matrix on a field is similar to its Frobenius normal form
- Every matrix on an algebraically closed field is similar to its Jordan normal form

We also studied techniques to combine high-level mathematical descriptions and proofs of algorithms with executable implementations. This work led to a publication at ITP'12 [10]. We are still working on extending this work to rational numbers and real algebraic numbers.

We then worked on tools to automate proofs. In the ring tactic, all elements considered must belong to the same type. We worked on extending this tactic to dependent families of types, like the type of matrices where each dimension gives rise to a different type in the family and multiplications typically concern matrices of different types, while remaining associative.

5.4. Formal proof of the Feit-Thompson theorem

Participants: Laurence Rideau, Laurent Théry.

The Feit-Thompson theorem, established in the beginning of the 1960s, states that every odd-order finite group is solvable. The proof of this result was initially published in an article with around 250 pages. This proof was cleaned by a team of mathematicians and re-published in the form of two books, totaling approximately the same number of pages. But these books also rested on some general knowledge about groups and various areas of algebras.

All this knowledge is now formally described in the Mathematical Components library. The proof of the theorem has been completed in September 2012. The team that achieved this result includes members of the Marelle project-team, along with members of the Typical project-team at Inria Saclay-Ile de France, members of the Microsoft Research Cambridge laboratory, and guests from other institutions.

This year, the members of the Marelle team concentrated on the following topics:

- General character theory: chapters 5 and 6 of the book by Isaacs,
- Character theory for the odd order theorem: chapters 1 to 4 of the book of the book by Peterfalvi.

More information at http://www.msr-inria.inria.fr/Projects/math-components/feit-thompson.

5.5. Native execution for the Coq system

Participants: Maxime Dénès, Benjamin Grégoire, Yves Bertot.

We have continued our work on the native execution of dependently typed terms, aiming at the integration of this work in the main branch of the Coq system.

5.6. Provably correct approximations of elementary functions

Participants: Erik Martin-Dorel, Laurence Rideau, Laurent Théry.

The elementary functions are general purpose mathematical functions that are often implemented in the hardware of modern micro-processors: exponential and trigonometric functions, and inverse functions like arctan or square-root. We participate in a nationally funded project (ANR-TaMaDi) where precise approximations of these functions and their combinations must be computed. A first approach is to use Taylor models. We implemented such an approach and proved its correctness in the Coq system. This led to the publication [9].

We are now working on applying Bernstein polynomials to the problem of approximating transcendental functions.

5.7. Geometric algebras

Participant: Laurent Théry.

We translated our library to the ssreflect setting and provided a very concise certified implementation of geometric algebras based on binary trees.

5.8. Bourbaki in Coq

Participant: José Grimm.

In previous years, we developed a formal library describing the part of the Bourbaki books on set theory, cardinals, and ordinals. The whole development now runs under Coq 8.4, ssreflect 1.4. The main contribution this year is the study of some families of numbers (Stirling numbers of the second kind, Euler numbers, Bell numbers), and their relations to cardinalities (number of partitions of a set, number of partition with p parts, number of surjections $I_n \to I_p$). We have some explicit formulas for $\sum_{i<n} i^k$ as sums of binomial coefficients.

5.9. Reasoning on polynomial expressions

Participants: José Grimm, Julianna Zsido, Yves Bertot.

Continuing previous work by Bertot, we showed that if p is a polynomial on any ordered ring, that has n positive roots, the list of its coefficients has at least n sign changes. If there is exactly one sign change, and the ring is an Archimedian field, there is a number a such that the polynomial is negative on $[0, a]$ and strictly increasing after a; thus it has at most one positive root, and there is a Cauchy sequence x_i such that $p(x_i) < 0$ but $p(x_i + c/2^n) > 0$.

The publication by Bertot, Mahboubi, and Guilhot in 2011 on Bernstein polynomials describes a procedure that works only for polynomials with simple roots. We added the proofs that describe how to obtain such polynomials, starting from arbitrary ones. In other words, we proved the following statement: *for every polynomial p, p divided by the greatest common divisor of p and its derivative has the same roots as p and all the roots are simple.*

We started working on a proof that the dichotomy process based on Bernstein polynomials is bound to terminate, concentrating on a theorem known as *the theorem of three circles*.

5.10. Higher-Order Abstract Syntax

Participant: Julianna Zsido.

With Martin Hyland from the University of Cambridge, we worked on an approach to reconcile the points of view of Fiore, Plotkin, and Turi on the one hand and Hirschowitz and Maggesi on the other hand. This approach relies on a large monad that abstracts over the two approaches.

5.11. Proofs in cryptography

Participants: Gilles Barthe [IMDEA Software Institute], Juan Manuel Crespo [IMDEA Software Institute], Benjamin Grégoire, Sylvain Heraud [Prove&Run], César Kunz [IMDEA Software Institute], Yassine Lakhnech [University of Grenoble], Pierre-Yves Strub [IMDEA Software Institute], Santiago Zanella Béguelin [IMDEA Software Institute].

We are continuing our work on providing a user-friendly tool for cryptographers who want to develop formal proofs of correctness, based on Certicrypt and SMT provers. There were invited talks at ITP, CPP, MPP, SAS, and JFLA. There was also an article in ERCIM news, whose contents is more oriented towards the open public. See also the web page http://easycrypt.gforge.inria.fr/.

As an illustrative example, we proposed a machine-checked proof of a construction of a hash function based on elliptic curves, where the correctness proof uses the Random Oracle Model. The proof is based on an extension of CertiCrypt for reasoning about approximate forms of observational equivalence and uses mathematical results from group theory and elliptic curves.

Thanks to our language-based approach to describing cryptographic constructions and our automatic approach to proving them correct, we can now explore systematically the space of possible designs. Using this approach, we have been able to explore over 1.3 million schemes, including more than 100 variants of OAEP studied in the literature and to prove the correctness of 250,000 schemes for one kind of model and 17,000 for another kind.

MEXICO Project-Team

6. New Results

6.1. Avoiding shared clocks in networks of timed automata

Networks of timed automata (NTA) are widely used to model distributed real-time systems. Quite often in the literature, the automata are allowed to share clocks, i.e. the transitions of one automaton may be guarded by a condition on the value of clocks reset by another automaton. This is a problem when one considers implementing such model in a distributed architecture, since reading clocks a priori requires communications which are not explicitly described in the model.

In [58], we focus on the following question: given an NTA $A_1 \| A_2$ where A_2 reads some clocks reset by A_1, does there exist an NTA $A'_1 \| A'_2$ without shared clocks with the same behavior as the initial NTA? For this, we allow the automata to exchange information during synchronizations only, in particular by copying the value of their neighbor's clocks.

We discuss a formalization of the problem and give a criterion using the notion of contextual timed transition system, which represents the behavior of A_2 when in parallel with A_1. Finally, we effectively build $A'_1 \| A'_2$ when it exists.

6.2. Model checking languages over infinite alphabets

In [61], we consider data words, i.e, strings where each position carries both a label from a finite alphabet and some values from an infinite domain. The latter can be used to represent an unbounded number of process identifiers so that data words are suitable to model the behavior of a concurrent program with dynamic process creation. A variety of formalisms, including logic and automata, have been studied in the literature to specify sets of data words in the context of verification. Most of them focus on the satisfiability problem of very restricted logics, as the general problem is undecidable.

Here, we consider the dual approach of restricting the domain of data words instead of pruning the logic. This allows us to tackle the model-checking problem with respect to monadic second-order (MSO) properties. As model checking is undecidable for nearly all known automata models (including the model presented in the first part of the talk), we introduce data pushdown automata (DPA). DPA come with multiple pushdown stacks (where the access to stacks is bounded by a number of phase switches) and are enriched with parameters that can be instantiated with data values. DPA can model interesting protocols like a leader election protocol with an unknown number of processes. While satisfiability for MSO logic is undecidable (even for weaker fragments such as first-order logic), we show that one can decide if all words generated by a DPA satisfy a given formula from the full MSO logic.

6.3. Construction of Hanf sentences

A classical result by Hanf from the 60s states that first-order formulas over structures of bounded degree are equivalent to boolean combinations of statements of the form: "pattern P occurs at least n times". Hanf's theorem has many model-theoretic applications, in particular in automata theory and database query answering.

However, until recently, no elementary construction was known. In [49], we present the first algorithm that computes a Hanf normal in elementary time. More precisely, our algorithm is triply exponential, which we also show to be optimal.

6.4. A probabilistic Kleene theorem

In [63], we establish a Kleene theorem for (Rabin) probabilistic automata over finite words. Probabilistic automata generalize deterministic finite automata and assign to a word an acceptance probability. For convenient specification of probabilistic queries, we provide probabilistic expressions with probabilistic choice, guarded choice, concatenation, and a star operator. Our expressions are closer to language-theoretic operations than previous calculi for probabilistic systems, which were rather motivated by system modeling than query answering. We prove that probabilistic expressions and probabilistic automata are expressively equivalent. Our result extends to two-way probabilistic automata with pebbles and corresponding expressions.

6.5. A temporal logic for frequency properties

In, [62], we introduce fLTL, a quantitative extension of the widely used specification language LTL that allows us to express relative frequencies by a generalization of temporal operators. This facilitates the specification of requirements such as the deadlines in a real-time system must be met in at least 95% of all cases. For our novel logic, we establish an undecidability result regarding the satisfiability problem but identify a decidable fragment which strictly increases the expressiveness of LTL by allowing, e.g., to express non-context-free properties.

6.6. Adding pebbles to weighted automata: Easy specification & efficient evaluation

In [67], we extend weighted automata and weighted rational expressions with 2-way moves and (reusable) pebbles. We show with examples from natural language modeling and quantitative model-checking that weighted expressions and automata with pebbles are more expressive and allow much more natural and intuitive specifications than classical ones. We extend Kleene-Schützenberger theorem showing that weighted expressions and automata with pebbles have the same expressive power. We focus on an efficient translation from expressions to automata. We also prove that the evaluation problem for weighted automata can be done very efficiently if the number of (reusable) pebbles is low.

6.7. MSO decidability of multi-pushdown systems via split-width

Multi-threaded programs with recursion are naturally modeled as multi-pushdown systems. The behaviors are represented as multiply nested words (MNWs), which are words enriched with additional binary relations for each stack matching a push operation with the corresponding pop operation. Any MNW can be decomposed by two basic and natural operations: shuffle of two sequences of factors and merge of consecutive factors of a sequence. We say that the split-width of an MNW is k if it admits a decomposition where the number of factors in each sequence is at most k. The MSO theory of MNWs with split-width k is decidable. In [66], we introduce two very general classes of MNWs that strictly generalize known decidable classes and prove their MSO decidability via their split-width and obtain comparable or better bounds of tree-width of known classes.

6.8. Contextual Petri nets

Contextual nets (c-nets) are an extension of Petri nets that – unlike ordinary Petri nets – faithfully models concurrent read accesses to shared resources. This is not only interesting from a semantic but also from an algorithmic point of view, as the analysis of such nets can better exploit the fact that concurrent reads are independent and concurrent. In particular, the unfolding of a contextual net may be up to exponentially smaller in certain situations.

In previous work carried out in the Mexico project, we established theoretical foundations [6] and efficient algorithms for constructing c-net unfoldings [42]. More recently, we have investigated verification techniques based on c-nets. These exploit the advantages mentioned above to obtain results more efficiently. The results have been published in the Concur 2012 conference [70]. In parallel, the development of the Cunf tool has continued, see 5.1.2 . We are currently exploring how the technique can be combined with that of merged processes [107] for further speed-ups, and its applications in diagnosis.

6.9. Expressivity and Complexity of Timed Models

In [68], we show how to reliably compute fast-growing functions with timed-arc Petri nets and data nets. This construction provides ordinal-recursive lower bounds on the complexity of the main decidable properties (safety, termination, regular simulation, etc.) of these models. Since these new lower bounds match the upper bounds that one can derive from wqo theory, they precisely characterise the computational power of these so-called "enriched" nets. In [50], we characterize the importance of resources (like counters, channels, or alphabets) when measuring the expressiveness of Well-Structured Transition Systems (WSTS). We establish, for usual classes of well partial orders, the equivalence between the existence of order reflections (non-monotonic order embeddings) and the simulations with respect to coverability languages. We show that the non-existence of order reflections can be proved by the computation of order types. This allows us to extend the current classification of WSTS, in particular solving some open problems, and to unify the existing proofs.

6.10. Concurrent Games on Infinite State Systems

In [65], we propose to study concurrent games on a new extension of Vector Addition Systems with States, where inhibition conditions are added for modeling purposes. Games are a well-suited framework to solve control problems, and concurrent semantics reflect realistic situations where the environment can always produce a move before the controller, although it is never required to do so. This is in contrast with previous works, which focused mainly on turn-based semantics. Moreover, we consider asymmetric games, where environment and controller do not have the same capabilities, although they both have restricted power. In this setting, we investigate reachability and safety objectives, which are not dual to each other anymore, and we prove that (i) reachability games are undecidable for finite targets, (ii) they are 2-EXPTIME-complete for upward-closed targets and (iii) safety games are co-NP-complete for finite, upward-closed and semi-linear targets. Moreover, for the decidable cases, we build a finite representation of the corresponding controllers.

6.11. Rare Event Analysis for Markovian Systems

Model checking real time properties on probabilistic systems requires computing transient probabilities on continuous time Markov chains. Beyond numerical analysis ability, a probabilistic framing can only be obtained using simulation. This statistical approach fails when directly applied to the estimation of very small probabilities. In [60], combining the uniformization technique and extending our previous results, we design a method which applies to continuous time Markov chains and formulas of a timed temporal logic. The corresponding algorithm has been implemented in our tool cosmos. We present experimentations on a relevant system, with drastic time reductions with respect to standard statistical model checking.

Statistical model-checking is an alternative verification technique applied on stochastic systems whose size is beyond numerical analysis ability. Given a model (most often a Markov chain) and a formula, it provides a confidence interval for the probability that the model satisfies the formula. One of the main limitations of the statistical approach is the computation time explosion triggered by the evaluation of very small probabilities. In order to solve this problem, we develop in [59] a new approach based on importance sampling and coupling. The corresponding algorithms have been implemented in our tool cosmos. We present experimentation on several relevant systems, with estimated time reductions reaching a factor of $10^{1}20$.

6.12. Conformance Relations for Labeled Event Structures

In [69], we have extended several well known conformance (ioco style) relations for sequential models, to the concurrent framework of labeled event structures. With the interleaving semantics, the relations we obtained boil down to the same relations defined for labeled transition systemss. By contrast, under the partial order semantics, the relations we obtain allow to distinguish explicitly implementations where concurrent actions are implemented concurrently, from those where they are interleaved, i.e. implemented sequentially. Therefore, these relations will be of interest when designing distributed systems, since the natural concurrency between actions that are performed in parallel by different processes can be taken into account. In particular, the fact of being unable to control or observe the order between actions taking place on different processes will not be considered as an impediment for testing.

A complete testing framework for concurrent systems has been developped. We studied what kind of systems are testable in such a framework and we have proposed sufficient conditions for obtaining a complete test suite. Finally, an algorithm to construct a test suite with such properties was proposed. These result are summarized in a paper that is being prepared for a journal submission.

PAREO Project-Team

6. New Results

6.1. Model transformation

Participants: Jean-Christophe Bach, Pierre-Etienne Moreau.

In [10], we have proposed a general method to transform high level models by using *Tom* strategies. High-level models we consider are *EMF-ECore* models that we represent by terms whose mappings have been generated by the *Tom-EMF* tool. The proposed method consists in decomposing a complex transformation into many elementary transformations (*definitions*) encoded by *Tom* strategies. These *definitions* are applied on a source model without any consideration of execution order. Therefore, we proposed a mechanism to address the problem of dependency between elementary transformations without introducing any scheduling between rewriting rules. This mechanism relies on the use of temporary elements which play the roles of the target elements until the last part of the transformation : the *Resolve* phase. The goal of this phase is to find and replace all temporary elements by real target ones, and therefore to reconnect all partial target models obtained during elementary transformations to build the resulting model.

In [11], [15], we presented a first proposal of a high-level transformation language included in *Tom* which implements the aforementioned general method. We used this language to implement an avionic case study — AADL2Fiacre — which was proposed by Airbus for the *quarteFt* project.

6.2. Improvements of theoretical foundations

6.2.1. Termination under strategies

Participants: Horatiu Cirstea, Pierre-Etienne Moreau.

Several approaches for proving the confluence and the termination of term rewriting systems have been proposed [16] and the corresponding techniques have been implemented in tools like Aprove [23] and TTT2 [32]. On the other hand, there are relatively few works on the study of these properties in the context of strategic rewriting and the corresponding results were generally obtained for some specific strategies and not within a generic framework. It would thus be interesting to reformulate these notions in the general formalism we have previously proposed [21] and to establish confluence and termination conditions similar to the ones used in standard rewriting.

We have first focused on the termination property and we targeted the rewriting strategies of the *Tom* language. We propose a direct approach which consists in translating *Tom* strategies into a rewriting system which is not guided by a given evaluation strategy and we show that our systematic transformation preserves the termination. This allowed us to take advantage of the termination proof techniques available for standard rewriting and in particular to use existing termination tools (such as Aprove and TTT2) to prove the termination of strategic rewriting systems. The efficiency and scalability of these latter tool has a direct impact on the performances of our approach especially for complex strategies for which an important number of rewrite rules could be generated. We have nevertheless proposed a meta-level implementation of the automatic transformation which improves significantly the performances of the approach.

6.2.2. Automatizing the certification of induction proofs

Participant: Sorin Stratulat.

Largely adopted by proof assistants, the conventional induction methods based on explicit induction schemas are non-reductive and local, at schema level. On the other hand, the implicit induction methods used by automated theorem provers allow for lazy and mutual induction reasoning. In collaboration with Amira Henaien [13], we devised a new tactic for the Coq proof assistant able to perform automatically implicit induction reasoning. By using an automatic black-box approach, conjectures intended to be manually proved by the certifying proof environment that integrates Coq are proved instead by the `Spike` implicit induction theorem prover. The resulting proofs are translated afterwards into certified Coq scripts.

As a case study, conjectures involved in the validation of a non-trivial application [35] have been successfully and directly certified by Coq using the `Spike` tactic. The proofs of more than 60% of them have been performed completely automatically, i.e., the Coq user does not need to provide any argument to the tactic. On the other hand, its application is limited to Coq specifications transformable into conditional specifications whose axioms can be oriented into rewrite rules.

6.2.3. Cyclic proofs by induction methods

Participant: Sorin Stratulat.

In a first-order setting, two different 'proof by induction' methods are distinguished: the conventional induction, based on explicit induction schemas, and the implicit induction, based on reductive procedures. In [14], we proposed a new cycle-based induction method that keeps their best features, i.e., performs local and non-reductive reasoning, and naturally fits for mutual and lazy induction. The heart of the method is a proof strategy that identifies in the proof script the subset of formulas contributing to validate the application of induction hypotheses. The conventional and implicit induction are particular cases of our method.

6.3. Integration of formal methods in programming languages

6.3.1. Multi-focus strategies

Participants: Jean-Christophe Bach, Christophe Calvès, Horatiu Cirstea, Pierre-Etienne Moreau.

Like most rewriting engines, *Tom* patterns combined with traversal strategies, gives the possibility to match and rewrite at any position in a given term. We have extended this classical approach with multi-focus strategies which enable us to match and rewrite several positions simultaneously. More precisely, the action performed at a given position can depend on the other positions involved in the corresponding strategy. This extension is particularly well-suited for programming-language semantics specification, semantics which usually require gathering several subterms (code, memory, input/output channels, ...) to perform one action.

The multi-focus library is a conservative extension of *Tom* standard strategies and provides combinators to handle multi-position traversal, matching and rewriting. Compared to the original *Tom* strategy library, the multi-focus version provides global backtracking. The library is available at http://gforge.inria.fr/projects/tom.

6.3.2. Formal islands grammars parsing

Participants: Jean-Christophe Bach, Pierre-Etienne Moreau.

Extending a language by embedding within it another language presents significant parsing challenges, especially if the embedding is recursive. The composite grammar is likely to be nondeterministic as a result of tokens that are valid in both the host and the embedded language. In [9], we examined the challenges of embedding the *Tom* language into a variety of general-purpose high level languages. The current parser of *Tom* is complex and difficult to maintain. In this paper, we described how *Tom* can be parsed using island grammars implemented with the Generalised LL (*GLL*) parsing algorithm. The grammar is, as might be expected, ambiguous. Extracting the correct derivation relies on a disambiguation strategy which is based on pattern matching within the parse forest. We described different classes of ambiguity and proposed patterns to solve them.

6.4. Security policies specification and analysis

Participants: Horatiu Cirstea, Hélène Kirchner, Pierre-Etienne Moreau.

Access control policies, a particular case of security policies should guarantee that information can be accessed only by authorized users and thus prevent all information leakage. We proposed [12] a framework where the security policies and the systems they are applied on are specified separately but using a common formalism. This separation allows not only some analysis of the policy independently of the target system but also the application of a given policy on different systems. In this framework, we propose a method to check properties like confidentiality, integrity or confinement over secure systems based on different policy specifications.

PARSIFAL Project-Team

6. New Results

6.1. Recovering Proof Structures in the Sequent Calculus

Participants: Kaustuv Chaudhuri, Stefan Hetzl, Dale Miller.

The *sequent calculus* is often criticized as a proof syntax because it contains a lot of noise. It records the precise minute sequence of operations that was used to construct a proof, even when the order of some proof steps in the sequence is irrelevant and when some of the steps are unnecessary or involve detours. These features lead to several technical problems: for example, cut-elimination in the classical sequent calculus LK, as originally developed by Gentzen, is not confluent, and hence proof composition in LK is not associative. Many people choose to discard the sequent calculus when attempting to design a better proof syntax with the desired properties.

In recent years, there has been a project at Parsifal to recover some of these alternative proof syntaxes by imposing a certain abstraction over sequent proofs. The earliest example of this was in [37], where we showed a class of sequent proofs that were isomorphic to proof nets for multiplicative linear logic. In 2012, we were able to obtain a similar result for first-order classical logic, wherein we defined a class of sequent proofs that are isomorphic to expansion trees, a generalization of Herbrand disjunctions that is in some sense a minimalistic notion of proof for classical logic. This result was published at the CSL 2012 conference [22] and a journal version is in preparation.

Our technique for recovering these dramatically different proof structures directly in the sequent calculus involves the use of *maximal multi-focusing* which gives a syntactic characterization of those sequent proofs that: (1) have a "don't care" ordering of proof steps where the order does not matter, and (2) groups larger logical steps, called *actions*, into a maximally parallel form where only important orderings of actions are recorded. This technique was pioneered at Parsifal, and we have barely scratched the surface of its applications.

6.2. Compact Proof Certificates By Bounded Contractions

Participant: Kaustuv Chaudhuri.

An important engineering question in the ProofCert project is that of communicating, manipulating, and storing formal proof certificates. A fully detailed proof certificate, especially one generated by proof search, can be very large. Using such proofs would require a high bandwidth interface between the proof producer and consumer, which limits the scalability of the *ensemble of proving systems* approach. It is therefore natural to ask if there are more compact formats for proof certificates. The ideal format would have a tunable level of detail, so that the size of the certificates can be tailored to the application domain.

Suppose the proof consumer is equipped with some proof search capabilities. What then needs to be transmitted to the consumer to guarantee that it can check a proof within desired complexity bounds? It turns out that there is a systematic and general answer to this problem: use *focusing* and record only the "decision" rules of focusing in the proof certificate. From a high level perspective, this answer is equivalent to designing a proof system where the contraction rules are carefully bounded.

A proposal along these lines was published at the CPP 2012 conference [21]. In fact, this paper solves a harder than necessary problem by building proof certificates for linear logic, where unconstrained proof search has very high complexity even in the propositional fragment. The proposed solution is a spectrum of certificates that trades off the size of the certificate for the complexity of checking the certificate. At one end we have a very compact certificate that essentially amounts to a maximum depth of the proof, but reconstructing a proof with only a depth bound tends to be infeasible as the search space grows super-exponentially with the depth. Certificates at other end of the spectrum contain information about all the contractions in the proof; these certificates can be checked deterministically, in time proportional to the size of the certificate. Moreover, there is a simple abstraction mechanism between different levels of detail in this spectrum that allows for a *proof elaborator* to alter the level of detail in the certificate.

6.3. A Two-level Approach to Reasoning about Computation

Participant: Dale Miller.

In a paper that appeared in the J. of Automated Reasoning, Gacek, Miller, and Nadathur [12] described the foundations and architecture of a new interactive theorem prover capable of reasoning with rich collections of inductive and coinductive relations. This prover, called Abella, also contains the "generic" quantifier ∇ that provides a direct and elegant treatment of term-level binding.

A novel aspect of Abella is that it can define provability in various simple logics and can also reason about provability in such logics. The current system includes a *specification logic* that is a (restricted) intuitionistic logic programming language (a sublanguage of λProlog). The main logic of Abella is then the second logic, called the *reasoning logic*, and it is capable of reasoning about provability in the specification language.

This approach to reasoning about computation has interesting applications. For example, the reasoning logic is aware of the fact that the cut and substitution rules can be eliminated in the specification logic. As a consequence, the notoriously difficult "substitution lemmas" that occur repeated in the study of operational semantics are proved essentially for free (that is, they are an immediate consequence of cut-elimination).

In [17], Accattoli showed that when one reasons about the *untyped*λ-calculus, the specification logic is often not needed. In particular, Accattoli reinterpreted the formalization by G. Huet of the meta-theory of λ-calculus residuals in Abella and showed that the resulting meta-theory had a much more elegant and natural specification than the one presented early by Huet in Coq. While the use of two-levels of logic was not important for this particular (untyped) example, other aspects of Abella—relation specifications, ∇-quantification, and strong induction principles—were critical for improving the expressivity of this prover.

6.4. A Non-local Method for Robustness Analysis of Floating Point Programs

Participants: Dale Miller, Ivan Gazeau.

Programs that must deal with floating point programs and their associate errors can have erratic behavior. In particular, a program that yields outputs that depend continuously on their inputs (in an idealized arithmetic setting) can behave non-continuously when using floating point arithmetic. There are few tools for reasoning about program correctness in a setting that allows for such discontinuous operators.

In [23], Gazeau, Miller, and Palamidessi provide an approach to reason about some programs that are not continuous. In that paper, they introduce the notion of "robustness", which intuitively means that if the input to the program changes less than a fixed small amount then the output changes only slightly. This notion is useful in the analysis of rounding error for floating point programs because it helps to establish bounds on output errors introduced by both measurement errors and by floating point computation. Compositional methods often do not work since key constructs—like the conditional and the while-loop—are not robust. The authors proposed a method for proving the robustness of a while-loop. This method is non-local in the sense that instead of breaking the analysis down to single lines of code, it checks certain global properties of its structure. This paper shows that both the CORDIC computation of the cosine and Dijkstra's shortest path algorithm are robust.

6.5. Herbrand Confluence

Participants: Stefan Hetzl, Lutz Straßburger.

It is well-known that cut-elimination in the sequent calculus for classical first-order logic is in its most general form, is neither confluent nor strongly normalizing. But if one takes a coarser (and mathematically more realistic) look at cut-free proofs, one can analyze which witnesses they choose for which quantifiers, or in other words: one can only consider the Herbrand-disjunction of a cut-free proof. This yields a surprising confluence result for a natural class of proofs: all (possibly infinitely many) normal forms of the non-erasing cut reduction lead to the same Herbrand-disjunction. This result has been presented at CSL 2012 [25].

6.6. Semi-Star-Autonomous Categories

Participants: Willem Heijltjes, Lutz Straßburger.

A curious aspect of Girard's proof nets for multiplicative linear logic without units is that, despite being a canonical representation of proof, their categorical semantics is not obvious—this in contrast to the situation *with* units, where star-autonomous categories form a natural semantics, but no canonical proof nets are known.

In the middle of the past decade several proposals for a categorical semantics of proof nets, a notion of *semi-star-autonomous* categories, were investigated: by Robin Houston and Dominic Hughes, by Kosta Došen, and by François Lamarche and Lutz Straßburger.

The present effort by Willem Heijltjes and Lutz Straßburger completes the notion in such a way that proof nets constitute the *free* semi-star-autonomous category.

6.7. Foundations and applications of explicit substitutions

Participant: Beniamino Accattoli.

Starting from the study of Linear Logic proof nets, a new approach to explicit substitutions for ł-calculus has recently been introduced by Accattoli and D. Kesner [31]. This approach has been systematically explored by Accattoli and his co-authors.

The rewriting theory of these new explicit substitutions *at a distance* has been studied in [11] and [16]. In [11] Accattoli and Kesner study the preservation of λ-calculus strong normalization (PSN) when explicit substitutions are extended with permutative axioms allowing to swap constructors in the term, generalizing considerably the already difficult case of PSN with composition of substitutions. In [16] Accattoli developed an abstract technique for proving factorizations theorems for generic explicit substitution calculi. The factorization theorem for λ-calculus says that any reduction can be re-organized as an *head* reduction followed by a non-head reduction.

In [16] it is shown how to prove this theorem in an uniform way for many explicit subsitutions calculi. The technique emerged as a generalization of the proofs for explicit substitutions at a distance, which are simpler than usual explicit substitutions and thus lead to cleaner and more compact arguments, easier to generalize.

Applications of explicit substitutions at a distance have been studied in [19], [18], [20]. In [19] Accattoli and Dal Lago show that the length of the head reduction in calculi at a distance is a measure of time complexity. More precisely, they show that such a quantity is polynomially related (in both directions) to the cost of evaluating with Turing Machines. This result is an important step forward towards the solution of the long-standing open problem of finding a time cost model for ł-calculus.

In [20] Accattoli and Paolini apply substitutions at a distance in a call-by-value setting. They show that in this new framework there is a natural characterization of *solvability*, an important notion related to denotational semantics and the representation of partial recursive functions. In [26] (a work presented to a workshop and currently submitted to the post-proceedings of the workshop) Accattoli shows the tight relations between the framework in [20] and linear logic proof nets, providing a new characterization of the proof nets representing the call-by-value λ-calculus.

Finally, in [18] Accattoli and Kesner introduce a calculus generalizing many different extensions of λ-calculus with permutations, appeared in various contexts (studies about call-by-value, postponing of reductions, monadic languages, etc) and prove confluence and preservation of strong normalization, exploiting and extending their own results in [11].

6.8. Sequent Calculus with Calls to a Decision Procedure

Participants: Mahfuza Farooque, Stéphane Lengrand.

In the PSI project, we have designed a version of the focussed sequent calculus (for first-order classical logic) that can call external decision procedures. Since the last Activity Report, we have finished proving the essential meta-theory for it: soundness, invertibility of asynchronous rules, cut-elimination, the fact that polarities do not affect provability but only the shape of proofs, and finally completeness.

The first properties are the object of [27], while the latter ones have been obtained later in 2012.

A side-product of this meta-theory is a technical device that could be used to encode other techniques from automated reasoning like *connection tableaux*.

Secondly, we have encoded the SMT-solving algorithm DPLL(T) as the incremental construction of proof-trees in that sequent calculus [29], [28]. A very interesting aspect of the encodings is that the basic rules of DPLL(T) makes use of cuts on atoms in sequent calculus, while the advanced jrules (e.g. backjumping) makes use of general cuts. This sheds a new light on the computational speed-ups that those advanced rules provide.

We have done the encoding for two distinct presentations of DPLL(T) in the literature, and we have formalised the connection between those two descriptions [29].

6.9. Martin-Löf Identity Type in the Category of Small Categories

Participant: François Lamarche.

For the last five or six years there has been a surge of interest in finding models for the identity type in Martin-Löf type theory, and it has been clear for some time that there was a tight connection with path objects in abstract homotopy theory. A lot of proposals have been made, but there are very few semantics that fit the necessary requirements of having dependent products and also an identity type which is fully stable under substitution. The most famous model of the sort is the one proposed by Voevodsky, in his Univalent Foundations project, which uses for base category the category of simplicial sets and models dependent types by the means of Kan Fibrations. In [13] François Lamarche proposes another such model, where the base category is the category of small categories, and dependent types are modelled with Grothendieck bifibrations (maps between categories that are Grothendieck fibrations as well as their duals between the opposite categories). The full requirements of modelling Martin-Löf type theory are met. Calculations show that the model shows some amount of degeneracy “in dimensions above 2” for the associativity of equality (which should not be strict in any dimension), which is a great improvement over the models on strict groupoids and strict ω-groupoids. The construction that models the identity type is a concrete path functor for categories. It is showing itself to be very useful in homotopy theory.

PI.R2 Project-Team

6. New Results

6.1. Proof-theoretical and effectful investigations

Participants: Federico Aschieri, Pierre Boutillier, Pierre-Louis Curien, Hugo Herbelin, Guillaume Munch-Maccagnoni, Pierre-Marie Pédrot, Alexis Saurin, Arnaud Spiwack.

6.1.1. Sequent calculus and computational duality

System L syntax. Pierre-Louis Curien studied in some detail the differences (and translations) between variants of "system L" syntax for polarised classical logic (developed by Guillaume Munch-Maccagnoni and himself):

- weakly focalised systems (where negatives can be worked on at any moment in a proof) versus focalised systems (where negative and positive phases alternate strictly), versus strongly focalised systems (where furthermore negative phases have to decompose negatives completely);
- systems where changes of polarity are implicit (like in Girard's LC) versus systems where they are explicitly marked using shift operators. These shift operators are formally adjoint, and as a matter of fact a suitable intuitionistic fragment of system L corresponds exactly to Levi's CBPV;
- systems with stoup (which retain only proofs that follow the focalisation discipline) versus (still focalised) systems without stoup (where the focalisation is forced by the dynamics of reduction);
- one-sided systems (with an implicit negation given by De Morgan duality) versus two-sided systems (allowing for explicit negation, and for distinguishing the left/right and the positive/negative dualities).

Pierre-Louis Curien is also currently studying a polarised version of a notion of general connective suggested earlier by Hugo Herbelin (unpublished work), and the composition structure of these connectives (in the spirit of operads).

Categorical semantics. Guillaume Munch-Maccagnoni investigated a notion of "direct style" for adjunction models, inspired by his work on polarisation in the "L" system, in the lineage of Führmann's [47] direct-style characterisation of monadic models. (It is part of joint work with Marcelo Fiore and Pierre-Louis Curien.)

Polarised Peano arithmetic. Guillaume Munch-Maccagnoni investigates the computational contents of polarised classical logic in arithmetic and in natural deduction. This allows him to compare the constructivisation of the principle $\neg\forall \Rightarrow \exists\neg$ based on classical realisability (Krivine) and the one based on delimited control (via "double negation shift"); both of which seem to be simplified by a better understanding of the "formulae-as-types" paradigm for a negation which is involutive in a strong sense.

Guillaume Munch-Maccagnoni investigates how a notion of classical realisability structure (inspired by Krivine's) can be used to prove properties of type systems which are usually regarded as syntactic.

Classical call-by-need and the duality of computation. In 2011, Zena Ariola, Hugo Herbelin and Alexis Saurin characterised the semantics of call-by-need calculus with control in the framework of *the duality of computation*. The same set of authors extended with Paul Downen and Keiko Nakata worked on abstract machines and continuation-passing-style semantics for call-by-need with control, resulting into a paper presented at FLOPS 2012 [20].

Further work has been done by Zena Ariola, Hugo Herbelin, Luís Pinto, Keiko Nakata and José Espírito Santo on typing the continuation-passing-style of call-by-need calculus, opening the way to a proof of normalisation of simply-typed call-by-need with control, and from there to a proof of consistency of classical arithmetic with dependent choice.

Zena Ariola also investigated how to formulate a parametric theory which encompasses call-by-value, call-by-name and call-by-need. Each theory is obtained by giving the appropriate definition of what is a value and a co-value. The theory also includes so called lifting axioms which allow one to relax the syntactic restrictions previously imposed on the call-by-value, call-by-name and call-by-need calculi. The theory also allows to include the η-rules which before were causing confluence to fail. The approach can be applied to natural deduction and this allows to express different embeddings of natural deduction into sequent calculus directly in the theory. The advantage of the new formalisation is that analogously to natural deduction, one can experiment with different strategies starting from the same term. Moreover, the theory is well-suited for continuation passing style transformation and, in particular, it leads to a different and simpler formalisation of classical call-by-need, its abstract machine and continuation passing style.

6.1.2. Dependent monads

Pierre-Marie Pédrot generalised the notion of monad in order to be able to use it in a dependent framework. This new structure allows to write effects in a pure functional language, such as Coq, through a monadic encoding.

This way, the whole monadic apparatus can be lifted to dependent programs, as well as proofs.

6.1.3. Linear dependent types

Arnaud Spiwack continued his investigations on dependently typed linear sequent calculus (based on Curien & Herbelin's $\mu\tilde{\mu}$). The current version of his system resembles Andreoli's focalised linear logic (yet to be published).

Pierre-Marie Pédrot has been working on a delimited CPS translation of the Calculus of Inductive Constructions, seen through the prism of polarised linear logic. Restricting dependencies to positives naturally fits into the scenery of delimited control, while extending negatives to infinitary objects permits to recover some properties of the involutivity of linear double-negation.

6.1.4. Proving with side-effects

Axiom of dependent choice. Hugo Herbelin showed that classical arithmetic in finite types extended with strong elimination of existential quantification proves the axiom of dependent choice. To get classical logic and choice together without being inconsistent is made possible first by constraining strong elimination of existential quantification to proofs that are essentially intuitionistic and secondly by turning countable universal quantification into an infinite conjunction of classical proofs evaluated along a call-by-need evaluation strategy so as to extract from them intuitionistic contents that complies to the intuitionistic constraint put on strong elimination of existential quantification. This work has been presented at LICS 2012 [22].

Memory assignment, forcing and delimited control. Hugo Herbelin investigated how to extend his work on intuitionistically proving Markov's principle [54] and the work of Danko Ilik on intuitionistically proving the double negation shift (i.e. $\forall x\,\neg\neg A \to \neg\neg\forall x\,A$) [15] to other kind of effects. In particular, memory assignment is related to Cohen's forcing as emphasised by Krivine [58] and by the observation that Cohen's translation of formula P into $\forall y \leq x \exists z \leq y\, P(z)$ is similar to a state-passing-style transformation of type P into $S \to S \times P$.

Hugo Herbelin then designed a logical formalism with memory assignment that allows to *prove* in direct-style any statement provable using the forcing method, the same way as logic extended with control operators allows to support direct-style classical reasoning. Thanks to the use of delimiters over "small" formulas similar to the notation of Σ^0_1-formulas in arithmetic, the whole framework remains intuitionistic, in the sense that it satisfies the disjunction and existence property.

Two typical applications of proving with side-effects are global-memory proofs of the axiom of countable choice and an enumeration-free proof of Gödel's completeness theorem.

The main ideas of this research program have been communicated during the Logic and Interaction weeks in Marseille in February 2012.

In the continuation of his work with Silvia Ghilezan [4] on showing that Saurin's variant $\Lambda\mu$ [8] of Parigot's $\lambda\mu$-calculus [65] for classical logic was a canonical call-by-name version of Danvy-Filinski's call-by-value calculus of delimited control, Hugo Herbelin studied with Alexis Saurin and Silvia Ghilezan another variant of call-by-name calculus of delimited control. This is leading to a general paper on call-by-name and call-by-value delimited control.

Classical logic, stack calculus and stream calculus. Alexis Saurin studied the connection between the stack calculus recently proposed by Ehrhard et al and $\lambda\mu$-calculus and how the former can be precisely compared to the target of the CPS of the latter. He also investigated separation issues related to the stack calculus. During a visit to UPenn in the spring, Alexis Saurin and Marco Gaboardi investigated type systems for a stream calculus which contains $\Lambda\mu$.

Moreover, Alexis Saurin's paper *Böhm theorem and Böhm trees for the $\Lambda\mu$-calculus* [16] was published in TCS early 2012.

6.1.5. PTS and delimited control

From the study of one-pass CPS on the one side and of previous presentations of pure type systems with control operators on the other side, Pierre Boutillier and Hugo Herbelin have investigated how splitting terms into categories opens a new way to merge dependent types and monads. A preliminary set of rules has been presented during the third week of Logic and Interaction in Marseille.

It was refined since then but has not reached yet the maturity required to be accepted for publication in an international conference.

6.1.6. Interactive realisability

Thanks to the Curry-Howard correspondence for classical logic, it is possible to extract programs from classical proofs. These programs use control operators as a way to implement backtracking and processes of intelligent learning by trial and error. Unfortunately, such programs tend to be poorly efficient. The reason is that, in a sense, they are designed in order to keep their correctness and termination proofs simple. Each small modification of these programs seems, at best, to require major and difficult adaptations of their correctness proofs. This is due to a lack of understanding and control of the backtracking mechanism that interprets classical proofs. In order to write down more efficient programs, it is necessary to describe exactly: a) what the programs learn, b) how the knowledge of programs varies during the execution.

A first step towards this goal is the theory of Interactive Realisability, a semantics for intuitionistic arithmetic with excluded middle over semi-decidable predicates. It is based on a notion of state, which describes the knowledge of programs coming from a classical proof, and explains how the knowledge evolves during computation.

Federico Aschieri has extended the theory of interactive realisability to a full classical system, namely first-order Peano arithmetic with Skolem axioms. This is a very expressive system, with non-trivial axioms of choice and comprehension. The resulting programs are interpreted as stratified-learning algorithms, which build in a very organised way the approximations of the Skolem functions used in the proofs. The work has appeared in the proceedings of the conference Computer Science Logic 2012. A careful implementation of this extended theory –yet to be developed – will lead to a dramatic efficiency improvement over the already existing computational interpretations.

Federico Aschieri has also showed how to use interactive realisability to provide purely proof-theoretic results. He proved with a new method the conservativity of Peano arithmetic with Skolem axioms over Peano arithmetic alone for arithmetical formulas. In particular, the method can be seen as a constructivisation and substantial refinement of Avigad's forcing. The work has appeared in the proceedings of the workshop Classical Logic and Computation 2012.

6.1.7. Reverse mathematics

Hugo Herbelin explored with Gyesik Lee and Keiko Nakata the constructive content of the big five subsystems of second-order arithmetic considered in the context of (classical) reverse mathematics. They obtained a

uniform characterisation of these systems in terms of variants of the comprehension axiom called separation, co-separation and interpolation.

This is the first step in a larger project attempting first to connect to predicative type theory the subsystems of System F underlying the proof-as-program structure of the big five subsystems of second-order arithmetic, and secondly to reformulate these subsystems in terms of pure systems of inductive definitions.

Jaime Gaspar has several projects running simultaneously. For example, in one of his projects he created a small unoptimised automated theorem prover, and he hopes to optimise it and use it to obtain a certain completely formalised proof to which he can apply a proof interpretation in order to extract computational content. As another example, in another project he his trying to show that several classical proof interpretations are instances of a unified proof interpretation, in a parallel way to what is known for intuitionistic proof interpretations.

6.2. Type theory and the foundations of Coq

Participants: Pierre Boutillier, Pierre-Louis Curien, Hugo Herbelin, Pierre-Marie Pédrot, Yann Régis-Gianas, Alexis Saurin, Matthieu Sozeau.

6.2.1. Calculus of inductive constructions and typed equality

The work of Hugo Herbelin and Vincent Siles on the equivalence of Pure Type Systems with typed or untyped equality has been published [17].

6.2.2. Substitutions and isomorphisms

Pierre-Louis Curien completed his joint work with Martin Hofmann (Univ. of Munich) and Richard Garner (MacQuarie University, Sydney) on comparing two categorical interpretations of (extensional) type theories. More precisely, we wanted to compare two ways of giving a categorical interpretation of Martin-Löf type theory, both overcoming the following mismatch: syntax has exact substitutions, while their categorical interpretation, in terms of pullbacks or fibrations, "implements" substitutions only up to isomorphism. One can then either change the model (strictification) [55], or modify the syntax (by introducing explicit substitutions and more importantly explicit coercions between types that are now only isomorphic) [2]. In the latter case, one has to prove a coherence theorem to show that the interpretation is in the end independent from these coercion decorations. Such a proof was given in [2], using rewriting methods. These approaches turn out to be related through a general machinery that relates three kinds of categories, with strict or non strict objects and morphisms. As a bonus we get a new, more conceptual proof of coherence. These results are now being written up for a special issue in honour of Glynn Winskel. In further work, we wish to address intensional, and homotopy type-theoretic versions of these coherence problems.

6.2.3. Homotopy type theory

The univalence axiom proposed by Voevodsky states that for any two types to be equal exactly means being of same cardinality. This new axiom for type theory turns to have very interesting consequences for the practical foundations of formal mathematical reasoning: it smoothly implies other axioms such as functional extensionality or propositional existential but before all it says that any property proved about some mathematical structure immediately applies to any other other type ("sets" informally) which it is isomorphic to.

This axiom however contradicts the current logical foundations of Coq (in the presence of Streicher's axiom K). Investigations have then been started to understand how to weaken the Calculus of Inductive Constructions implemented in Coq so as to make it compatible with univalence. In a first step, this resulted in the design of a new rule for singleton elimination that has been implemented by Hugo Herbelin as an optional feature of Coq (singleton elimination is the ability to build objects in datatypes from canonically-proved propositional properties such as equality).

6.2.4. Models of type theory

The existing models of homotopy type theory are based on simplicial sets or on their extensions as Kan complexes. Hugo Herbelin developed a concrete type-theoretic formalisation of semi-simplicial sets following ideas from Steve Awodey, Peter LeFanu Lumsdaine and other researchers both at Carnegie-Mellon University and at the Institute of Advanced Study. The technique he used seems to straightforwardly generalise to provide type-theoretic constructions for arbitrary presheaves on inductively generated categories.

6.2.5. Forcing in type theory

Together with Nicolas Tabareau and Guilhem Jaber (Inria Ascola team, École des mines de Nantes), Matthieu Sozeau investigated an internalisation of the presheaf model of the Calculus of Inductive Constructions (CIC). They published their work at LICS'12 [23]. This work corresponds to adapting the idea of Forcing due to Cohen in Type Theory. An internal model construction allows to enrich the logical type theory with new modalities and define their semantics by translation to CIC. The usual Cohen forcing can be realised using this framework to show the independence of the continuum hypothesis in CIC, but more practical applications are possible as well. Notably, the step-indexed technique for building models of imperative languages with rich type structure can be phrased as a forcing/presheaf construction. Sozeau, Tabareau and Jaber developed a plugin that can handle this example [32] which relies on a modified Coq version implementing proof-irrelevance and eta-rules for records.

6.2.6. Proof irrelevance, eta-rules

Matthieu Sozeau continued his work on proof-irrelevance by implementing a variant of Werner's proof-irrelevant CIC in Coq [72]. An article describing this work is in preparation. The new system also handles the extensional eta-rules for records, extending the technique implemented by Hugo Herbelin to handle eta-expansion of functions in Coq.

6.2.7. Unification

The unification algorithm of Coq now essentially dwells in the λ-calculus part of the language. Pierre Boutillier started a refactoring of the code in order to deal with algebraic datatypes. Hugo Herbelin and Pierre Boutillier investigated how to reformulate unification on top of an abstract machine (i.e. on top of sequent calculus). Hugo Herbelin added various heuristics to the unification algorithm of Coq, making them both more powerful and customisable.

Matthieu Sozeau is continuing work in collaboration with Beta Ziliani (PhD student of Derek Dreyer at MPI Saarbrücken, two one week visits in 2012), and Aleksandar Nanevski (Researcher at IMDEA Madrid) on giving a clear formalisation for the unification algorithm of Coq. This will help understand better the working of advanced features like Canonical Structures and Type Classes that are heavily used in big developments, as the spectacular recently completed formalisation of the proof of Feit-Thompson's Odd theorem by the Mathematical Components team.

Matthieu Sozeau adapted the existing unification algorithm to be universe-aware, resulting in more predictability and earlier error-reporting in both the type inference and tactic unification algorithms of Coq.

6.3. Homotopy of rewriting systems

Participants: Pierre-Louis Curien, Yves Guiraud, Philippe Malbos.

6.3.1. Coherence in monoidal structures

Yves Guiraud and Philippe Malbos have applied the Squier's homotopical theorem [70], which they had generalised to higher-dimensional rewriting systems [52], to several types of categories with monoidal structures. This work develops a formal setting to produce constructive proofs of coherence conditions, applied to the cases of monoidal categories, symmetric monoidal categories and braided categories. These results have been published in Mathematical Structures in Computer Science [12].

6.3.2. Computation of resolutions of monoids

Yves Guiraud and Philippe Malbos have extended Squier's homotopical theory to the higher dimensions of presentations of monoids to get an algorithm transforming a convergent word rewriting system into a polygraphic resolution of the presented monoid, in the setting of Métayer [63]. From this polygraphic resolution, this work gives an explicit procedure to recover several of the known Abelian resolutions of the monoid, generalising and relating algebraic invariants of monoids. Moreover, a polygraphic resolution corresponds to the normalisation strategies of rewriting systems and they contain all the proofs of equality between elements, together with the proofs of equality of those proofs of equality, and so on. This work has been published in Advances in Mathematics [13]. By nature, polygraphic resolutions bear many similarities with the higher-dimensional groupoids that appear in homotopical type theory when one considers the towers of identity types: this connection will be investigated by Pierre-Louis Curien, Yves Guiraud, Hugo Herbelin and Matthieu Sozeau.

6.3.3. Coherent presentations of Artin groups

With Stéphane Gaussent (Institut Camille Jordan, Université de Saint-Étienne), Yves Guiraud and Philippe Malbos are currently finishing an article on the rewriting properties and coherence issues in Artin groups, a class of groups that is fundamental in algebra and geometry. This work uses the formal setting of coherent presentations (a truncation of polygraphic resolutions at the level above relations) to formulate, in a common language, several known results in combinatorial group theory: one by Tits about the fundamental group of a graph associated to an Artin group [71], and one by Deligne about the actions of Artin groups on categories [44], both proved by geometrical and non-constructive methods. In this work, an improvement of Knuth-Bendix's completion procedure is introduced, called the homotopical completion-reduction procedure, and it is used to give a constructive proof of both those theorems. In fact, the method even improves the formerly known results: for example, it generalises Deligne's result to cases where his geometrical proof cannot be applied. A preliminary version of this work is available online [31]. The next objective of this collaboration is to extend the formal setting and methods to compute polygraphic resolutions of Artin groups, with a view towards two open problems of combinatorial group theory with respect to Artin groups: the decidability of the word problem and the verification that a precise topological space is a classifying space.

6.3.4. Higher-dimensional linear rewriting

With Samuel Mimram (CEA Saclay) and Pierre-Louis Curien, Yves Guiraud and Philippe Malbos investigate the links between set-theoretic rewriting theory and the computational methods known in symbolic algebra, such as Gröbner bases [36]. In particular, this work is interested in extending the setting of higher-dimensional rewriting to include "linear rewriting" and, as a consequence, to be able to apply its methods in symbolic computation. One particular direction is to understand Anick's resolution [33], and to improve it with the completion-reduction methodology, in order to get better algorithms to compute homological invariants and to prove important properties such as Koszulness. This research direction has been presented to the first call for projects of the IDEX Sorbonne-Paris-Cité, together with Eric Hoffbeck and Muriel Livernet (LAGA, Université Paris 13) and François Métayer (PPS, Université Paris 7).

6.4. Coq as a functional programming language

Participants: Nicolas Ayache, Pierre Boutillier, François Bobot, Guillaume Claret, Lourdes del Carmen Gonzalez Huesca, Tim Griffin, Hugo Herbelin, Pierre Letouzey, Matthias Puech, Yann Régis-Gianas, Matthieu Sozeau.

6.4.1. Type classes and libraries

Pierre Castéran from Inria Bordeaux and Matthieu Sozeau published a tutorial on the use of type classes [30] that was used as the basis of an invited lecture by M. Sozeau at the JFLA conference in February 2012. It will be published as part of the new version of the Coq'Art book.

6.4.2. Dependent pattern-matching

Pierre Boutillier experimented about how to integrate gently in Coq the compilation process he came up with last year to simulate Agda-style dependent pattern-matching. As a consequence, pattern grammar in Gallina has changed, much more notations can be used and users can write patterns instead of simple abstractions in the pattern-matching return clause.

Matthieu Sozeau continued maintenance and polishing of the Equations plugin that allows dependent pattern-matching on inductive families. A first official release is planned for the beginning of 2013.

6.4.3. Modularised arithmetical libraries

The modularised arithmetical libraries elaborated by Pierre Letouzey during the previous year(s) have been released as part of Coq 8.4. They provided greater uniformity of available functions and lemmas across the various Coq numerical datatypes. These libraries seem to work quite well, the only remaining issue is the documentation: due to this complex modular organisation, it is currently tedious for the user to browse the available functions and results. We expect to tackle this issue next year, by providing various documentation views, either the external summary of all available elements at once, or the internal layout of these elements.

6.4.4. Library of finite sets

Pierre Letouzey has integrated an additional Coq implementation in the MSets library of finite sets. This additional implementation is an improved version of the Red-Black-Tree library contributed by Andrew Appel. Using these RBT instead of the previously available AVL could be more efficient, at least in Coq, since they trigger no computations of integer numbers coding the tree depth.

6.4.5. Library on XPath processing

As part of the ANR Typex (http://typex.lri.fr), Matthieu Sozeau is developing a library for the certification of efficient XPath/XQuery engines in collaboration with Kim Nguyen (LRI) and Alan Schmitt (Inria Grenoble).

6.4.6. Mathematics of routing

Tim Griffin's primary focus during his visit to πr^2 was the development of a "metalanguage" for algebraic structures using Coq. Since he was something of a beginner with Coq, this involved learning the basics as well as more advanced work on representing algebraic structures. He made very good progress on this while in Paris and is now continuing this work in Cambridge.

6.4.7. Incrementality in proof languages

Matthias Puech and Yann Régis-Gianas worked on incremental type checking. This preliminary work has been presented during a contributed talk at TLDI 2012 [25]. It sets the ground for an incremental proof development and checking system, by means of a representation language for repositories of proofs and proof changes.

The traditional interaction with a proof-checker is a batch process. Coq (among others) refines this model by providing a read-eval print loop with a global undo system, implemented in an ad-hoc way. A more general approach to incrementality is being developed by means of a finer-grained analysis of dependencies. The approach developed is adaptable to any typed formal language: the language is specified in a meta-language close to the Edinburgh Logical Framework, in which subsequent versions of a development can be type-checked incrementally. Applications of this framework are: proof language for proof assistants, integrated development environments for proof or programming languages, typed version control systems.

6.4.8. Proofs of programs in Coq

As part of the CerCo European project, in collaboration with Roberto Amadio (PPS, Paris 7), François Bobot, Nicolas Ayache and Yann Régis-Gianas maintained a prototype compiler for a large subset of the C language whose specificity is to annotate input source programs with information about the worst-case execution cost of their machine code. Yann Régis-Gianas started a mechanised version of the proof technique used to prove the correctness of such an annotating compiler.

Yann Régis-Gianas maintained another compiler for Core ML that uses a generalisation of CerCo technique in order to obtain certified worst case execution time bounds on functional programs. This compiler produces proof obligations in Coq. The corresponding paper is published in January 2012 in the proceedings of FOPARA 2011 [19].

Nicolas Ayache developed a Frama-C plugin distributed in the CerCo software suite whose role is to synthesize cost annotations out of C programs. François Bobot developed a new version of this plugin. In particular, this new version handles C programs that manipulate pointers.

In collaboration with Roberto Amadio, Yann Régis-Gianas extended the cost annotating compilation chain of the FOPARA paper to handle the cost of memory management. A journal paper is about to be published in HOSC.

6.4.9. Lightweight proof-by-reflection

In the context of the ANR project Paral-ITP, Lourdes del Carmen Gonzalez Huesca, Guillaume Claret and Yann Régis-Gianas developed a new technique for proof-by-reflection based on a notion of *a posteriori* simulation of effectful computations in Coq.

PROSECCO Project-Team

6. New Results

6.1. Verification of Security Protocols in the Symbolic Model

The symbolic model of protocols, or Dolev-Yao model is an abstract model in which messages are represented by terms. Our protocol verifier ProVerif relies on this model. This year, we have mainly worked on the verification of protocols with lists and on an extension of ProVerif to prove more observational equivalences.

6.1.1. Verification of Protocols with Lists

Participants: Bruno Blanchet [correspondant], Miriam Paiola.

security protocols, symbolic model, automatic verification, Horn clauses, secrecy

We have designed a novel, simple technique for proving secrecy properties for security protocols that manipulate lists of unbounded length, for an unbounded number of sessions [32]. More specifically, our technique relies on the Horn clause approach used in the automatic verifier ProVerif: we show that if a protocol is proven secure by our technique with lists of length one, then it is secure for lists of unbounded length. Interestingly, this theorem relies on approximations made by our verification technique: in general, secrecy for lists of length one does not imply secrecy for lists of unbounded length. Our result can be used in particular to prove secrecy properties for group protocols with an unbounded number of participants and for some XML protocols (web services) with ProVerif.

6.1.2. Proving More Process Equivalences with ProVerif

Participants: Bruno Blanchet [correspondant], Vincent Cheval.

security protocols, symbolic model, automatic verification, observational equivalence, privacy

We have extended the automatic protocol verifier ProVerif in order to prove more observational equivalences [28]. ProVerif can prove observational equivalence between processes that have the same structure but differ by the messages they contain. In order to extend the class of equivalences that ProVerif handles, we extend the language of terms by defining more functions (destructors) by rewrite rules. In particular, we allow rewrite rules with inequalities as side-conditions, so that we can express tests "if then else" inside terms. Finally, we provide an automatic procedure that translates a process into an equivalent process that performs as many actions as possible inside terms, to allow ProVerif to prove the desired equivalence. These extensions have been implemented in ProVerif and allow us to automatically prove anonymity in the private authentication protocol by Abadi and Fournet.

6.2. Verification of Security Protocols in the Computational Model

The computational model of protocols considers messages as bitstrings, which is more realistic than the formal model, but also makes the proofs more difficult. Our verifier CryptoVerif is sound in this model. This year, we have worked on a compiler from CryptoVerif speficications to OCaml, and we have used CryptoVerif to verify the password-based protocol One-Encryption Key Exchange (OEKE).

6.2.1. Generation of Implementations Proved Secure in the Computational model

Participants: Bruno Blanchet [correspondant], David Cadé.

security protocols, computational model, implementation, verification, compiler

We have designed a novel approach for proving specifications of security protocols in the computational model and generating runnable implementations from such proved specifications. We rely on the computationally-sound protocol verifier CRYPTOVERIF for proving the specification, and we have implemented a compiler that translates a CRYPTOVERIF specification into an implementation in OCaml [26]. We have also proved that this compiler preserves security [27]. We have applied this compiler to the SSH Transport Layer protocol: we proved the authentication of the server and the secrecy of the session keys in this protocol and verified that the generated implementation successfully interacts with OpenSSH. The secrecy of messages sent over the SSH tunnel cannot be proved due to known weaknesses in SSH with CBC-mode encryption.

6.2.2. Proof of One-Encryption Key Exchange using CryptoVerif

Participant: Bruno Blanchet [correspondant].

security protocols, computational model, automatic proofs, formal methods, password-based authentication

We have obtained a mechanized proof of the password-based protocol One-Encryption Key Exchange (OEKE) using the computationally-sound protocol prover CRYPTOVERIF [25]. OEKE is a non-trivial protocol, and thus mechanizing its proof provides additional confidence that it is correct. This case study was also an opportunity to implement several important extensions of CRYPTOVERIF, useful for proving many other protocols. We have indeed extended CRYPTOVERIF to support the computational Diffie-Hellman assumption. We have also added support for proofs that rely on Shoup's lemma and additional game transformations. In particular, it is now possible to insert case distinctions manually and to merge cases that no longer need to be distinguished. Eventually, some improvements have been added on the computation of the probability bounds for attacks, providing better reductions. In particular, we improve over the standard computation of probabilities when Shoup's lemma is used, which allows us to improve the bound given in a previous manual proof of OEKE, and to show that the adversary can test at most one password per session of the protocol.

6.3. New Attacks on RSA PKCS#1 v1.5

Participants: Graham Steel [correspondant], Romain Bardou.

cryptographic hardware, security API, key management, vulnerabilities

RSA PKCS#1v1.5 is the most commonly used standard for public key encryption, used for example in TLS/SSL. It has been known to be vulnerable to a so-called padding-oracle attack since 1998 when Bleichenbacher described the vulnerability at CRYPTO. The attack, known as the "million message attack" was not thought to present a practical threat, due in part to the large number of oracle messages required. In a paper published at CRYPTO 2012 [22] we gave original modifications showing how the attack can be completed in a median of just 15 000 messages. The results lead to widespread interest, indicated by over 1400 downloads of the long version of the paper from the HAL webpage and articles in the New York Times, Boston Globe and Süddeutscher Zeitung.

6.4. Security Proofs for Revocation

Participants: Graham Steel [correspondant], Véronique Cortier, Cyrille Wiedling.

security API, key management, formal methods, security proofs

Revocation of expired or corrupted keys is a common feature of industrially deployed key management systems but an aspect that is almost always missing from formal models. We succeeded in adding revocation to a formal specification of a key management API allowing the proof of strong security properties after corrupted keys are revoked. In particular we showed a self-healing property whereby after a corrupted key expires, after a certain amount of time, the system is safe again. The work was published at ACM CCS 2012.

6.5. Discovering Concrete Attacks on Web Applications by Formal Analysis

Participants: Karthikeyan Bhargavan [correspondant], Sergio Maffeis, Chetan Bansal, Antoine Delignat-Lavaud.

web application security, formal methods, automated verification, vulnerabilities Social sign-on and social sharing are becoming an ever more popular feature of web applications. This success is largely due to the APIs and support offered by prominent social networks, such as Facebook, Twitter, and Google, on the basis of new open standards such as the OAuth 2.0 authorization protocol. A formal analysis of these protocols must account for malicious websites and common web application vulnerabilities, such as cross-site request forgery and open redirectors. We model several configurations of the OAuth 2.0 protocol in the applied pi-calculus and verify them using ProVerif. Our models rely on WebSpi, a new library for modeling web applications and web-based attackers that is designed to help discover concrete website attacks. Our approach is validated by finding dozens of previously unknown vulnerabilities in popular websites such as Yahoo and WordPress, when they connect to social networks such as Twitter and Facebook. This work was published in CSF'12 [21].

To protect sensitive user data against server-side attacks, a number of security-conscious web applications have turned to client-side encryption, where only encrypted user data is ever stored in the cloud. We formally investigate the security of a number of such applications, including password managers, cloud storage providers, an e-voting website and a conference management system. We show that their security relies on both their use of cryptography and the way it combines with common web security mechanisms as implemented in the browser. We model these applications using the WebSpi web security library for ProVerif, we discuss novel attacks found by automated formal analysis, and we propose robust countermeasures. Some of the attacks we discovered were presented at WOOT'12 [24]. Our formal models and verified countermeasures are going to be presented at POST'13 [20].

6.6. Attacks and Proofs for TLS Implementations

Participants: Alfredo Pironti [correspondant], Karthikeyan Bhargavan, Pierre-Yves Strub, Cedric Fournet, Markulf Kohlweiss.

cryptographic protocol, formal methods, automated verification, traffic analysis, vulnerabilities

TLS is possibly the most used secure communications protocol, with a 18-year history of flaws and fixes, ranging from its protocol logic to its cryptographic design, and from the Internet standard to its diverse implementations. We have been engaged in a long-term project on verifying TLS implementations and this project is now coming to fruition, with a number of papers are now in the pipeline. We list two new results below, both are submitted for review.

Websites commonly use HTTPS to protect their users' private data from network-based attackers. By combining public social network profiles with TLS traffic analysis, we present a new attack that reveals the precise identities of users accessing major websites. As a countermeasure, we propose a novel length-hiding scheme that leverages standard TLS padding to enforce website-specific privacy policies. We present several implementations of this scheme, notably a patch for GnuTLS that offers a rich length-hiding API and an Apache module that uses this API to enforce an anonymity policy for sensitive user files. Our implementations are the first to fully exercise the length-hiding features of TLS and our work uncovers hidden timing assumptions in recent formal proofs of these features. Compared to previous work, we offer the first countermeasure that is standards-based, provably secure, and experimentally effective, yet pragmatic, offering websites a precise trade-off between user privacy and bandwidth efficiency. This work is available as an Inria technical report [36].

We develop a verified reference implementation of TLS 1.2. Our code fully supports its wire formats, ciphersuites, sessions and connections, re-handshakes and resumptions, alerts and errors, and data fragmentation, as prescribed in the RFCs; it interoperates with mainstream web browsers and servers. At the same time, our code is carefully structured to enable its modular, automated verification, from its main API down to computational assumptions on its cryptographic algorithms. Our implementation is written in F# and specified in F7. We present security specifications for its main components, such as authenticated stream encryption for the record layer and key establishment for the handshake. We describe their verification using the F7 refinement typechecker. To this end, we equip each cryptographic primitive and construction of TLS with a new typed interface that captures its security prop- erties, and we gradually replace concrete implementations

with ideal functionalities. We finally typecheck the protocol state machine, and thus obtain precise security theorems for TLS, as it is implemented and deployed. We also revisit classic attacks and report a few new ones. This work is under review and will be released as an Inria technical report in January 2013.

SECSI Project-Team

6. New Results

6.1. Dishonest keys (Objective 2)

Participants: Hubert Comon-Lundh, Guillaume Scerri.

One of the main issues in the formal verification of the security protocols is the validity (and scope) of the formal model. Otherwise, it may happen that a protocol is proved and later someone finds an attack. This paradoxical situation may happen when the formal model used in the proof is too abstract.

A main stream of research therefore consists in proving full abstraction results (also called *soundness*): if the protocol is secure in the (symbolic) model, then an attack can only occur with negligible probability in a computational model. Such results have two main drawbacks: first they are very complicated, and have to be completed again and again for each combination of security primitives. Second, they require strong hypotheses on the primitives, some of which are not realistic. For instance, it is assumed that the attacker cannot forge his own keys (or that all keys come with their certificates, even for symmetric encryption keys).

Hubert Comon-Lundh, Véronique Cortier and Guillaume Scerri [31] propose an extension of the symbolic model, and prove it computationally sound, without this restriction on the dishonest keys.

6.2. Unconditional Soundness (Objective 2)

Participant: Hubert Comon-Lundh.

Hubert Comon-Lundh, Véronique Cortier and Guillaume Scerri [31] show how one can drop one of the assumptions of computational soundness results. However, the proofs remain very complicated and there are still assumptions such as the absence of key cycles, or no dynamic corruption... that are still necessary for all these results.

Gergei Bana and Hubert Comon-Lundh investigated a completely different approach to formal security proofs [25], which does not make any such assumptions. The idea can be stated in a nutshell: whereas all existing formal models state the attacker's abilities, they propose to formally state what the attacker *cannot* do.

This makes a big difference, since the soundness need only to be proved formula by formula and only the very necessary assumptions are used for such formulas (for instance, no absence of key cycles is needed). This does not need to be proved again when a primitive is added.

The counterpart of this nice approach is the difficulty of the automation: a tool is required for checking the consistency of a set of axioms, together with the conditions accumulated along a trace. This problem is the subject of research for the next year(s).

6.3. QRB-Domains (Objective 4)

Participant: Jean Goubault-Larrecq [correspondant].

One of the outstanding problems that remains in the denotational semantics of higher-order programming languages with probabilistic choice is the existence of a suitable, convenient category of domains for defining the denotations of types. Technically, a category of so-called continuous domains is sought after, which would be Cartesian-closed and stable by the action of the probabilistic powerdomain functor. This is not known to exist, and is part of the Jung-Tix conjecture. Jean Goubault-Larrecq found out that relaxing continuity to quasi-continuity helped gaining stability by the action of the probabilistic powerdomain functor [20]. This is an extended version of previous work published at the LICS'10 conference.

6.4. Complete WSTS

Participant: Jean Goubault-Larrecq [correspondant].

Well-structured transition systems form a large class of infinite-state transition systems on which one can decide coverability (a slightly relaxed form of reachability). These include Petri nets, lossy channel systems, and various process algebras.

With Alain Finkel, Jean Goubault-Larrecq developed a theory of *complete* well-structured transition systems, allowing one to generalize Karp and Miller's coverability tree construction for Petri nets to all well-structured transition systems. This work culminated in [19], following two conference papers (STACS'09, ICALP'09). The general theory was the topic of the invited talk [34].

6.5. Static Analysis of Programs with Imprecise Probabilities

Participant: Jean Goubault-Larrecq [correspondant].

Static analyses allows one to obtain guarantees about the behavior of programs, without running them. Programs that handle numerical data such as feedback control loops pose a challenge in this area. This gets even harder when one considers programs that read numerical data from sensors, and write to actuators, as these data are imprecise, and are governed by probability distributions that may themselves be unknown, and only know to fall into some interval of distributions. As part of the ANR projet blanc CPP, an efficient static analysis framework that deals with this kind of programs was proposed [16], based on P-boxes and Dempster-Shafer structures to handle imprecise probabilities. This is based on work first presented at the SCAN'11 conference.

6.6. New Attacks on RSA PKCS#1v1.5 (Objective 2)

Participants: Graham Steel [correspondant], Romain Bardou.

RSA PKCS#1v1.5 is the most commonly used standard for public key encryption, used for example in TLS/SSL. It has been known to be vulnerable to a so-called padding-oracle attack since 1998 when Bleichenbacher described the vulnerability at CRYPTO. The attack, known was the "million message attack" was not thought to present a practical threat, due in part to the large number of oracle messages required. In a paper published at CRYPTO 2012 [26] we gave original modifications showing how the attack can be completed in a median of just 15 000 messages. The results led to widespread interest, indicated by over 1400 downloads of the long version of the paper from the HAL webpage and articles in the New York Times, Boston Globe and Süddeutscher Zeitung.

6.7. Deciding trace equivalence (Objectives 1, 3)

Participants: Vincent Cheval, Hubert Comon-Lundh, Stéphanie Delaune, Rémy Chrétien.

Most existing results focus on trace properties like secrecy or authentication. There are however several security properties, which cannot be defined (or cannot be naturally defined) as trace properties and require the notion of indistinguishably. Typical examples are anonymity, privacy related properties or statements closer to security properties used in cryptography.

In the framework of the applied pi-calculus [44], as in similar languages based on equational logics, indistinguishability corresponds to a relation called trace equivalence. Roughly, two processes are trace equivalent when an observer cannot see any difference between the two processes. Static equivalence applies only to observations on finite sets of messages, and does not take into account the dynamic behavior of a process, whereas trace equivalence is more general and takes into account this aspect.

6.7.1. Static equivalence.

As explained above, static equivalence is a cornerstone to provide decision procedures for observational equivalence.

Stéphanie Delaune, in collaboration with Mathieur Baudet and Véronique Cortier, has designed a generic procedure for deducibility and static equivalence that takes as input any convergent rewrite system [15]. They have shown that their algorithm covers most of the existing decision procedures for convergent theories. They also provide an efficient implementation, and compare it briefly with the tools ProVerif and KiSs. This paper is a journal version of the work presented in [47].

In [17], Ştefan Ciobâcă, Stéphanie Delaune and Steve Kremer propose a representation of deducible terms to overcome the limitation of the procedure mentionned above. This new procedure terminates on a wide range of equational theories. In particular, they obtain a new decidability result for the theory of trapdoor bit commitment encountered when studying electronic voting protocols. The algorithm has been implemented in the KiSs tool. This paper is a journal version of the work presented in [49].

In [18], Stéphanie Delaune, in collaboration with Véronique Cortier (LORIA, France), shows that existing decidability results can be easily combined for any disjoint equational theories: if the deducibility and indistinguishability relations are decidable for two disjoint theories, they are also decidable for their union. They also propose a general setting for solving deducibility and indistinguishability for an important class (called *monoidal*) of equational theories involving AC operators. This paper is a journal version of the works presented in [45], [50].

6.7.2. Trace equivalence.

When processes under study do not contain replication, trace equivalence can be reduced to the problem of deciding symbolic equivalence, an equivalence relation introduced by M. Baudet [46].

Stéphanie Delaune, Steve Kremer, and Daniel Pasaila study this symbolic equivalence problem when cryptographic primitives are modeled using a group equational theory, a special case of monoidal equational theories. The results strongly rely on the correspondance between group theories and rings. This allows them to reduce the problem under study to the problem of solving systems of equations over rings. This result was published at IJCAR'12 [33],

When processes under study contain replication, the approach relying on symbolic equivalence does not work anymore. Moreover, since it is well-known that deciding reachability properties is undecidable under various restrictions, there is actually no hope to do better for equivalence-based properties. Rémy Chrétien, Véronique Cortier, and Stéphanie Delaune provide the first results of (un)decidability for certain classes of protocols for the equivalence problem. They consider a class of protocols shown to be decidable for reachability properties, and establish a first undecidability result. Then, they restrained the class of protocols a step further by making the protocols deterministic in some sense and preventing it from disclosing secret keys. This tighter class of protocols was then shown to be decidable after reduction to an equivalence between deterministic pushdown automata (see [42])

To deal with replication, another approach was studied by Vincent Cheval in collaboration with Bruno Blanchet. They propose an extension of the automatic protocol verifier ProVerif. ProVerif can prove observational equivalence between processes that have the same structure but differ by the messages they contain. In order to extend the class of equivalences that ProVerif handles, they extend the language of terms by defining more functions (destructors) by rewrite rules. These extensions have been implemented in ProVerif and allow one to automatically prove anonymity in the private authentication protocol by Abadi and Fournet. This work is currently under submission [40].

6.8. Mobile ad-hoc networks (Objectives 1, 3)

Participants: Rémy Chrétien, Stéphanie Delaune, Graham Steel.

Mobile ad hoc networks consist of mobile wireless devices which autonomously organize their communication infrastructure: each node provides the function of a router and relays packets on paths to other nodes. Finding these paths in an a priori unknown and constantly changing network topology is a crucial functionality of any ad hoc network. Specific protocols, called *routing protocols*, are designed to ensure this functionality known as *route discovery*. Secured versions of routing protocols have been proposed to provide more guarantees on the resulting routes, and some of them have been designed to protect the privacy of the users.

However, existing results and tools do not apply to routing protocols. This is due in particular to the fact that all possible topologies (infinitely many) have to be considered. Véronique Cortier, Jan Degrieck, and Stéphanie Delaune propose a simple reduction result: when looking for attacks on properties such as the validity of the route, it is sufficient to consider topologies with only four nodes, resulting in a number of just five distinct topologies to consider. As an application, several routing protocols, such as the SRP applied to DSR and the SDMSR protocols, have been analysed using the ProVerif tool. This work was published at POST'12 [32].

Rémy Chrétien and Stéphanie Delaune propose a framework for analysing privacy-type properties for routing protocols. They use the notion of equivalence between traces to formalise three security properties related to privacy, namely indistinguishability, unlinkability, and anonymity. They study the relationship between these definitions and we illustrate them using two versions of the ANODR routing protocol. This work is currently under submission [43].

In the context of vehicular ad-hoc networks, to improve road safety, a vehicle-to-vehicle communication platform is currently being developed by consortia of car manufacturers and legislators. In [51], Morten Dahl, Stéphanie Delaune and Graham Steel propose a framework for formal analysis of privacy in location based services such as anonymous electronic toll collection. They give a formal definition of privacy, and apply it to the VPriv scheme for vehicular services. They analyse the resulting model using the ProVerif tool, concluding that the privacy property holds only if certain conditions are met by the implementation. Their analysis includes some novel features such as the formal modelling of privacy for a protocol that relies on interactive zero-knowledge proofs of knowledge and list permutations.

6.9. Composition results (Objective 1)

Participants: Vincent Cheval, Stéphanie Delaune.

Formal methods have proved their usefulness for analysing the security of protocols. However, protocols are often analysed in isolation, and this is well-known to be not sufficient as soon as the protocols share some keys. Nowdays, several composition results exist for trace-based properties, but there is a lack of composition results for equivalence-based properties.

Myrto Arapinis, Vincent Cheval, and Stéphanie Delaune study the notion of trace equivalence and we show how to establish such an equivalence relation in a modular way. They show that composition works even when the processes share secrets provided that they satisfy some reasonable conditions. Their composition result allows one to prove various equivalence-based properties in a modular way, and works in a quite general setting. In particular, they consider arbitrary cryptographic primitives and processes that use non-trivial else branches. As an example, they consider the ICAO e-passport standard, and they show how the privacy guarantees of the whole application can be derived from the privacy guarantees of its sub-protocols. This work was published at CSF'12 [22].

TASC Project-Team

6. New Results

6.1. Constraint and Abstract Interpretation

Participants: Marie Pelleau, Charlotte Truchet, Frédéric Benhamou, Antoine Miné.

We apply techniques from Abstract Interpretation (AI), a general theory of semantic abstractions, to Constraint Programming (CP), which aims at solving hard combinatorial problems with a generic framework based on first-order logics. We highlight some links and differences between these fields: both compute fix-points by iteration but employ different extrapolation and refinement strategies; moreover, consistencies in Constraint Programming can be mapped to non-relational abstract domains.

- In a first step, we redefine all the components of CP on abstract domains, instead of the usual cartesian, domain-specific domains (boxes or integer sets), obtaining a generic method that can be specified for any of the AI abstract domains.
- In a second step, we then use the correspondences between AI and CP to build an abstract constraint solver that leverages abstract interpretation techniques (such as relational domains) to go beyond classic solvers. We present encouraging experimental results obtained with our prototype implementation, called AbSolute. In particular, AbSolute is able to solve problems on both discrete and continuous variables.

The work is done in collaboration with Antoine Miné.

A corresponding paper *A constraint solver based on abstract domains* [26] will appear at the 14th International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI'13).

6.2. Analytic Combinatorics and Lazy Filtering

Participants: Jérémie du Boisberranger, Danièle Gardy, Xavier Lorca, Charlotte Truchet.

The ANR Boole project (2009-2013) aims at quantifying different formats of boolean formulas, including SAT of constraints. Within the project, we have started a collaboration with Danièle Gardy, UVSQ, expert in analytic combinatorics and average-case study of algorithms. The goal of the collaboration was to quantify, within a high level probabilistic model, how often the bound-consistency propagator of an *alldifferent* constraint is likely to do something (or nothing). During year 2012, a particular focus has been put on calculating the probabilistic indicator, with an accepted publication at Analco 2013 (to appear). Further research include implementing and testing different possible uses for this indicator. A post-doc, Vincent Armant, has been recruited on the Boole project for this.

The corresponding paper *When is it worthwhile to propagate a constraint? A probabilistic analysis of* alldifferent [29] was accepted for publication at the ANALCO 13th Meeting on Analytic Algorithmics and Combinatorics (Analco 2013).

6.3. Learning Constraint Models

Participants: Nicolas Beldiceanu, Naina Razakarison, Helmut Simonis.

We designed a system which generates finite domain constraint models from positive example solutions, for highly structured problems. The system is based on the global constraint catalog, providing the library of constraints that can be used in modeling, and the constraint seeker tool, which finds a ranked list of matching constraints given one or more sample call patterns. We have tested the modeler with 230 examples, ranging from 4 to 6,500 variables, using between 1 and 7,000 samples. These examples come from a variety of domains, including puzzles, sports-scheduling, packing and placement, and design theory. Surprisingly, in many cases the system finds usable candidate lists even when working with *a single*, positive example.

The corresponding paper *A Model Seeker: Extracting Global Constraint Models From Positive Examples* [19] was published at the 18th International Conference on Principles and Practice of Constraint Programming (CP 2012).

6.4. Scalable Resource Scheduling Constraints

Participants: Nicolas Beldiceanu, Mats Carlsson, Arnaud Letort.

Following up on our work on scalable placement constraints for rectangle and box packing, and initially motivated by multidimensional bin packing problems that arise in the context of data centers, we have focussed this year our work on scalable resource scheduling constraints.

First we came up with a sweep based algorithm for the *cumulative* constraint, which can operate in filtering mode as well as in greedy assignment mode. Given n tasks, this algorithm has a worst-case time complexity of $O(n^2)$. In practice, we use a variant with better average-case complexity but worst-case complexity of $O(n^2 \log n)$, which goes down to $O(n \log n)$ when all tasks have unit duration, i.e. in the bin-packing case. Despite its worst-case time complexity, this algorithm scales well in practice, even when a significant number of tasks can be scheduled in parallel. It handles up to 1 million tasks in one single cumulative constraint in both CHOCO and SICStus.

Second we generalize the previous sweep algorithm to directly handle multiple resources. Given n tasks and k resources, this algorithm has a worst-case time complexity of $O(k \cdot n^2)$ but scales well in practice. In greedy assignment mode, it handles up to 1 million tasks with 64 resources in one single constraint. In filtering mode, on our benchmarks, it yields a speed-up of about $k^{0.75}$ when compared to its decomposition into k independent *cumulative* constraints.

A first paper *A Scalable Sweep Algorithm for the cumulative Constraint* [24] was published at the 18th International Conference on Principles and Practice of Constraint Programming (CP 2012). A second paper *A Synchronized Sweep Algorithm for the k-dimensional cumulative Constraint* was accepted for publication at the 10th International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR 2013).

6.5. Reification of Global Constraints

Participants: Nicolas Beldiceanu, Mats Carlsson, Pierre Flener, Justin Pearson.

Being able expressing the negation of global constraints is something that is required in contexts such as testing the equivalence of two constraints models (see the PhD thesis of N. Lazaar) or in the context of learning constraints. Motivated by that, we introduce a simple idea for deriving reified global constraints in a systematic way. It is based on the observation that most global constraints can be reformulated as a conjunction of total function constraints together with a constraint that can be easily reified.

The corresponding paper *On the Reification of Global Constraints* [12] was published in the Constraints journal. A companion technical report [35] provides such reifications for 82% of the constraints of the global constraint catalog [36].

6.6. Optimization and Soft Problems

Participant: Thierry Petit.

Many optimization problems involve business constraints, which are complementary to an objective function that aggregates cost variables. These constraints involve the same cost variables. They are generally non linear. In the literature, several approaches were proposed for balancing constraints. We address the reverse concept, that is, concentrating high cost values in a restricted number of areas. This concept is motivated by several concrete examples, such as resource constrained scheduling problems with machine rentals. We present a new global constraint called *focus*. We provide a complete and optimum time complexity filtering algorithm for our constraint.

The corresponding paper *Focus : A Constraint for Concentrating High Costs* [27] was published at the 18th International Conference on Principles and Practice of Constraint Programming (CP 2012).

6.7. Consistency and Filtering

Participants: Nicolas Beldiceanu, Mats Carlsson, Gilles Chabert, Sophie Demassey, Thierry Petit, Jean-Charles Régin.

Following up on our work on efficient filtering algorithms for common conjunctions of widely used constraints (e.g., *among*, *alldifferent*, *linear constraint*, *inequalities constraints*) we provide:

1. An $O(n \log n)$ bound consistency filtering algorithm for the conjunction of an *alldifferent* and a *linear inequality* constraint. The $O(n \log n)$ complexity is equal to the complexity of the bound consistency algorithm of the *alldifferent* constraint.
2. A polynomial time bound consistency algorithm for the conjunction of *among* constraints where the variable and value domains are interval.

Motivated by the need to define more formally incomplete filtering algorithms we have proposed a new theoretical scheme for characterizing, comparing and classifying the intermediary levels of consistency of global constraints.

The corresponding papers, *An $O(n \log n)$ Bound Consistency Algorithm for the Conjunction of an alldifferent and an Inequality between a Sum of Variables and a Constant, and its Generalization* [17], *The Conjunction of Interval* among *Constraints* [21] and *Intermediary Local Consistencies* [28] were published at the 20th European Conference on Artificial Intelligence (ECAI 2012) as well as at the 9th International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR 2012).

6.8. Automata and Matrix Models

Participants: Nicolas Beldiceanu, Mats Carlsson, Pierre Flener, Justin Pearson.

Matrix models are ubiquitous for constraint problems. Many such problems have a matrix of variables $\mathcal{M}$, with the same constraint C defined by a finite-state automaton $\mathcal{A}$ on each row of $\mathcal{M}$ and a global cardinality constraint gcc on each column of $\mathcal{M}$. We give two methods for deriving, by double counting, necessary conditions on the cardinality variables of the gcc constraints from the automaton $\mathcal{A}$. The first method yields linear necessary conditions and simple arithmetic constraints. The second method introduces the *cardinality automaton*, which abstracts the overall behaviour of all the row automata and can be encoded by a set of linear constraints. We also provide a arc-consistency filtering algorithm for the conjunction of lexicographic ordering constraints between adjacent rows of $\mathcal{M}$ and (possibly different) automaton constraints on the rows. We evaluate the impact of our methods in terms of runtime and search effort on a large set of nurse rostering problem instances.

The corresponding paper *On Matrices, Automata, and Double Counting in Constraint Programming* [11] was published in the Constraints journal.

6.9. Parallelization

Participants: Salvador Abreu, Yves Caniou, Philippe Codognet, Daniel Diaz, Florian Richoux.

During these last decades, many sequential algorithms for Constraint Satisfaction Problems (CSP) have been developed to be able to solve real problems from industry. However these problems become more and more complex and it remains important to treat them as fast as possible. Until the mid-2000's, one developed computers power by increasing CPU frequency. Nevertheless for about five years, this solution is not possible anymore since it asks too much energy (problem linked to heat dissipation issues), thus our machines architecture turns to be more and more multi-core oriented.

Nowadays we still have very few algorithms for constraint problems adapted to multi-core architecture. This year, we obtained very good results with the parallelization of meta-heuristic methods, reaching linear speed-ups over 8,192 cores on the *Costas Array Problem* [22], [23]. We also proposed in [20] two ways to perform smart cooperations between parallel local search processes, leading to very promising new approaches to solve constraint-based problems in parallel.

TOCCATA Team

6. New Results

6.1. Proofs of (Imperative) Programs

- A. Charguéraud has extended his ICFP'11 paper [70] into a journal paper, which is currently under review. This paper describes in more details the theory of characteristic formulae and the tool *CFML*, which supports the verification of *OCaml* programs through interactive *Coq* proofs.
- J.-C. Filliâtre has verified a two lines C program (solving the N-queens puzzle) using *Why3*. This case study has been presented at VSTTE 2012 [27].
- With M .Pereira and S. Melo de Sousa (Universidade da Beira Interior, Covilhã, Portugal), J.-C. Filliâtre developed an environment for proving ARM assembly code. It uses *Why3* as an intermediate VC generator. It was presented at the Inforum conference [34] (best student paper).
- F. Bobot and J.-C. Filliâtre have presented the notion of separation predicates introduced in the PhD of F. Bobot (defended December 2011) at ICFEM 2012 [21].
- S. Conchon and A. Mesbout, in collaboration with F. Zaïdi (Fortesse team, LRI) and A. Goel and S. Krstić (Strategic Cad Labs, INTEL), have presented a tool paper about the Cubicle model checker at CAV 2012 [24]. A more detailed description of the main algorithms implemented in Cubicle will be presented during the JFLA 2013 [73].
- A significant effort was dedicated to the development of *Why3*, with 3 public releases [39], [40], [41]. Associated with this activity, we actively participe to the new trend (that emerged in 2010-2011) of construction of international program verification benchmarks and organization of program verification competitions. We participated to the joint paper that reports on the first FoVeOOS competition [23] (http://proval.lri.fr/gallery/cost11comp.en.html). J.-C. Filliâtre and A. Paskevich wrote a detailed report [33] on the 2nd competition VSTTE competition (https://sites.google.com/site/vstte2012/compet) that they organized, published in the proceedings of the COMPARE workshop. This paper describes the competition, presents the five problems that were proposed to the participants, and gives an overview of the solutions sent by the 29 teams that entered the competition.

  Our own gallery of verified programs (http://toccata.lri.fr/gallery/index.en.html) was augmented significantly, with now approximately 100 examples, classified by topics, tools, etc.

6.2. Floating-Point and Numerical Programs

- The PhD thesis of T. Nguyen was defended in June [12]. It includes an improved version of the former approach [102] that we proposed for proving floating-point programs while taking into account architecture- and compiler-dependent features, such as the use of the x87 stack in Intel micro-processors. The underlying tool analyzes the assembly code generated by the compiler. It also includes a preliminary and independent approach for proving floating-point programs involving bit-level operations.
- C. Lelay, under the supervision of S. Boldo and G. Melquiond, has worked on easing proofs of differentiability and integrability in *Coq*. The use case was the existence of a solution to the wave equation thanks to D'Alembert's formula; the goal was to automate the process as much as possible [30]. While a major improvement with respect to *Coq* standard library, this first approach was not user-friendly enough for parametric intervals. So a different approach based on the pervasive use of total functions has been experimented with [22].
- S. Boldo, F. Clément, J.-C. Filliâtre, M. Mayero, G. Melquiond and P. Weis finished the formal proof of a numerical analysis program: the second order centered finite difference scheme for the one-dimensional acoustic wave [14].

- S. Boldo has developed a formal proof of an algorithm for computing the area of a triangle, an improvement of its error bound and new investigations in case of underflow [60].
- S. Boldo, J.-H. Jourdan, X. Leroy, and G. Melquiond have extended CompCert to get the first formally verified compiler that provably preserves the semantics of floating-point programs [63].
- G. Melquiond has kept improving the floating-point and interval theories used to perform proofs by computations in *Coq* [16].

6.3. Automated Deduction

- In collaboration with Assia Mahboubi (from Typical Inria project-team), and Guillaume Melquiond, the group involved in the development of *Alt-Ergo*, implemented and proved the correctness of a novel decision procedure for quantifier-free linear integer arithmetic [20]. This algorithm tries to bridge the gap between projection and branching/cutting methods: it interleaves an exhaustive search for a model with bounds inference. These bounds are computed provided an oracle capable of finding constant positive linear combinations of affine forms. An efficient oracle based on the Simplex procedure has been designed. Our algorithm is proved sound, complete, and terminating and is implemented in the *Alt-Ergo* theorem prover.
- In their LMCS journal paper [15], S. Conchon, É. Contejean and M. Iguernelala present a modular extension of ground AC-completion for deciding formulas in the combination of the theory of equality with user-defined AC symbols, uninterpreted symbols and an arbitrary signature disjoint Shostak theory X. This paper extends the results presented in [72] by showing that a simple preprocessing step allows to get rid of a full AC-compatible reduction ordering, and to simply use a partial multiset extension of a *non necessarily AC-compatible* ordering.
- In [31], S. Conchon, G. Melquiond and C. Roux described a dedicated procedure for a theory of floating-point numbers which allows reasoning on approximation errors. This procedure is based on the approach of the Gappa tool: it performs saturation of consequences of the axioms, in order to refine bounds on expressions. In addition to the original approach, bounds are further refined by a constraint solver for linear arithmetic. This procedure has been implemented in *Alt-Ergo*.
- In [42], [32], C. Dross and J. Kanig from AdaCore, in collaboration with S. Conchon and A. Paskevich propose a generic framework for adding a decision procedure for a theory or a combination of theories to an SMT prover. This mechanism is based on the notion of instantiation patterns, or *triggers*, which restrict instantiation of universal premises and can effectively prevent a combinatorial explosion. A user provides an axiomatization with triggers, along with a proof of completeness and termination in our framework, and obtains in return a sound, complete and terminating solver for his theory. A prototype implementation was realized in the Alt-Ergo prover. As a case study, a feature-rich axiomatization of doubly-linked lists was proved comlpete and terminating.
- In [38], A. Paskevich in collaboration with J. Blanchette from TU München, introduced a new format in the TPTP family (http://tptp.org), called TFF1, which extends the earlier TFF0 format (many-sorted first-order logic) with rank-1 type polymorphism. The technical report presents the syntax, typing rules, and semantics, as well as a sound and complete translation from TFF1 to TFF0. The format is designed to be easy to process by existing reasoning tools that support ML-style polymorphism. It opens the door to useful middleware, such as monomorphizers and other translation tools that encode polymorphism in FOF or TFF0. Ultimately, the hope is that TFF1 will be implemented in popular automatic theorem provers.
- A. Paskevich and J.-C. Filliâtre implemented a new *Coq* tactic that is able call an automated prover from *Coq* environment. It uses *Why3* as an intermediate tool. This new tactic brings a very significant improvement of proof automation within *Coq*. For example, the development of a certified VC generator in *Why3* made an intensive use of this tactic. The combination of automatic and interactive theorem proving was the subject of invited talks given by J.-C. Filliâtre at the workshop "Automation in Proof Assistants" [17] (satellite workshop of ETAPS 2012) and at the international workshop on Intermediate Verification Languages [18] (BOOGIE 2012, Berkeley, California, USA, July 2012).

- Together with O. Hermant (ISEP, Paris), D. Cousineau studied the cut elimination property for deduction modulo theories. They were able to show a strong relationship the syntactic cut-elimination property and the semantic construction of pre-models: they made a full semantic proof that the existence of a pre-model entails the cut elimination property for the considered theory in deduction modulo. This is published at the RTA Conference [26].
- *TLA+* is a specification language based on standard set theory and temporal logic, developed by the TLA groupe of Microsoft Research (http://research.microsoft.com/en-us/um/people/lamport/tla/tla.html). During the first part of his post-doc, D. Cousineau finalized a work on describing how to write *TLA+* proofs and check them with *TLAPS*, the *TLA+* Proof System. It was published as a tool description at FM Conference [25].
- S. Conchon defended his *habilitation à diriger des recherches* in December 2012. The memoir [11] provides a very good and useful survey of the scientific work of the past 10 years, around the SMT solving techniques, that leaded to the tools *Alt-Ergo* and Cubicle as they are nowadays.

6.4. Certification

- P. Herms, together with C. Marché and B. Monate (CEA List), developed a certified VC generator, using Coq. The program for VC calculus and its specifications are both written in Coq, but the code is crafted so that it can be extracted automatically into a stand-alone executable. It is also designed in a way that allows the use of arbitrary first-order theorem provers to discharge the generated obligations [28].
- On top of the previous generic VC generator, P. Herms developed a certified VC generator for C source code annotated using ACSL. This work is the main result of his PhD thesis which will be defended in January 2013.
- A. Tafat and C. Marché started experiments of development of a certified VC generator using Whyt instead of Coq. The challenge was to formalize the operational semantics of an imperative language, and a corresponding weakest precondition calculus, without the possibility to use Coq advanced features such as dependent types nor higher-order functions. The classical issues with local bindings, names and substitutions were solved by identifying appropriate lemmas. It was shown that *Why3* can offer a very significantly higher amount of proof automation compared to Coq [43]. This will be presented at the JFLA conference in February 2013 [95]
- The work that we started in 2011, about the use of the *Why3* environment and its back-end provers as an alternative to the built-in prover of "Atelier B", was published at the ABZ conference [29]. This work continues in the context of the new ANR project BWare.
- With J. Almeida, M. Barbosa, J. Pinto and B. Vieira (University do Minho, Braga, Portugal), J.-C. Filliâtre developed a method for certifying programs involving cryptographic methods. It uses *Why* as an intermediate language. A journal article will appear on *Science of Computer Programming* [13].
- Watermarking techniques are used to help identify copies of publicly released information. They consist in applying a slight and secret modification to the data before its release, in a way that should remain recognizable even in (reasonably) modified copies of the data. Using the *Coq*ALEA library, which formalizes probability theory and probabilistic programs, D. Baelde together with P. Courtieu, D. Gross-Amblard from Rennes and C. Paulin have established new results about the robustness of watermarking schemes against arbitrary attackers. The technique for proving robustness is adapted from methods commonly used for cryptographic protocols and our work illustrates the strengths and particularities of the induced style of reasoning about probabilistic programs. This work has been presented at the conference ITP 2012 [19].
- Supervised by J. Falcou and C. Paulin during his M2 internship, N. Lupinski developed a formalisation of a skeleton language for automated generation of parallel programs. A kernel of the language has been identified, its semantics has been formalised in *Coq* where a construction is interpreted by a

relation between lists of entries and lists of outputs. A transformation scheme from the skeleton language towards JOCaml programs has been proposed and proven correct with respect to the relational semantics. This work is described in [44].

- A. Charguéraud is currently working on the JsCert project (http://jscert.org), which aims at the formalization of the semantics of the JavaScript programming language (as described in *ECMAScript Language Specification, version 5.1*) and the development of a verified JavaScript interpreter. This project is joint work with Philippa Gardner, Sergio Maffeis, Gareth Smith, Daniele Filaretti and Daiva Naudziuniene from Imperial College, and Alan Schmitt and Martin Bodin from Inria Rennes - Bretagne Atlantique. As of today, the formalization already covers a substantial amount of the JavaScript language, and the verified interpreter is able to execute a number of benchmarks taken from standard JavaScript test suites.

  The formalization of the semantics of JavaScript makes use of a novel technique, called *pretty-big-step semantics*, for representing reduction rules in big-step style without suffering from a duplication of several premises accross different rules. This duplication is indeed typical in big-step semantics describing the behavior of exceptions and of divergence. The pretty-big-step semantics is described by A. Charguéraud in a paper to appear at ESOP 2013 [71].

TYPICAL Project-Team

5. New Results

5.1. Feit-Thompson

The Feit-Thompson is an important theorem stating that every finite group of odd order is solvable. It is an important step in the classification of finite groups. Its proof is remarkable through its difficulty and its length (more than 1000 pages of dense mathematical text).

This proof was entirely formalized in Coq. This effort was started six years ago, as a joint project of the project teams Typical, Marelle (Sophia-Antipolis) and the Inria-MSR joint center, under the supervision of Georges Gonthier. The proof was finished in september 2012 and is considered a remarkable achievement. It also gave birth to several side products, such as enhancements of the SSReflect proof language. For Typical, Assia Mahboubi, Enrico Tassi and Cyril Cohen were instrumental in this effort.

5.2. Formal Semantics of Type Theory

Bruno Barras finished an extensive formalization of Coq's type theory in Coq, as well as a large formalization of set theory. This work includes several new results and insights in the study of Type Theory and is the body of Barras' habilitation thesis to be defended early in 2013.

5.3. Study of Type Theories

Bruno Barras finished an extensive formalization of Coq's type theory in Coq, as well as a large formalization of set theory. This work includes several new results and insights in the study of Type Theory and is the body of Barras' habilitation thesis to be defended early in 2013.

Chantal Keller, with Marc Lasson, has presented a notion of parametricity in impredicative type theories, which yields some possible application in proof search [18].

5.4. Formal and computable algebra

Cyril Cohen and Assia Mahboubi have worked on representing various algebraic objects in Coq, in a way that allows computation. In particular, Cohen proposed and developed a representation of algebraic numbers in Coq, as presented in [16]. Assia Mahboubi has collaborated with Frédéric Chyzak (Inria Paris-Roquencourt, Algo team) on the certification of algorithms for D-finite objects.

5.5. Certifiable real optimization

Under the joint supervision of Stéphane Gaubert and Benjamin Werner, with Xavier Allamigeon, Victor Magron is investigating ways to check difficult real inequalities, over bounded domains, in ways which can be re-checked by proof systems like Coq. One such algorithm, combining convex optimization and Max-plus techniques is submitted for publication at ECC 2013.

5.6. Binder representation in Coq

Benjamin Werner has developed a generic tree datatype in Coq, which can encode any language with fixed-arity operators with binders. The application towards smoother formal treatment of such languages is still in progress.

5.7. SMT and Coq

Chantal Keller has enhanced the performances of her SMT-Coq interface based automatic tactic. More precisely, the code has been made more modular which allowed:

- A first interfacing with the renowned Z3 SMT prover from Microsoft Research,
- Extending SMT-Coq to the theory of Coq's native 31 bits integers.

5.8. Automated decision procedures

Assia Mahboubi has woked with members of the Proval team on a new decision procedure for integer arithmetics now intergrated in the Alt-Ergo SMT solver. Assia Mahboubi has worked with Stéphane Lengrand and Mahfuza Farooque on the design of a sequent calculus with focussing and on the conception of a proof search strategy in this calculus which simulates the Davis-Putman-Logemann-Loveland algorithme modulo theory (DPPL(T)) which is implemented by modern SMT-solvers. An implementation developped by Stéphane Lengrand illustrate this approach on standard SMT benchmarks.

VERIDIS Project-Team

6. New Results

6.1. Automated and Interactive Theorem Proving

6.1.1. Combination of decision procedures

Participants: Pascal Fontaine, Simon Halfon, Stephan Merz, Christoph Weidenbach.

SMT solvers, combination, decision procedures, theorem proving

We investigate the theoretical limits of combining decision procedures and reasoners, as these are important for the development of the veriT solver (see section 5.1). It has long been known that it is possible to extend any decidable language (subject to a minor requirement on cardinalities) with predicates described by a Bernays-Schönfinkel-Ramsey theory (BSR). A formula belongs to the BSR decidable fragment if it is a conjunction of universal, function-free formulas. As a consequence of this theoretical result, it is possible to extend a decidable quantifier-free language with sets and set operators, relations, orders and similar concepts. This can be used to significantly extend the expressivity of SMT solvers. In previous work, we generalized this result to the decidable first-order class of monadic predicate logic, and to the two-variable fragment. In subsequent joint work with Carlos Areces from Universidad Nacional de Córdoba, Argentina, we showed that two other important decidable fragments (namely the Ackermann fragment, and several guarded fragments) are also easily combinable. In 2012, we considered, in the same spirit, the combination of theories that are not necessarily decidable [18]. In particular, we considered combinations of decision procedures and refutationally complete semi-decision procedures, as well as black-box combinations of different refutationally complete theorem provers, together with finite model finders. These results in particular yield theoretical foundations for how FOL provers can be combined with SMT techniques in a black-box style of integration.

6.1.2. Using symmetries in SMT

Participants: Pascal Fontaine, Stephan Merz.

theorem proving, SMT solvers, decision procedures, symmetry

Methods exploiting problem symmetries have been very successful in several areas including constraint programming and SAT solving. We proposed similar techniques for enhancing the performance of SMT-solvers by detecting symmetries in the input formulas and using them to prune the search space of the SMT algorithm. These techniques are based on the concept of (syntactic) invariance by permutation of symbols. In 2011, we presented a technique restricted to constants but which exhibited impressive results for some categories of formulas [4]; this technique was quickly implemented in major SMT solvers, including CVC4 and Z3.

In 2012, we designed a more general approach, based on graph isomorphism, for symmetry detection in the SMT context. Experimental analysis indicates that many formulas from the SMT-LIB repository exhibit symmetries that are left unexploited by the previous techniques. Finding new techniques to exploit these is the subject of ongoing work with the University of Cordoba in Argentina; we expect that breaking those symmetries will yield again some significative efficiency improvement.

6.1.3. Encoding TLA+ proof obligations for SMT solvers

Participants: Stephan Merz, Hernán-Pablo Vanzetto.

system verification, SMT solving, TLA

The TLA$^+$ proof system TLAPS (see section 5.2) is being developed within a project at the MSR-Inria Joint Centre to which we contribute. Proof obligations that arise during the verification of typical TLA$^+$ specifications require reasoning about the principal TLA$^+$ data structures such as sets, functions, arithmetic, tuples, and records. None of the backend provers present in the initial versions of TLAPS was able to reason effectively about steps involving several of these features, and in 2011 we started developing an improved backend for translating TLA$^+$ proof obligations to SMT-Lib, the generic input language of SMT solvers. The main challenge was to design a sound translation from untyped TLA$^+$ to the multi-sorted first-order logic that underlies SMT-Lib, and our original proposal was based on deriving type assignments to TLA$^+$ expressions in a custom type system useful for SMT-Lib. This approach sometimes failed to derive types for subexpressions or required stronger typing assumptions than those required by the semantics of untyped TLA$^+$.

In 2012, based on a suggestion by Ken McMillan, we investigated a different approach whose main idea is to embed SMT sorts such as integers in the global universe of TLA$^+$ values, and to axiomatically define operations such as addition or multiplication on the image of that embedding. This approach effectively delegates type inference to the SMT solver and can therefore handle arbitrary TLA$^+$ expressions. However, it generates many quantified background axioms that may render SMT solvers ineffective, and we developed powerful pre-processing techniques for replacing quantified axioms by their required ground instances. The SMT backend in the current release of TLAPS is based on a hybrid approach to translation, where type inference is used whenever possible in order to obtain simpler SMT input. The two translation techniques have been published in 2012 [19], [20], and they have been validated over many case studies in TLAPS. For example, it enables proving the correctness of simple mutual-exclusion algorithms essentially without user interaction, and of the Paxos consensus algorithm in just 130 interactions, whereas a previous proof attempt using the traditional backend provers was unsuccessful.

6.1.4. Compression of SMT proofs

Participants: Pascal Fontaine, Stephan Merz.

theorem proving, SMT solvers, decision procedures, combination of decision procedures

Integrating an SMT solver in a certified environment such as TLAPS or an LF-style proof assistant requires the solver to output proofs. Unfortunately, those proofs may be quite large, and the overhead of rechecking the proof may account for a significant fraction of the proof time. In previous work, we proposed a technique for reducing the size of propositional proofs based on the analysis of resolution graphs, which were justified in an algebra of resolution. Unfortunately, the complexity of these techniques turned out to be prohibitive, but we proposed practical and efficient algorithms for more restricted compression techniques. We continue to develop this line of work with our partners at TU Wien.

6.1.5. Augmenting the Expressiveness of Spass

Participants: Evgeny Kruglov, Arnaud Fietzke, Daniel Wand, Christoph Weidenbach.

automated theorem proving, superposition, linear arithmetic, proof assistants

In 2012 we focused on bridging the gap between the input logic of SPASS and more expressive logics as they are used by systems supporting full-fledged verification such as Isabelle and TLAPS. Main contributions were a specific version of an order-sorted language that can be eventually translated in a many-sorted logic. The latter is implemented in Spass in a prototypic way and first experiments showed significant improvements on proof obligations out of Isabelle/HOL. Actually, the enhancements allowed Spass to become the most powerful automated theorem proving system supporting Isabelle [14]. We are currently working on a coupling with TLAPS (see section 5.2).

A second important branch is the integration of arithmetic into SPASS and the development of the respective hierarchic superposition calculus. In the past [31], [38] we experimented with a black box integration of LP solvers and Z3 to delegate arithmetic reasoning tasks. Now we started our own white box implementation for linear arithmetic and could achieve significant speed-ups. Our own reasoning procedure, dedicated to the specific form of the arithmetic proof obligations generated by SPASS is 50 to 200 times faster than any black box integration [29]. On the calculus side we could prove hierarchic superposition modulo linear arithmetic

to be a decision procedure for the ground case, thus strictly generalizing the DPLL(LA) set up, and to be a decision procedure [39], [40] for timed automata reachability and extensions thereof [17].

6.1.6. Verification of linear hybrid automata

Participant: Uwe Waldmann.

automated theorem proving, superposition, linear arithmetic, proof assistants

We propose an improved symbolic algorithm for the verification of linear hybrid automata with large discrete state spaces. Large discrete state spaces arise naturally in industrial hybrid systems, due to the need to represent discrete inputs, counters, sanity checkbits, possibly multiple concurrent state machines, system-degradation modes, and finite switching variables. To prove safety properties of such systems, it is necessary to combine techniques for analyzing a complex dynamic behaviour with state space exploration methods that can deal with hundreds of discrete variables. In our approach, we represent both the discrete part and the continuous part of the hybrid state space symbolically using a variant of AIGs (And-Inverter-Graphs). Key components of our method are redundancy elimination (to maintain a compact symbolic representation by deleting superfluous linear constraints) and constraint minimization (exploiting the fact that states already reached in previous iterations of the model-checking algorithm can be interpreted as "don't cares" in later steps). A journal article describing the technique appeared in Science of Computer Programming [9].

6.2. Proved development of algorithms and systems

6.2.1. Incremental development of distributed algorithms

Participants: Dominique Méry, Manamiary Andriamiarina.

distributed algorithms, refinement, verification, distributed protocols

The development of distributed algorithms and, more generally, of distributed systems, is a complex, delicate, and challenging process. The approach based on refinement helps to gain formality by using a proof assistant, and proposes to apply a design methodology that starts from the most abstract model and leads, in an incremental way, to the most concrete model, for producing a distributed solution. Our works help to formalize pre-existing algorithms, develop new algorithms, as well as develop models for distributed systems.

Our research was initially (until 2010) carried out within the ANR project RIMEL, in joint work with Mohammed Mosbah and Mohammed Tounsi from the LABRI laboratory, and we are maintaining a joint project B2VISIDIA with LABRI on these topics. More concretely, we aim at an integration of the correct-by-construction refinement-based approach into the *local computation* programming model. The team of LABRI develops an environment called VISIDIA that provides a toolset for developing distributed algorithms expressed as a set of rewriting rules of graph structures. The simulation of rewriting rules is based on synchronization algorithms and we have developed these algorithms by refinement.

More precisely, we show how state-based models can be developed for specific problems and how they can be simply reused by controlling the composition of state-based models through the refinement relationship. Consequently, we obtain a redevelopment of existing distributed algorithms in the *correct-by-construction* approach, and a framework for deriving new distributed algorithms (by integrating models) whose correctness is ensured by construction. Traditionally, distributed algorithms are supposed to run on a fixed network, whereas we consider a network with a changing topology. We have illustrated our methodology with the study of the protocol ANYCAST RP.

The contribution is related to the development of proof-based patterns providing effective help to the developer of formal models of applications, such as dynamic routing or the snapshot problem [13]. In fact, we have developed patterns for simplifying the development of distributed systems using refinement. The applicability of a pattern for routing has been reapplied to the development of a network on chip [12] with our partners of the French-Algerian cooperation described in section 8.3 .

6.2.2. Modeling and verifying the Pastry routing protocol

Participants: Tianxiang Lu, Stephan Merz, Christoph Weidenbach.

distributed hash table, peer-to-peer protocol, Pastry, model checking, theorem proving

As a significant case study for the techniques that we are developing within VeriDis, we are modeling and verifying the routing protocol of the Pastry algorithm [36] for maintaining a distributed hash table in a peer-to-peer network. As part of his PhD work, Tianxiang Lu has developed a TLA$^+$ model of the Pastry routing protocol, which has uncovered several issues in the existing presentations of the protocol in the literature, and in particular a loophole in the join protocol that had been fixed by the algorithm designers in a technical report that appeared after the publication of the original protocol.

As a first step towards proving correctness of the Pastry routing protocol, we identified in 2011 a number of candidate invariants and formally proved in TLAPS (see section 5.2) that these implied the high-level correctness property. In 2012, we consolidated these invariants and proved them correct for our model under the strong assumption that no node ever leaves the network, and the minor assumption that any active node can at any time only allow one new node to join the network. It is still not clear at the moment to which extent nodes can be allowed to leave the network without breaking the virtual ring maintained by Pastry. The invariant proofs contain almost 15000 interactions and constitutes the largest case study carried out so far using TLAPS. We have more recently been able to obtain better automation using the new SMT backend (see section 6.1). The proof was presented at the TLA workshop of FM 2012 [23].

6.2.3. *Verification of distributed algorithms in the Heard-Of model*

Participants: Henri Debrat, Stephan Merz.

theorem proving, distributed algorithms, round-based computation, Byzantine failures

Distributed algorithms are often quite subtle, both in the way they operate and in the assumptions required for their correctness. Formal models are important for unambiguously understanding the hypotheses and the properties of a distributed algorithm. We focus on the verification of round-based algorithms for fault-tolerant distributed systems expressed in the Heard-Of model of Charron-Bost and Schiper [37], and have previously established a reduction theorem that allows to pretend that nodes operate synchronously.

In 2012, we have consolidated our formal proofs in Isabelle/HOL. In particular, we have finished the formal proof of the reduction theorem within Isabelle, produced a generic encoding of the Heard-Of model as a locale in Isabelle/HOL, and used this representation for verifying six different Consensus algorithms: three algorithms tolerating benign failures and three others designed for malicious failures, such as corrupted values. Our Isabelle theories have been published at the Archive of Formal Proofs [27]. The proof of the reduction theorem required formalizing the notion of stuttering invariance, which can be of independent interest and that has also been accepted at the Archive of Formal Proofs [28].

As a significant extension of this work, we have studied the formal verification of probabilistic Consensus algorithms in the Heard-Of model, in particular the Ben-Or algorithm.

6.2.4. *Model checking within SimGrid*

Participants: Marie Duflot-Kremer, Stephan Merz.

model checking, distributed algorithms, message passing, communication primitives, partial-order reduction

For several years we have cooperated with Martin Quinson from the AlGorille project team on adding model checking capabilities to the simulation platform SimGrid for message-passing distributed C programs. The expected benefit of such an integration is that programmers can complement simulation runs by exhaustive state space exploration in order to detect errors such as race conditions that would be hard to reproduce by testing. As part of the thesis work of Cristián Rosa (defended in 2011), a stateless model checker was implemented within the SimGrid platform that can be used to verify safety properties of distributed C programs that communicate by message passing. The ongoing thesis of Marion Guthmuller builds upon this work and aims to extend it for verifying certain liveness properties. This requires rethinking the stateless design, as well as adapting the dynamic partial-order reduction algorithm that is essential to limiting the part of the state space that must actually be explored.

6.2.5. Modeling Medical Devices

Participant: Dominique Méry.

Formal modelling techniques and tools have attained sufficient maturity for formalizing highly critical systems in view of improving their quality and reliability, and the development of such methods has attracted the interest of industrial partners and academic research institutions. Building high quality and zero-defect medical software-based devices is a particular domain where formal modelling techniques can be applied effectively. In [21], we present a methodology for developing critical systems from requirement analysis to automatic code generation based on a standard safety assessment approach. This methodology combines refinement, proof, model checking, and animation, and ultimately can automatically generate source code. This approach is intended to contribute to further the use of formal techniques for developing critical systems with high integrity and to verify complex properties. An assessment of the proposed methodology is given through developing a standard case study: the cardiac pacemaker.

Medical devices are very prone to showing unexpected system behaviour in operation when traditional methods are used for system testing. Device-related problems have been responsible for a large number of serious injuries. Officials of the US Food and Drug Administration (FDA) found that many deaths and injuries related to these devices are caused by flaws in product design and engineering. Cardiac pacemakers and implantable cardioverter-defibrillators (ICDs) are among the most critical medical devices and require closed-loop modelling (integrated system and environment modelling) for verification purposes before obtaining a certificate from the certification bodies. In [24] we present a methodology for modelling a biological system, such as the heart. The heart model is based mainly on electrocardiography analysis, which provides a model at the cellular level. Combining this environment model with a formal model of the pacemaker, we obtain a closed-loop model over which the overall correctness can be verified.

Clinical guidelines systematically assist practitioners in providing appropriate health care in specific clinical circumstances. Today, a significant number of guidelines and protocols are lacking in quality. Indeed, ambiguity and incompleteness are likely anomalies in medical practice. In [25] we use the Event-B modeling language to represent guidelines for subsequent validation. Our main contributions are: to apply mathematical formal techniques to evaluate real-life medical protocols for quality improvement, to derive verification proofs for the protocol and properties according to medical experts, and to publicize the potential of this approach. An assessment of the proposed approach is given through a case study, relative to a real-life reference protocol concerning ECG interpretation, for which we uncovered several anomalies.

Finally, we propose a refinement-based methodology [10] for complex medical systems design, which possesses the required key features. A refinement-based combined approach of formal verification, model validation using a model-checker and refinement chart is proposed in this methodology for designing a high-confidence medical device. Furthermore, we show the effectiveness of this methodology for the design of a cardiac pacemaker system.

6.2.6. Fundamentals of Network Calculus in Isabelle/HOL

Participant: Stephan Merz.

networked systems, min-plus algebra, formal proof

The design of networked and embedded systems has traditionally been accompanied by formal methods for design and analysis. Network Calculus [42] is a well-established theory, based on the $(\min, +)$ dioid, that is designed for computing delay and memory bounds in networks. The theory is supported by several commercial and open-source tools and has been used in major industrial applications, such as the design and certification of the Airbus A380 AFDX backbone. Nevertheless, it is difficult for certification authorities to assess the correctness of the computations carried out by the tools supporting Network Calculus, and we propose the use of *result certification* techniques for increasing the confidence in the Network Calculus toolchain. In joint work with Marc Boyer from ONERA in Toulouse, and with Loïc Fejoz and Nicolas Navet from the RealTime at Work (RTaW) company, we have supervised the master thesis of Etienne Mabille to evaluate the feasibility of the approach. Parts of the theory underlying Network Calculus were formalized in the proof

assistant Isabelle/HOL, and this encoding was used to formally derive theorems that underly the computation of bounds in network servers. The Network Calculus tool produced by RTaW was instrumented to generate traces of its computation, and the correctness of simple systems could in this way be certified by Isabelle. A publication of this work is in preparation, and we intend to continue and extend it in a future joint project.

6.2.7. Bounding message length in attacks against security protocols

Participant: Marie Duflot-Kremer.

security protocols, verification

Security protocols are short programs that describe communication between two or more parties in order to achieve security goals. Despite the apparent simplicity of such protocols, their verification is a difficult problem and has been shown to be undecidable in general. This undecidability comes from the fact that the set of executions to be considered is of infinite depth (an infinite number of protocol sessions can be run) and infinitely branching (the intruder can generate an unbounded number of distinct messages). Several attempts have been made to tackle each of these sources of undecidability. Together with Myrto Arapinis, we have shown [32] that, under a syntactic and reasonable condition of "well-formedness" on the protocol, we can get rid of the infinitely branching part. Following this conference publication, we are preparing a journal version of this result extending the set of security properties to which the result is applicable, in particular including authentication properties.

6.2.8. Evaluating and verifying probabilistic systems

Participant: Marie Duflot-Kremer.

verification, probabilistic systems, performance evaluation

Since its introduction in the 1980s, model checking has become a prominent technique for the verification of complex systems. The aim was to decide whether or not a system was fulfilling its specification. With the rise of probabilistic systems, new techniques have been designed to verify this new type of systems, and appropriate logics have been proposed to describe more subtle properties to be verified. However, some characteristics of such systems cannot fall in the field of model checking. The aim is thus not to tell wether a property is satisfied but how well the system performs with respect to a certain measure. Together with researchers from ENS de Cachan and University Paris Est Créteil we have designed a statistical tool made to tackle both performance and verification issues. Following several conference talks, a journal paper is currently written to present both the approach as well as application to a concrete case study: flexible manufacturing systems.